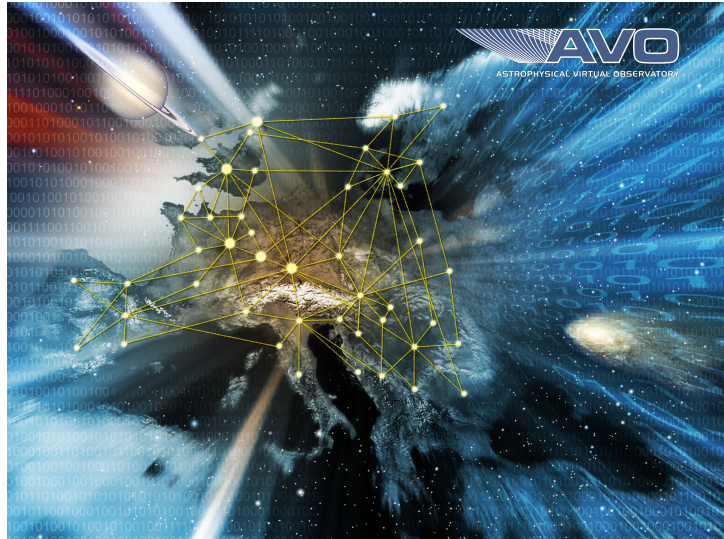


# Spektralanalyse im Virtuellen Observatorium



**Dissertation**

der Mathematisch-Naturwissenschaftlichen Fakultät  
der Eberhard Karls Universität Tübingen  
zur Erlangung des Grades eines  
Doktors der Naturwissenschaften  
(Dr. rer. nat.)

vorgelegt von

**Ellen Michaela Müller-Ringat**

aus Tübingen

Tübingen  
2013

Titelbild: Künstlerische Interpretation des AVO – einer der ersten Organisationen des Virtuellen Observatoriums.  
Quelle: ESO, ESA, ASTROGRID, CDS (CNRS/ULP), TERAPIX (CNRS) and Jodrell Bank Observatory.

Tag der mündlichen Qualifikation:

20.03.2013

Dekan:

Prof. Dr. Wolfgang Rosenstiel

1. Berichterstatter:

Prof. Dr. Klaus Werner

2. Berichterstatter:

PD Dr. Thorsten Nagel

# **Spektralanalyse im Virtuellen Observatorium**

**Dissertation**

der Mathematisch-Naturwissenschaftlichen Fakultät  
der Eberhard Karls Universität Tübingen  
zur Erlangung des Grades eines  
Doktors der Naturwissenschaften  
(Dr. rer. nat.)

vorgelegt von

**Ellen Michaela Müller-Ringat**

aus Tübingen

Tübingen  
2013

Tag der mündlichen Qualifikation:

Dekan:

1. Berichterstatter:

2. Berichterstatter:

20.03.2013

Prof. Dr. Wolfgang Rosenstiel

Prof. Dr. Klaus Werner

PD Dr. Thorsten Nagel

## Zusammenfassung

Die Erzeugung astronomischer Daten durchläuft eine rasante Entwicklung. Mittlerweile liefern Teleskope kontinuierlich mehrere Terabyte an Beobachtungsdaten pro Tag. Simulationen benötigen nur wenige Tage um mehrere hundert Terabyte an Daten zu produzieren. Das Ende der Erzeugung immer größerer Datenmengen in immer kürzerer Zeit ist jedoch noch lange nicht erreicht. In etwa zehn Jahren wird die Exabyte-pro-Tag-Grenze überschritten werden. Um mit dieser Entwicklung mithalten und die Daten auch verarbeiten zu können wurde Anfang des Jahrtausends das Virtuelle Observatorium (VO) gegründet. Ziel des VOs ist es, die astronomischen Daten über das Internet zugänglich zu machen, Standards für eine einheitliche Verarbeitung zu definieren und intuitiv zu bedienende Analysewerkzeuge zu entwickeln, die eine schnelle Datenverarbeitung ermöglichen und auf viele unterschiedliche Datensätze angewandt werden können. Vor allem durch die Bemühungen des VOs ist es möglich, Astronomie nach heutigen Standards zu betreiben.

Der deutsche Beitrag zum VO wird durch das German Astrophysical Virtual Observatory (GAVO) geleistet. Im Rahmen von GAVO wurden vom Institut für Astronomie und Astrophysik Tübingen (IAAT) VO-Dienste entwickelt, über die theoretische Sternspektren über das Internet zur Verfügung gestellt werden und Spektralanalyse jedem einfach und ohne viel Hintergrundwissen ermöglicht wird. Mit den Diensten können Spektralanalysen auf drei verschiedenen Niveaus durchgeführt werden. Der Nutzer kann bereits berechnete synthetische Spektren über den *TheoSSA*-Dienst herunterladen. Für genauere Analysen können über den *TMAW*-Dienst individuelle Spektren erzeugt werden. Erfahrene Nutzer können die zugrunde liegenden Atomdaten aus der *TMAD*-Datenbank herunterladen und für eigene Sternatmosphärenprogramme verwenden. Die vorliegende Arbeit beschäftigt sich mit der Weiterentwicklung dieser Dienste. Dafür wurden sowohl die bestehenden Dienste erweitert als auch neue Dienste erzeugt und diese in Spektralanalysen angewandt.

**TIRO:** Mit diesem neu entwickelten Dienst kann der VO-Nutzer individuelle Querschnitte und Modellatome für Eisengruppenelemente berechnen lassen. Motivation für die Einführung des *TIRO*-Dienstes ist, dass die Eisengruppenelemente starke Auswirkungen auf die Atmosphärenstruktur und damit auf die gesamte Analyse haben können. Sie verursachen die Line-Blanketing-Effekte, die sich auch auf die Linienprofile anderer Elemente auswirken. Die Berücksichtigung der Eisengruppenelemente erfolgt aufgrund ihrer Vielzahl an Energieniveaus und zugehörigen Linienübergängen jedoch nicht analog zu anderen Elementen. Ein statistisches Verfahren ist nötig, um die Energieniveaus zu kombinieren und die Querschnitte zu sampeln und so zu wenigen zusammenzufügen. Dieses Verfahren wird in einem separaten Schritt vor der Modellatmosphärenberechnung durchgeführt und basiert auf dem bestehenden *IrOnIc*-Programm.

Um eine Verbesserung und Beschleunigung des *IrOnIc*-Programms zu erreichen, wurden einige seiner Parameter und Unterprogramme getestet und die Bedienung des Programms durch den webbasierten *TIRO*-Dienst verändert. Damit können realistischere Querschnitte zweimal schneller als zuvor erzeugt werden. Zusätzlich wurde eine Datenbank aufgebaut, basierend auf einem sehr feinen Frequenzgitter. Bei jeder Anfrage wird nun geprüft, ob eine Interpolation der Querschnitte aus dieser Datenbank möglich ist. Damit wird die Wartezeit von Tagen auf wenige Minuten verkürzt. Ist dies nicht möglich, wird die Berechnung auf einem von Nutzer hochgeladenen oder nach Nutzervorgaben automatisch erzeugten Frequenzgitter durchgeführt. Die dem Programm zugrunde liegenden Atomdaten werden ständig erweitert und (zum Teil im Rahmen dieser Arbeit) für das *IrOnIc*-Programm regelmäßig aktualisiert.

**TMAW:** Mit diesem Dienst kann der VO-Nutzer individuelle Sternspektren berechnen lassen. Dafür müssen nur die fundamentalen Parameter der Sternatmosphäre auf einer Webseite angegeben werden, die Berechnung wird anschließend automatisch durchgeführt. Für den bestehenden *TMAW*-Dienst wurde eine Qualitätskontrolle eingeführt, die dem Administrator die Betreuung des Dienstes erleichtert und dem Nutzer Informationen über die Genauigkeit der resultierenden synthetischen Spektren im Vergleich zu von einem Experten mit dem selben Sternatmosphärencode berechneten, detaillierteren Spektren liefert. Außerdem wurde die Modellierung von *TMAW*-Spektren auf Computern, die über das *AstroGrid-D*

zugänglich sind, ausgebaut und die Berechnung der synthetischen Modelle um weitere Schritte ergänzt, die die Genauigkeit der *TMAW*-Spektren steigern, aber dennoch wenig zusätzliche Rechenzeit benötigen.

**Anwendung von *IrOnIc* und *TMAW* in der Spektralanalyse:** Im Rahmen der Einführung einer Qualitätskontrolle für *TMAW* wurden differentielle Spektralanalysen für ausgewählte heliumreiche, heiße unterleuchtkräftige Sterne und PG 1159-Sterne durchgeführt und deren Ergebnisse mit *TMAW*-Resultaten verglichen. Für die unterleuchtkräftigen Sterne wurde dafür zuerst eine Analyse mit HHe-, später HHeCNO-Modellen durchgeführt, die eine niedrigere Effektivtemperatur der Objekte zum Ergebnis hat. Die Parameter der PG 1159-Sterne wurden zuerst mit HHeCNO-Modelle ermittelt. Ein Vergleich der jeweils resultierenden Modelle beider Objektklassen mit *TMAW*-Modellen zeigt, dass *TMAW* für Analysen, die maximal 4% bis 12% in der resultierenden Effektivtemperatur und 4% bis 6% in der Oberflächenschwerebeschleunigung von den detaillierteren Analysen mit dem selben Sternatmosphärencode abweichen, geeignet ist. Diese Genauigkeit übertrifft die Erwartungen (10% bei den Flüssen und 20% bei den aus einer Analyse resultierenden Parametern) deutlich.

Um die Auswirkung der neuen Atomdaten für Eisengruppenelemente auf das Ergebnis einer Analyse zu untersuchen wurden in die finalen Modelle der heißen unterleuchtkräftigen Sterne Eisengruppenelemente eingefügt. Für die PG 1159-Sterne wurde eine vollständige Analyse, die zusätzlich zu den Eisengruppenelementen auch Spurenelemente einschließt, durchgeführt. Dadurch kann gezeigt werden, dass sich die aus einer Analyse resultierende Effektivtemperatur und Oberflächenschwerebeschleunigung bei Vernachlässigung der Eisengruppenelemente um bis zu ca. 10% ändern kann.

Außerdem wurden die Auswirkungen der erweiterten Atomdaten für Eisengruppenelemente auf den Flussverlauf von Modellen wasserstoffreicher, heißer unterleuchtkräftiger Sterne untersucht, bei denen extrem hohe Eisengruppenhäufigkeiten vermutet bzw. in früheren, optischen Analysen festgestellt wurden. Es zeigt sich, dass die Eisengruppenelemente eine deutliche Absenkung des UV-Flusses verursachen können, die in den Beobachtungen bisher nicht erklärt werden konnte. Die dafür nötigen stark erhöhten Eisengruppenhäufigkeiten haben außerdem Einfluss auf die Balmerlinien. Für eine Analyse einzelner Spektrallinien der Eisengruppenelemente sind die Unsicherheiten in den Atomdaten allerdings zu hoch und die spektrale Auflösung der vorhandenen Beobachtungen nicht ausreichend. Ein weiteres Resultat ist, dass im Rahmen dieser Arbeit berechnete Diffusionsmodelle die extrem hohen Eisengruppenhäufigkeiten erklären können, chemisch homogene Atmosphären die Beobachtung für die untersuchten Objekte allerdings besser reproduzieren.

***TMAD*:** Ein weiterer Beitrag zur Spektralanalyse über das Internet ist die Erweiterung der *TMAD*-Modellatomdatenbank. Durch die neuen Ionen sind Häufigkeitsbestimmungen zusätzlicher Elemente möglich, wodurch z.B. weitere Rückschlüsse auf die Fusionsprozesse in Sternen möglich sind.

***TEUV*:** Im EUV-Wellenlängenbereich hat die interstellare Absorption, verursacht durch gebunden-frei-Übergänge einzelner Ionen großen Einfluss auf den beobachteten Flussverlauf von Sternen. Existierende Programme modellieren diese Absorption meist nur für die Ionen H I, He I und He II. Zur interstellaren Absorption im EUV-Bereich können zusätzlich die Ionen der Elemente C, N und O bis zur Ionisationsstufe III beitragen. Der neue *TEUV*-Dienst ermöglicht dem *VO*-Nutzer die Modellierung der Absorption durch Ionen von H, He, C, N und O basierend auf Opacity Project-Daten. Gesteuert wird er über eine im Rahmen dieser Arbeit entwickelte Webseite, auf der der Nutzer (interaktiv) verschiedene Parameter, die das interstellare Medium beschreiben, anpassen kann.

Die im Rahmen dieser Arbeit durchgeführten Tests anhand verschiedener Objekte zeigen, dass die neuen und erweiterten Tübinger *VO*-Dienste einen wichtigen Beitrag zur Spektralanalyse im *VO* liefern. Sie ermöglichen die Analyse unterschiedlicher Objekte – ohne selbst Rechenzeit in die Modellrechnung zu investieren – sowohl einer breiten Nutzergemeinde als auch den Spezialisten. Durch die neuen Dienste können Spektralanalysen über das *VO* realistischer und mit einer höheren Genauigkeit als zuvor durchgeführt werden.

## Summary

Astronomical data production experiences a rapid evolution. Today's telescopes steadily produce several terabyte of observational data each day. Simulations need only few days to generate a few hundred terabyte of data. Generating more and more data in less time will go on. The exabyte-per-day limit will be reached in about ten years. To keep up with this evolution and to be able to handle the amount of data, the Virtual Observatory (*VO*) was established at the beginning of the century. Its aims are making astronomical data accessible via the internet, developing standards for uniform data handling and generating intuitively operated tools for fast processing of as many datasets as possible. Doing astronomy by today's standards is mainly possible because of these efforts.

The German contribution to the *VO* is called German Astrophysical Virtual Observatory (*GAVO*). Within the *GAVO* project, the Institute for Astronomy and Astrophysics Tübingen developed *VO* services to make theoretical stellar spectra accessible and furthermore to enable everybody to perform spectral analysis easily and without detailed knowledge. With these services spectral analysis can be done on three levels. Pre-calculated synthetic spectra can be downloaded via the *TheoSSA* service. For more detailed analyses individual spectra can be created with the *TMAW* service. Experienced users can download the underlying atomic data from the *TMAD* database to use it as input for their own model-atmosphere codes. This work deals with the further development of these services by improving the existing services and creating new ones. Additionally the new and improved services were tested in spectral analyses.

***TIRO*:** This new service enables the user to generate cross-sections and model atoms for iron-group elements. The motivation behind this tool is the possible strong effect of iron-group elements on the structure of the stellar atmosphere and thereby on the result of the analysis. They cause the line blanketing effects that also affect lines of other species. Due to the huge number of energy levels and lines iron-group elements cannot be accounted for as other elements. Hence a statistical treatment is used to combine the levels and sample the lines. This treatment is already implemented in the existing *IrOnIc* program that is used in a separate step executed before of the model atmosphere is calculated.

To improve and accelerate the *IrOnIc* program, several of its parameters and subroutines were tested and the handling of the program (via the web-based *TIRO* service) was restructured. More realistic cross-sections can be calculated two times faster now. Additionally a database based on a very fine frequency grid was built up. *TIRO* checks every request if an interpolation between database values is possible. This reduces the waiting time from days to minutes. If this is not possible, the calculation is performed for an uploaded or specifically created frequency grid. The underlying atomic data for *IrOnIc* are constantly extended and (partly within this work) regularly updated.

***TMAW*:** This service provides the user with individually calculated synthetic spectra. Therefore only the fundamental parameters have to be entered on the web interface, the calculation is performed automatically. A quality control was established for the *TMAW* service that enables the user to judge the quality of the resulting synthetic spectra compared to more detailed spectra calculated by an expert with the same model-atmosphere code. Furthermore it enables easier handling of the service by the administrator. Additionally the calculation of *TMAW* models on computer resources of *AstroGrid-D* was improved. Extra calculation steps were introduced for all *TMAW* calculations to enhance the accuracy of the models with only little additional calculation time.

**Application of *IrOnIc* and *TMAW* to spectral analysis:** Within the establishment of a quality control for the *TMAW* service differential spectral analyses were performed for heliumrich, hot subdwarf stars and PG 1159 stars. The outcomes were compared to *TMAW* results. For the hot subdwarf stars an analysis with HHe, later HHeCNO models was performed. The resulting effective temperatures are lower than the literature values. The parameters of the PG 1159 stars were determined with HHeCNO models first. A comparison of the resulting models of both object classes to *TMAW* models shows that *TMAW* is suitable for analyses that deviate maximum 4% to 12% in effective temperature and 4% to 6% in surface

gravity from results of more detailed analyses performed with the same model-atmosphere code. This accuracy clearly exceeds the expectations (10% in fluxes and 20% in resulting parameters of an analysis).

To investigate the effects of the new iron-group atomic data on the results of the analysis, iron-group elements were included in the final models of the hot subdwarf stars. For the PG 1159 stars a complete analysis was performed that includes trace elements additionally to iron-group elements. These investigations show that the resulting effective temperature and surface gravity of an analysis can be changed by about 10% by including iron-group elements.

Besides, the effect of the updated iron-group data on the overall flux shape of hot subdwarf star models was examined. Very high iron-group abundances were suspected in the atmospheres or found in previous optical analyses of these stars. The investigation shows that iron-group elements can cause very low fluxes in the UV wavelength range that could not be explained in the observations by now. The high iron-group abundances also have effects on the Balmer lines. However, the underlying uncertainties of the atomic data are too high and the spectral resolution of the available observations is not good enough for detailed line profile analysis of iron-group elements. Another result is that diffusion models calculated within this work confirm these high iron-group abundances but the observations of the considered objects are reproduced better with homogeneous models.

**TMAD:** Another contribution to enable spectral analyses via the internet is the extension of the *TMAD* model-atom database. With the new ions, abundance determinations of additional elements are possible whereby e.g. further conclusions on nuclear processes in stars can be drawn.

**TEUV:** In the EUV wavelength range, interstellar absorption caused by bound-free absorption of ions has a big influence on the observed fluxes of stars. Existing programs model these effects mainly for the ions H I, He I und He II. However, ions of the elements C, N, and O with ionization stages up to III can be conducive to interstellar absorption in the EUV range. The newly created *TEUV* service enables the user to model interstellar bound-free absorption caused by ions of the elements H, He, C, N, and O based on Opacity Project data. It is controlled via a newly developed web interface where the user can (interactively) adapt the parameters of the interstellar medium.

The tests performed within this work and based on various objects show that the newly created and extended *VO* services built in Tübingen importantly contribute to spectral analyses in the *VO*. The services enable the community as well as specialists to analyze different objects – without spending computing time to calculate the stellar atmosphere. With the new services spectral analyses via the *VO* can be performed with higher accuracy than before.



# Inhaltsverzeichnis

<b>Tabellenverzeichnis</b>	<b>vii</b>
<b>Abbildungsverzeichnis</b>	<b>ix</b>
<b>Unix-Shell-Skripte und FORTRAN-Programme</b>	<b>xiii</b>
<b>1. Einleitung</b>	<b>1</b>
1.1. Spektralanalyse und Sternentwicklung . . . . .	3
1.2. Das Virtuelle Observatorium . . . . .	9
1.3. Aufbau und Ziele der Arbeit . . . . .	14
<b>2. Querschnitte und Modellatome für Eisengruppenelemente über das VO</b>	<b>17</b>
2.1. <i>IrOnIc</i> – Statistische Behandlung der Eisengruppenelemente . . . . .	19
2.1.1. Bedienung des <i>IrOnIc</i> -Programms . . . . .	20
2.2. Die Quelle für Eisengruppendaten: Linienlisten von Kurucz . . . . .	20
2.3. Die <i>IrOnIc</i> - und <i>TIRO</i> -Parameter . . . . .	24
2.3.1. Energiebänder und Superlevels . . . . .	24
2.3.2. Algorithmen für die Berechnung der Voigtfunktion . . . . .	25
2.3.3. Zusätzliche Ionen . . . . .	27
2.3.4. Generische Modellatome – Die Wahl der Ionisationsenergie . . . . .	27
2.3.5. Linientemperatur und Zahl der Frequenzpunkte . . . . .	28
2.3.6. Das Intervall zur Berechnung der Voigtprofile . . . . .	30
2.4. Der <i>TIRO</i> -Dienst . . . . .	33
<b>3. Das Tübinger GAVO-Projekt und seine Weiterentwicklung</b>	<b>37</b>
3.1. <i>TMAP</i> – Das Tübinger Modellatmosphärenpaket . . . . .	37
3.2. <i>TMAD</i> – Eine Modellatomdatenbank . . . . .	38
3.3. <i>TheoSSA</i> – Eine Datenbank für synthetische Sternspektren . . . . .	41
3.4. <i>TMAW</i> – Berechnung individueller synthetischer Spektren . . . . .	45
3.4.1. Hintergrundoperationen nach dem Absenden eines Requests . . . . .	45
3.5. Qualitätskontrolle . . . . .	53
3.5.1. Benchmarktests . . . . .	54
3.5.2. Weitere Änderungen am <i>TMAW</i> -Dienst zur Einführung einer Qualitätskontrolle . . . . .	59
3.6. <i>TMAW</i> -Rechnungen über das <i>AstroGrid-D</i> . . . . .	61
<b>4. GAVO-Dienste in der Spektralanalyse</b>	<b>65</b>
4.1. Die heliumreichen Sterne LSE 153 und LSE 263 . . . . .	65
4.1.1. Eigenschaften und Entwicklung unterleuchtkräftiger Sterne . . . . .	66
4.1.2. LSE 153 und LSE 263 . . . . .	68
4.2. Die PG 1159-Sterne PG 1424+535 und PG 1707+427 . . . . .	87
4.2.1. Entstehung und Entwicklung von PG 1159-Sternen . . . . .	87
4.2.2. PG 1424+535 und PG 1707+427 . . . . .	89
4.3. sd(O)B-Sterne mit extrem hohen Eisengruppenhäufigkeiten . . . . .	102
4.3.1. Der Stern EC 11481–2303 . . . . .	103

4.3.2.	Die Sterne UVO 1758+36 und PG 0909+276 . . . . .	109
4.4.	(Interstellare) Absorption im EUV-Bereich . . . . .	120
4.4.1.	Eigenschaften von DO-Weißen Zwergen . . . . .	120
4.4.2.	RE 0503–289 und das EUV-Problem . . . . .	121
4.4.3.	Neue Ansätze zur Lösung des EUV-Problems . . . . .	122
4.4.4.	<i>TEUV</i> – Korrektur der interstellaren Absorption . . . . .	123
<b>5.</b>	<b>Zusammenfassung und Diskussion</b>	<b>135</b>
5.1.	Die neuen und weiterentwickelten Tübinger <i>VO</i> -Dienste . . . . .	135
5.2.	Anwendung der Dienste in der Spektralanalyse . . . . .	136
5.3.	Ausblick . . . . .	139
	<b>Literaturverzeichnis</b>	<b>141</b>
<b>A.</b>	<b>Anhang</b>	<b>149</b>
A.	<i>TIRO</i> -Prozeduren . . . . .	149
B.	<i>TMAW</i> -Prozeduren . . . . .	182
B.1.	Die Hauptprozeduren . . . . .	182
B.2.	<i>TMAW</i> -Prozeduren, die regelmäßig automatisch laufen . . . . .	241
B.3.	<i>TMAW</i> -Prozeduren zum Überprüfen der Requests nach Bedarf . . . . .	252
C.	<i>TMAW</i> -Prozeduren für die Berechnung über das <i>AstroGrid-D</i> . . . . .	260
D.	<i>TEUV</i> -Programme und Skripte . . . . .	309

# Tabellenverzeichnis

2.1. Statistik der Eisengruppenelemente in der Kurucz-Datenbank . . . . .	22
2.2. Zwei Energieniveaus der GF2701.GAM-Datei . . . . .	24
2.3. Berechnungsdauer der gebunden-gebunden-Querschnitte von Eisengruppenelementen . .	32
3.1. Statistik ausgewählter <i>TMAW</i> -Modellatome . . . . .	50
3.2. Wie Tabelle 3.1, für die Lineformation-Rechnung. . . . .	51
3.3. Die einzelnen Schritte der <i>TMAW</i> -Berechnung . . . . .	52
4.1. Parameter von LSE 153 und LSE 263 . . . . .	69
4.2. Statistik der HHeCNO-Modellatome für die <i>TMAP</i> -Rechnung. . . . .	73
4.3. Statistik der HHeCNO-Modellatome für die Lineformationrechnung. . . . .	74
4.4. Vergleich der Parameter mit den Literaturwerten für die sdO-Sterne . . . . .	74
4.5. Die Elementhäufigkeiten von PG 1424+535 und PG 1707+427 . . . . .	90
4.6. Scheinbare Helligkeiten von EC 11481–2303. . . . .	104
4.7. Scheinbare Helligkeiten der drei sd(O)B-Sterne . . . . .	110
4.8. Die Elementhäufigkeiten von PG 0909+276 . . . . .	113
4.9. Die Elementhäufigkeiten von UVO 1758+36 . . . . .	115
4.10. Die Elementhäufigkeiten der Modelle von PG 0909+276 und UVO 1758+36 . . . . .	115
4.11. Frühere Analysen von RE 0503–289 . . . . .	121
4.12. Obergrenzen der Eisen- und Nickelhäufigkeit von RE 0503–289 aus früheren Analysen .	123
4.13. Grundzustandswellenlängen der von <i>TEUV</i> verwendeten Ionen. . . . .	125
4.14. Die Säulendichten der Elemente des Interstellaren Mediums zu RE 0503–289 . . . . .	131



# Abbildungsverzeichnis

1.1.	Entwicklung der Speicherkapazität von Festplatten . . . . .	1
1.2.	Modellspektren zweier Sternatmosphären . . . . .	3
1.3.	Fraunhoferlinien . . . . .	4
1.4.	Hertzprung-Russell-Diagramm mit Entwicklungswegen verschiedener Sterne . . . . .	5
1.5.	Katzenaugennebel, Helixnebel, Stundenglasnebel und Ameisennebel . . . . .	6
1.6.	Durchlässigkeit der Erdatmosphäre . . . . .	7
1.7.	Ungefährer Gültigkeitsbereich von LTE- und NLTE-Modellen . . . . .	8
1.8.	Mitgliedsorganisationen der IVOA. . . . .	12
2.1.	Lage von radiativen gebunden-gebunden-Querschnitten von verschiedenen Fe-Ionen . . . . .	17
2.2.	Ionisationsbruchteile der Fe-Ionen V - VIII bei verschiedenen Temperaturen. . . . .	18
2.3.	Vergleich zweier RBB-Querschnitte aus alten und neuen Atomdaten . . . . .	21
2.4.	Linienprofile berechnet mit den Routinen von Hellwege und Curtis . . . . .	26
2.5.	Vergleich der Rechenzeiten der schnellsten getesteten Voigt-routinen . . . . .	26
2.6.	Ionisationsenergien der Eisengruppenelemente für die Ionisationsstufen III und IV . . . . .	28
2.7.	Radiative gebunden-gebunden-Querschnitte generischer Modellatome . . . . .	28
2.8.	Auswirkung der Linientemperatur auf die Temperaturstruktur . . . . .	29
2.9.	EUV-Fluss von HFe-Modellen in Abhängigkeit von $T_L$ . . . . .	30
2.10.	Auswirkung der Frequenzgitter auf die Temperaturstruktur . . . . .	31
2.11.	Auswirkung der Frequenzgitter auf den EUV-Fluss . . . . .	31
2.12.	Vergleich von Querschnitten von Fe VII . . . . .	32
2.13.	<i>TIRO</i> -Webseite. . . . .	33
2.14.	Schematische Übersicht des <i>TIRO</i> -Ablaufs. . . . .	35
3.1.	<i>TMAD</i> -Webseite – verfügbare Elemente . . . . .	39
3.2.	<i>TMAD</i> -Webseite – Neon . . . . .	40
3.3.	Grotriandiagramm von N IV . . . . .	40
3.4.	<i>TheoSSA</i> -Webseite . . . . .	42
3.5.	<i>TheoSSA</i> -Ergebnisliste . . . . .	43
3.6.	Auszug aus der IVOA-Dokumentation für SSA . . . . .	44
3.7.	<i>TMAW</i> -Request-Webseite. . . . .	46
3.8.	Zahl der extern eingegangenen <i>TMAW</i> -Requests und der angeforderten Modelle . . . . .	46
3.9.	Schematische Übersicht über die grundlegenden <i>TMAW</i> -Schritte. . . . .	48
3.10.	Schematische Übersicht über die Aufgaben der <i>TMAW</i> -Hauptprozeduren. . . . .	49
3.11.	Vergleich zweier <i>TMAP</i> - und <i>TMAW</i> -Flüsse . . . . .	55
3.12.	Vergleich der Linienprofile der H $\beta$ - und H $\delta$ -Linien des <i>TMAP</i> - und <i>TMAW</i> -Modells . . . . .	56
3.13.	Vergleich von Temperaturstrukturen berechnet mit <i>TMAP</i> und <i>TMAW</i> . . . . .	57
3.14.	Vergleich der Linienprofile ausgewählter Linien für eine NLTE- und LTE-Rechnung . . . . .	58
3.15.	Rechenzeiten verschiedener <i>TMAW</i> -Modelle mit unterschiedlichen Elementen . . . . .	58
3.16.	$\log T_{\text{eff}} - \log g$ -Diagramm zur Übersicht . . . . .	59
3.17.	Berechnung von <i>TMAW</i> -Requests über das <i>AstroGrid-D</i> . . . . .	62
4.1.	sdB- und sdO-Sterne im HRD . . . . .	67

4.2. Gitter der <i>TMAW</i> -Modelle für die sdO-Objekte . . . . .	70
4.3. Erfolgreich beendete <i>TMAW</i> -Modellrechnungen für unterschiedliche Berechnungsarten . . . . .	70
4.4. Vergleich zweier <i>TMAP</i> - und <i>TMAW</i> -Temperaturstrukturen . . . . .	71
4.5. Vergleich zweier <i>TMAP</i> - und <i>TMAW</i> -Flüsse . . . . .	71
4.6. Vergleich von HHe-Modellen, berechnet mit unterschiedlicher Levelzahl . . . . .	72
4.7. <i>TMAP</i> -Modelle verglichen mit dem CASPEC-Spektrum von LSE 153 . . . . .	75
4.8. Wie Abb. 4.7, für LSE 263 . . . . .	76
4.9. <i>TMAP</i> -HHeCNO-Modelle verglichen mit dem <i>TMAW</i> -Modell für LSE 263 . . . . .	77
4.10. <i>TMAP</i> -Modelle verglichen mit dem <i>TMAW</i> -Modell für LSE 153 . . . . .	78
4.11. <i>TMAP</i> -Modell von LSE 263 verglichen mit FUSE- und IUE-Ausschnitten . . . . .	80
4.12. <i>TMAP</i> -Modell von LSE 153 verglichen mit FUSE- und IUE-Ausschnitten . . . . .	81
4.13. <i>TMAW</i> - und <i>TMAP</i> -Modelle von LSE 153 im UV-Bereich . . . . .	82
4.14. Temperaturstruktur des HHeCNO- und HHeCNOIG-Modells von LSE 263 . . . . .	82
4.15. Die stärksten Eisen- und Nickellinien von LSE 153 . . . . .	83
4.16. Die stärksten Eisen- und Nickellinien von LSE 263 . . . . .	83
4.17. Flussverlauf des HHeCNO- und HHeCNOIG-Modells von LSE 263 . . . . .	84
4.18. Ionisationsbruchteile des HHeCNO- und HHeCNOIG-Modells von LSE 263 . . . . .	84
4.19. Linienprofile von He des HHeCNO- und HHeCNOIG-Modells von LSE 263 . . . . .	85
4.20. Linienprofile von He des HHeCNO- und HHeCNOIG-Modells von LSE 153 . . . . .	86
4.21. Innere Struktur eines Post-AGB-Sterns . . . . .	88
4.22. Entwicklungsweg eines $2.7 M_{\odot}$ Sterns mit einem VLTP . . . . .	88
4.23. Die Elementhäufigkeiten von PG 1424+535 und PG 1707+427 . . . . .	91
4.24. Ausgewählte C-, N- und O-Linien von PG 1707+427 . . . . .	92
4.25. Ausgewählte C-, N- und O-Linien von PG 1424+535 . . . . .	93
4.26. Ausschnitte aus den FUSE-Beobachtungen von PG 1424+535 und PG 1707+427 . . . . .	94
4.27. Ausschnitte aus den FUSE-Beobachtungen von PG 1424+535 und PG 1707+427 . . . . .	95
4.28. Eisenlinien von PG 1424+535 . . . . .	96
4.29. Eisenlinien von PG 1707+427 . . . . .	97
4.30. Nickellinien von PG 1707+427 . . . . .	97
4.31. Temperaturstrukturen der Modelle von PG 1707+427 . . . . .	98
4.32. Temperatursensitive UV-Linien von PG 1707+427 verglichen mit HHeCNO <sub>He</sub> - und HHeCNO <sub>He</sub> FSiPArCaFeNi-Modellen . . . . .	98
4.33. Wie Abb. 4.32, für PG 1424+535 . . . . .	99
4.34. Linienprofile der optischen Heliumlinien verglichen mit den Grenzmodellen . . . . .	99
4.35. Wie Abb. 4.32, für das <i>TMAW</i> - anstelle des HHeCNO <sub>He</sub> FSiPArCaFeNi-Modells. . . . .	100
4.36. Wie Abb. 4.33, für das <i>TMAW</i> - anstelle des HHeCNO <sub>He</sub> FSiPArCaFeNi-Modells. . . . .	100
4.37. Wie Abb. 4.34, für die <i>TMAW</i> -Modelle . . . . .	101
4.38. UV-Flussverlauf der ausgewählten sd(O)B-Sterne . . . . .	103
4.39. UV-Flussverlauf von EC 11481–2303 und die <i>TMAP</i> -Modelle . . . . .	105
4.40. UV-Flussverlauf von EC 11481–2303 und ein Diffusionsmodell . . . . .	106
4.41. Häufigkeiten der Eisengruppenlemente im Diffusionsmodell von EC 11481–2303 . . . . .	106
4.42. Ausschnitt aus der FUSE-Beobachtung von EC 11481–2303 verglichen mit dem <i>TMAP</i> -Modell . . . . .	107
4.43. Ausschnitt aus der FUSE-Beobachtung von EC 11481–2303 verglichen mit dem <i>NGRT</i> -Modell . . . . .	107
4.44. Balmerlinien von EC 11481–2303 . . . . .	108
4.45. Ausschnitt aus der FUSE-Beobachtung von EC 11481–2303 und das <i>OWENS</i> -Modell . . . . .	109
4.46. Ionisationsbruchteile von Co . . . . .	110
4.47. Bestimmung der interstellaren Extinktion von PG 0909+276 . . . . .	111
4.48. UV-Flussverlauf von PG 0909+276 . . . . .	112

4.49. Ausschnitt aus der IUE-Beobachtung von UVO 1758+36 . . . . .	114
4.50. Elementhäufigkeiten der Diffusionsmodelle von PG 0909+276 und UVO 01758+36 . . .	116
4.51. UV-Flussverlauf von PG 0909+276 und UVO1758+36 und das Diffusionsmodell . . . .	116
4.52. Linienprofile der Balmerreihe von PG 0909+276 . . . . .	117
4.53. Temperaturstrukturen des HHeCNO-, HCaScTiVCrMnFeCoNi- und HHeCNOCaScTiVCrMnFeCoNi-Modells von PG 0909+276 . . . . .	117
4.54. Die Linienprofile der Balmerreihe von UVO 1758+26 . . . . .	117
4.55. Wie Abb. 4.53, für UVO 1758+36. . . . .	118
4.56. Abb. 4 aus Geier et al. (2010) (modifiziert) . . . . .	119
4.57. FUSE- und EUVE-Spektrum von RE 0503–289 . . . . .	124
4.58. <i>TEUV</i> -Webseite. . . . .	126
4.59. Schematische Darstellung des neuen <i>TEUV</i> -Dienstes. . . . .	127
4.60. Webseite des neuen <i>TEUV</i> -Dienstes nach Upload des Modells . . . . .	128
4.61. Funktionsweise des neuen <i>TEUV</i> -FORTRAN-Programms. . . . .	129
4.62. Lyman $\beta$ -Linie des HHeCNO-Modells von RE 0503–289 . . . . .	130
4.63. FUSE- und EUVE-Spektrum von RE 0503–289 modelliert mit IRAF . . . . .	130
4.64. FUSE- und EUVE-Spektrum von RE 0503–289 modelliert mit <i>TEUV</i> . . . . .	131
4.65. FUSE- und EUVE-Spektrum von RE 0503–289 modelliert mit <i>TEUV</i> (Literatur) . . . .	132
4.66. FUSE- und EUVE-Spektrum von RE 0503–289 modelliert mit <i>TEUV</i> (Ge, Kr, Xe) . . .	133





# Unix-Shell-Skripte und FORTRAN-Programme

A.1.	<i>TIRO</i> :	checkrun.bat . . . . .	150
A.2.		run_ions.bat . . . . .	151
A.3.		startjob.bat . . . . .	159
A.4.		ironic_combine_web.job . . . . .	160
A.5.		ChangeIrOnIcHeaders . . . . .	162
A.6.		run_inter.sh . . . . .	163
A.7.		interpol.sh . . . . .	172
A.8.		ironic_CaSc.in . . . . .	178
A.9.		ironic_CaSc.job . . . . .	181
A.10.	<i>TMAW</i> :	checkreq.sh . . . . .	183
A.11.		TMAW_iaat_run.sh . . . . .	189
A.12.		TMAW_iaat_prepare.sh . . . . .	203
A.13.		TMAW_iaat_jobfile.sh . . . . .	222
A.14.		checkPARA.sh . . . . .	238
A.15.		resetPARA.sh . . . . .	241
A.16.		compile_MASTER.sh . . . . .	242
A.17.		jobcontrol.sh . . . . .	244
A.18.		jobcontrol2.sh . . . . .	245
A.19.		checktestgridH.sh . . . . .	246
A.20.		checktestgridH2.sh . . . . .	250
A.21.		checktestgridH3.sh . . . . .	251
A.22.		status_check.sh . . . . .	252
A.23.		status_check2.sh . . . . .	253
A.24.		status_check3.sh . . . . .	254
A.25.		delete_lastline.sh . . . . .	255
A.26.		restart.bat . . . . .	256
A.27.		restartall.bat . . . . .	257
A.28.		models_cleanup.sh . . . . .	258
A.29.		maintenance.sh . . . . .	259
A.30.		TMAW_grid_run.sh . . . . .	260
A.31.		TMAW_grid_prepare.sh . . . . .	267
A.32.		TMAW_grid_jobfile.sh . . . . .	287
A.33.		scp2li04.bat . . . . .	302
A.34.		scp2IAAT.bat . . . . .	304
A.35.		checkgridH.bat . . . . .	305
A.36.		checkgridH2.bat . . . . .	307
A.37.		checkquality.bat . . . . .	308
A.38.	<i>TEUV</i> :	TEUV.f90 . . . . .	309
A.39.		gaussd.f90 . . . . .	317
A.40.		gaussf.f90 . . . . .	318
A.41.		convolg.f90 . . . . .	319
A.42.		readOP.f90 . . . . .	320

A.43.	interpol.f90 . . . . .	322
A.44.	TEUV1.f90 . . . . .	323
A.45.	readOP1.f90 . . . . .	330
A.46.	TEUV.sh . . . . .	332

# 1. Einleitung

Seit Ende der 1990er Jahre erfolgt die Produktion astronomischer Daten mit einer stark beschleunigten Entwicklung. Sie folgt dabei dem Moore'schen Gesetz (Moore, 1965), das die Verdopplung der Zahl der Transistoren auf einem Computerchip etwa alle 18 Monate beschreibt (Abb. 1.1<sup>1</sup>). Dies liegt zum einen an immer größeren Teleskopen und Detektoren, zum anderen hat sich der Betrieb der Teleskope und ihrer Instrumente gewandelt. Multi-Objekt-Spektrographen liefern mittlerweile Spektren Hunderter Objekte gleichzeitig und mit so genannten Survey-Teleskopen wird der gesamte Himmel kontinuierlich beobachtet. Die Speicherung der Daten benötigt beispielsweise für die bereits abgeschlossenen Teilprojekte des Sloan Digital Sky Survey (SDSS<sup>2</sup>, 120 Megapixel-Kamera) 150 TB, während das geplante Large Synoptic Survey Telescope (LSST<sup>3</sup>, 3.2 Gigapixel-Kamera) etwa 30 TB pro Tag produzieren wird, was bei einer Laufzeit von zehn Jahren 60 PB Speicher allein für die Rohdaten voraussetzt.

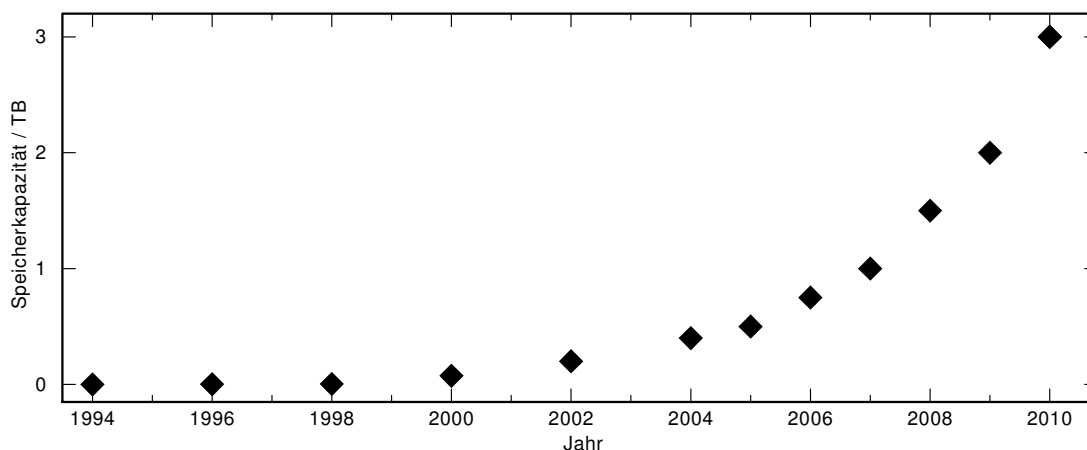


Abbildung 1.1.: Die Entwicklung der Speicherkapazität von Festplatten.

Eine zusätzliche Herausforderung entsteht durch den Datentransfer, der nötig ist, um die riesigen Datenmengen vom Teleskop zum Speicherort zu transportieren und dort zu verarbeiten. Wird zum Beispiel die Datenmenge der beiden abgeschlossenen SDSS-Projekte (150 TB) über eine Internetleitung mit einer Rate von 10 MB pro Sekunde übertragen, so benötigt dieser Transfer etwa 170 Tage.

Das geplante Square Kilometer Array (SKA<sup>4</sup>) wird ab etwa 2020 jeden Tag etwa ein Exabyte an Daten aufnehmen. Ein Transport dieser Datenmenge ist vergleichbar mit der Größe der momentan täglich weltweit über das Internet übertragenen Daten.

Auch theoretische Simulationen folgen diesem Trend der Erzeugung immer größerer Datenmengen in immer kürzerer Zeit. Bereits die Millennium-Simulation<sup>5</sup> (2005) belegte das Rechenzentrum der Max-Planck-Gesellschaft in Garching für einen Monat, benötigt 25 TB Speicher und simulierte ca. 20 Millionen Galaxien mit  $10^{10}$  Teilchen. Aktuelle Simulationen wie Millennium XXL<sup>6</sup> berücksichtigen mehr als

<sup>1</sup><http://www.tomshardware.co.uk/hard-drive-magnetic-storage-hdd,review-32264-7.html> und [http://www.jst.go.jp/EN/research/bt08\\_en.html](http://www.jst.go.jp/EN/research/bt08_en.html), abgerufen: 01.12.2012.

<sup>2</sup><http://www.sdss.org/>

<sup>3</sup><http://www.lsst.org/lsst/>

<sup>4</sup><http://www.skatelescope.org/>

<sup>5</sup><http://www.mpa-garching.mpg.de/galform/virgo/millennium/>

<sup>6</sup><http://galformod.mpa-garching.mpg.de/dev/mxxlbrowser/>

$3 \times 10^{11}$  Teilchen und benötigen (mehrere) hundert TB an Speicherplatz.

Eine klassische Auswertung der Daten, also der Transfer zum eigenen Arbeitsplatz, die Datenreduktion sowie die Speicherung am institutseigenen Rechner, ist mit diesen Datenmengen nicht mehr möglich. Bei einigen aktuellen Teleskopen wie z.B. Pan-STARRS<sup>7</sup> geschieht die Reduktion und Bearbeitung der Beobachtungen bereits vor Ort (also in der Zentrale am Standort oder am Fuß des Berges) und nur die wichtigsten Daten werden in zentralen Datenbanken gespeichert. Von dort können sie vom Nutzer heruntergeladen und direkt analysiert werden. Mit der weiterhin steigenden Datenmenge sind jedoch auch in Zukunft neue Lösungen für die Verarbeitung, den Transport und die Speicherung der Daten nötig.

Als Reaktion auf diese Situation entstanden die Ideen des Virtuellen Observatoriums (VOs). Bereits in den 1970er Jahren wurden erste Datenbanken und einheitliche Dateiformate geschaffen. Seit Anfang des Jahrtausends entstehen Organisationen, die sich im Rahmen des VOs um die Entwicklung von Standards zur einheitlichen Beschreibung und einfachen, schnellen Verarbeitung der Daten bemühen. Mittlerweile werden die meisten Daten in Datenbanken gespeichert, die mit globalen Suchfunktionen über das Internet abgefragt werden können. Services und Tools wurden und werden entwickelt, die eine einheitliche, schnelle Verarbeitung und Darstellung der Daten (meist web basiert und intuitiv zu bedienen) ermöglichen.

Im Rahmen des Virtuellen Observatoriums wurden von der Tübinger Arbeitsgruppe „Optische und UV-Astronomie“ Dienste entwickelt, die modellierte Sternspektren über das Internet zugänglich machen. Je nach Fachkenntnis beziehungsweise Zielsetzung kann der Nutzer bereits berechnete Spektren herunterladen, individuelle Spektren berechnen lassen oder auf die zugrunde liegenden Atomdaten als Input für eigene Sternatmosphärenprogramme zugreifen. In der Anfangsphase sind diese Dienste auf dem Tübinger Programm zur Modellierung von Sternatmosphären basiert, können aber von allen Sternatmosphärenprogrammen genutzt werden.

Die vorliegende Arbeit beschäftigt sich mit diesen Diensten. Sie werden aktuell vor allem für die Analyse von Sternen kurz vor ihrem Endstadium, dem so genannten Weißen Zwerg-Stadium verwendet. Die Arbeit auf diesem Gebiet ist von besonderem Interesse, da sie Rückschlüsse auf die Prozesse im Inneren eines Sterns während früherer Entwicklungsphase zulässt und so zum Verständnis und zur Überprüfung der Sternentwicklungstheorie insgesamt beiträgt. Das hierbei angewandte Verfahren ist die Spektralanalyse. Im Innern des Sterns laufen während seiner Entwicklung verschiedene nukleare Reaktionsprozesse ab. Dabei entstehen hochenergetische Photonen, die auf ihrem Weg aus dem Stern heraus durch Strahlungs- und Streuprozesse behindert werden und so beispielsweise bei der Sonne etwa  $10^7$  Jahre benötigen, um sie zu verlassen. Bei jedem dieser Prozesse erfahren sie eine Verschiebung ihrer Wellenlänge bzw. Energie. Zuletzt wechselwirken sie mit den Atomen (oder freien Elektronen) in der Atmosphäre des Sterns. Das resultierende Licht des Sterns wird von Spektrographen nach Wellenlängen zerlegt. An dieses beobachtete Spektrum wird im Rahmen einer Spektralanalyse ein Modell angepasst, um Aussagen über die Temperatur, Oberflächenschwerebeschleunigung und chemische Zusammensetzung der Sternatmosphäre machen zu können.

In diesem einleitenden Kapitel der Arbeit wird zuerst auf den Ursprung des Lichts eines Sterns und die Komponenten eines Sternspektrums eingegangen. Anschließend wird die Entwicklung von Sternen mit Anfangsmassen zwischen 0.8 und 8 Sonnenmassen ( $M_{\odot}$ ), die ihr Leben als Weiße Zwerge beenden, kurz beschrieben. Dann wird auf die Beobachtungen, die zur Analyse dieser Sterne kurz vor dem Weißen Zwerg-Stadium wichtig sind und die Modellierung dieser Spektren eingegangen. Im folgenden Abschnitt werden die Aufgaben und Ziele des VOs allgemein und der deutsche Beitrag dazu vorgestellt. Am Ende des Kapitels wird ein Überblick über den Aufbau und die Ziele dieser Arbeit gegeben.

---

<sup>7</sup><http://pan-starrs.ifa.hawaii.edu/public>

## 1.1. Spektralanalyse und Sternentwicklung

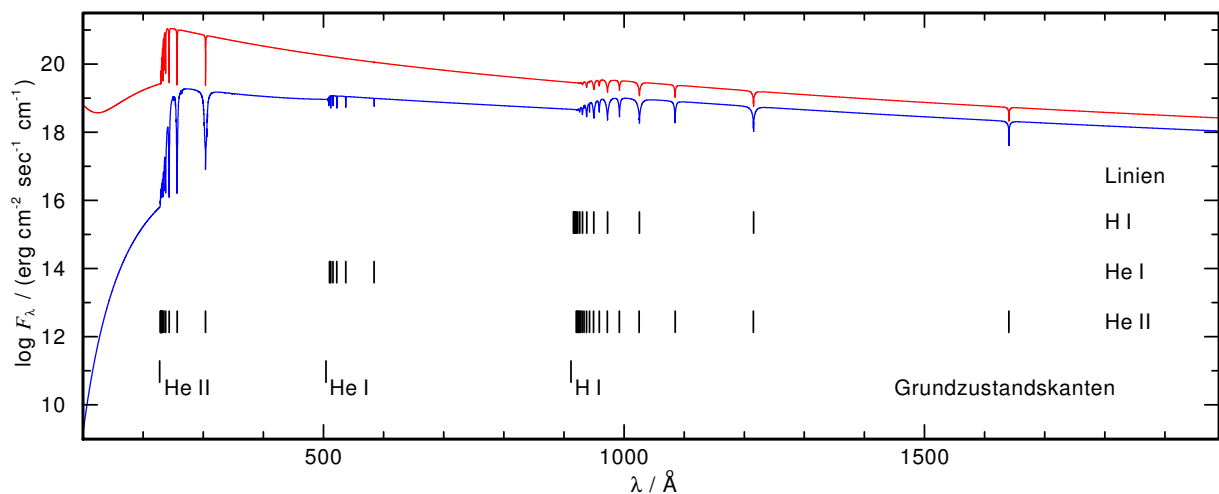


Abbildung 1.2.: Modellspektrum zweier Sternatmosphären mit  $\log g = 7.0$ , 70% Wasserstoff und 30% Helium (Massenbruchteile) und  $T_{\text{eff}} = 100\,000\text{ K}$  (rot) und  $T_{\text{eff}} = 50\,000\text{ K}$  (blau). Zusätzlich ist die Lage der einzelnen Linien und gebunden-frei-Absorptionskanten eingezeichnet.

Fast das gesamte Wissen über Sterne haben wir durch die Analyse ihres Lichts erlangt. Es entsteht in Form von hochenergetischen Photonen im Innern der Sterne bei den dort ablaufenden Kernfusionsprozessen. Aufgrund der hohen Opazität des Plasmas können die Photonen nicht direkt entweichen, sondern werden auf ihrem Weg nach außen gestreut, wodurch sie Energie verlieren. Bevor das Plasma optisch dünn wird und die Photonen den Stern verlassen können, wechselwirken sie mit den Elementen der Atmosphäre durch Absorption, Emission und Streuung.

Die Analyse dieser Photonen ist unsere wichtigste Informationsquelle, um die Zusammensetzung und den Aufbau der Sterne bzw. ihrer Atmosphäre, die Prozesse, die im Sterninneren ablaufen, und die Sternentwicklung zu verstehen. Für die Analyse stehen mehrere Methoden zur Verfügung. Beispielsweise können mit Hilfe von Lichtkurven pulsierende Sterne erforscht oder Begleitsterne entdeckt werden und die Helligkeit und Farbe (Differenz zweier Filterhelligkeiten) kann Rückschlüsse auf das Alter von Sternhaufen oder auf die Temperatur einzelner Sterne geben. Das wichtigste Verfahren zur Analyse des Lichts ist allerdings die Spektroskopie. Dafür wird das Licht eines Sterns mit Hilfe eines Prismas oder Gitters nach Wellenlängen zerlegt (Abb. 1.2). Newton (1671), der hierfür den Begriff „Spektrum“ geprägt hat, entdeckte auf diese Weise, dass sich das weiße Sonnenlicht aus einzelnen (optischen) Farben zusammensetzt. Lange Zeit wurde ausschließlich der optische Wellenlängenbereich betrachtet, bis Herschel (1800) zum ersten Mal die Infrarotstrahlung, die sich direkt an den optischen Wellenlängenbereich zu längeren Wellenlängen hin anschließt, detektieren konnte. Die UV-Strahlung, die am anderen Ende des optischen Wellenlängenbereichs die Skala fortsetzt, wurde ein Jahr später von Ritter (1801) nachgewiesen. Der analysierbare Wellenlängenbereich wurde so innerhalb kurzer Zeit deutlich erweitert. Nicht lange danach folgten weitere grundlegende Entdeckungen als von Fraunhofer (1817) erstmals Linien im optischen Spektrum der Sonne nachweisen konnte (Abb. 1.3<sup>8</sup>) und Kirchhoff (1859) zusammen mit Bunsen entdeckte, dass diese Linien eindeutig bestimmten chemischen Elementen zugeordnet werden können. Mit diesen fundamentalen Entdeckungen wurde der Grundstein zur modernen Spektralanalyse gelegt. Heute ist der gesamte Wellenlängenbereich von den Gammastrahlen bis zu den Radiowellen beobachtungstechnisch zugänglich und kann zur Analyse der Sterne verwendet werden.

<sup>8</sup><http://www.deutsches-museum.de/presse/presse-2008/tag-der-archive/#c28337>, abgerufen: 01.10.2012

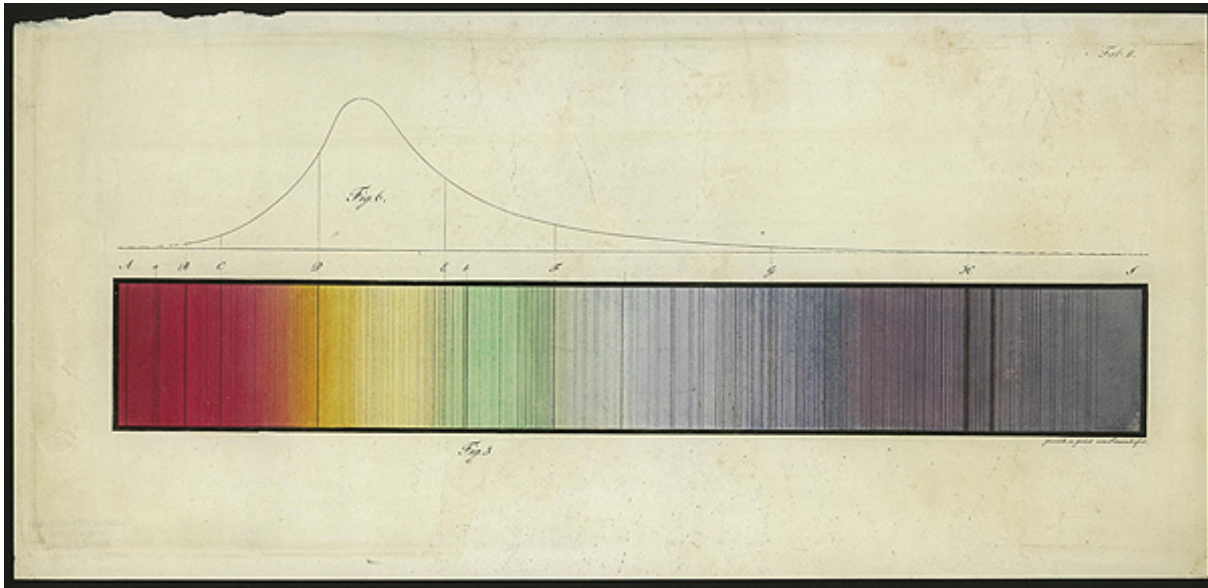


Abbildung 1.3.: Die von Joseph von Fraunhofer (1814) entdeckten Absorptionslinien im Sonnenspektrum (unten). Oben ist die Intensitätsverteilung des Sonnenspektrums dargestellt. (Foto: Deutsches Museum)

Das Spektrum eines Sterns setzt sich aus mehreren Bestandteilen zusammen. Es besteht aus einem Kontinuum, das ein von der Effektivtemperatur ( $T_{\text{eff}}$ ) abhängiges Maximum besitzt. Außerdem beinhaltet es Absorptions- und Emissionslinien, die von Elektronenübergängen zwischen den Energieniveaus der chemischen Elemente der Atmosphäre des Sterns stammen. Diese Linien haben eine, für das jeweilige Element mit dem die einzelnen Photonen zuletzt wechselwirkten, charakteristische Energie und damit Wellenlänge. Zusätzlich zu den Linien sind Absorptionskanten zu sehen, die von gebunden-frei-Übergängen (Photoionisationen, Kapitel 2.1) verursacht werden.

Für die Auswertung eines Spektrums, eine präzise Bestimmung der Sternatmosphärenparameter, ist die Modellierung der Atmosphäre und ein Vergleich des berechneten synthetischen Spektrums (spectral energy distribution, SED) mit dem beobachteten nötig. Die Parameter der Modellatmosphäre werden dabei so lange angepasst, bis Modell und Beobachtung übereinstimmen. Dadurch können Rückschlüsse auf die Parameter einer Sternatmosphäre (Effektivtemperatur  $T_{\text{eff}}$ , Oberflächenschwerebeschleunigung  $\log g$  und chemische Zusammensetzung) des Sterns gezogen werden.

Das Ziel der mit Hilfe der Spektralanalyse gewonnenen Daten ist, den Aufbau und die Entwicklung der Sterne zu verstehen und den analysierten Stern in eine theoretische Entwicklungssequenz einzuordnen bzw. seinen -status zu bestimmen. Aufgrund der großen Zeitskalen der Sternentwicklung ist eine direkte Beobachtung dieser Entwicklung nicht möglich. Aus den Momentaufnahmen vieler Sterne in unterschiedlichen Entwicklungsphasen kann jedoch eine Entwicklungstheorie abgeleitet werden. Ein wichtiges Hilfsmittel zum Verständnis der Sternentwicklung ist das Hertzsprung-Russell-Diagramm (HRD)<sup>9</sup>, Abb. 1.4). Sterne im selben Entwicklungsstadium befinden sich im HRD in bestimmten Bereichen und ihre Entwicklung kann durch „Wege“ im HRD gut beschrieben werden.

Die Entwicklung von Sternen mit mittlerer Masse (Anfangsmasse  $0.8 - 8 M_{\odot}$ ) wird nun kurz beschrieben, ausführlichere Darstellungen sind z.B. in Kippenhahn & Weigert (1990) zu finden. Die Entwicklung beginnt – wie die aller Sterne – mit dem Gravitationskollaps einer Gas- und Staubwolke mit einer Gesamtmasse, die die Jeansmasse (Jeans, 1902) übersteigt (typischerweise  $10^2 - 10^5 M_{\odot}$ ). Dabei bilden sich durch Fragmentierung viele einzelne Protosterne, die so lange weiter kontrahieren, bis Kerntempe-

<sup>9</sup>Das HRD wurde von Hertzsprung (1911) und Russell (1913) entwickelt. Im HRD wird die Leuchtkraft oder Helligkeit gegen den Spektraltyp, eine Farbe oder die Oberflächentemperatur aufgetragen.

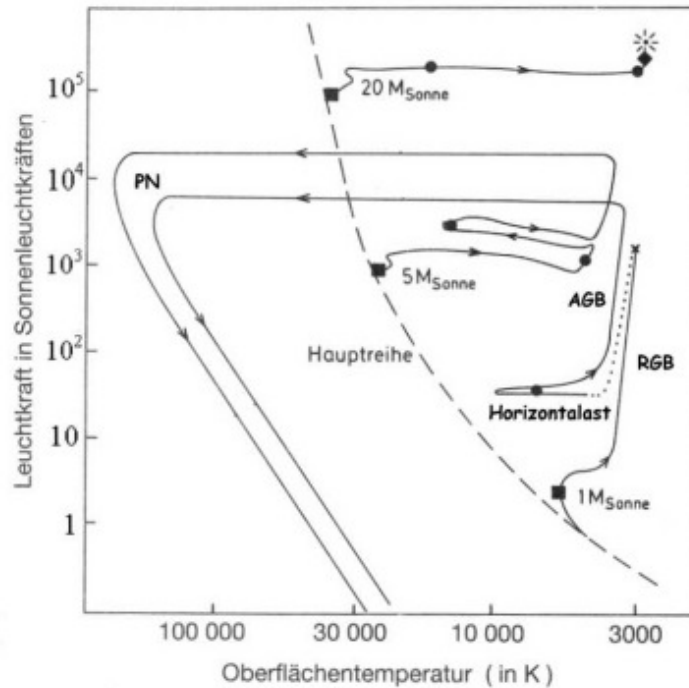


Abbildung 1.4.: Hertzsprung-Russell-Diagramm mit Entwicklungswegen von Sternen mit Anfangsmassen von  $1, 5$  und  $20 M_{\odot}$  (Langer, 1995).

ratur und -druck groß genug sind, um Wasserstoffbrennen zu zünden ( $\approx 10^7$  K). Der Stern befindet sich beim Einsetzen der Kernfusion im HRD auf der so genannten (Nullalter)Hauptreihe (Abb. 1.4).

Mit dem Fortschreiten des zentralen Wasserstoffbrennens sammelt sich immer mehr Helium im Kern des Sterns an und der Fusionsprozess wird in eine Schale verlagert. Dadurch expandiert die Hülle des Sterns während sein Kern kontrahiert. Im HRD folgt der Stern während dieser Entwicklung einem Pfad zu niedrigeren  $T_{\text{eff}}$  hin und zu höheren Leuchtkräften<sup>10</sup> in das Gebiet der Roten Riesen (red giant branch, RGB). Im Laufe dieser Entwicklung heizt sich das im Kern angesammelte Helium durch die Kontraktion immer weiter auf bis (bei  $\approx 10^8$  K) das Heliumkernbrennen zündet. Bei Sternen mit Anfangsmassen  $M < 2.5 M_{\odot}$  erfolgt dies aufgrund der quantenmechanischen Entartung ihres Kerns als Heliumflash. Im HRD wandert der Stern innerhalb kurzer Zeit auf den Horizontalast.

Durch das zentrale Heliumbrennen bildet sich mit der Zeit ein Kohlenstoff-Sauerstoff-Kern. Analog zum Wasserstoffbrennen wird dadurch auch das Heliumbrennen in eine Schale verlagert. In der folgenden Entwicklungsphase bewirkt das Heliumbrennen in der inneren Schale eine Expansion der darüber liegenden Schichten. Im HRD entwickelt sich der Stern während dieser Phase entlang des Asymptotischen Riesenastes (asymptotic giant branch, AGB). Gegen Ende dieser Phase wird die weiter außen befindliche Wasserstoffbrennschale durch die Expansion zu niedrigeren Temperaturen verschoben, bei denen der Brennprozess beinahe zum Erliegen kommt. Hierdurch steht der Heliumbrennschale immer weniger „Brennstoff“ zur Verfügung. Das Heliumbrennen wird schwächer, d.h. die Fusionsrate sinkt, wodurch der Stern kontrahiert und die Wasserstoffbrennschale wieder weiter nach innen zu höheren Temperaturen wandert. Dort kann dieser Brennprozess wieder effektiver durchgeführt werden. Nach der Bildung von genügend neuem Brennmaterial für das Heliumschalenbrennen beginnt der beschriebene Zyklus (ein so genannter thermischer Puls) von vorne. Diese thermischen Pulse ereignen sich, wenn sich der Stern an der Spitze des AGB befindet.

Während der gesamten AGB-Phase steigt die Leuchtkraft des Sterns immer weiter an und der radiativ

<sup>10</sup>  $L \propto R^2 T_{\text{eff}}^4$ , wobei  $L$  die Leuchtkraft und  $R$  den Sternradius beschreiben.



Abbildung 1.5.: Von links nach rechts: Der Katzenaugennebel, Quelle: NASA, ESA, HEIC, and The Hubble Heritage Team (STScI/AURA). Der Helixnebel, ESO. Der Stundenglasnebel, Raghvendra Sahai and John Trauger (JPL), the WFPC2 science team, and NASA/ESA. Der Ameisennebel, James Long & the ESA/ESO/NASA Photoshop FITS Liberator.

getriebene Sternwind sorgt gegen Ende der Phase dafür, dass große Teile der äußeren Hülle (bis zu 90%<sup>11</sup>) abgestoßen werden. Durch diesen Massenverlust tragen die Sterne zur chemischen Entwicklung der interstellaren Materie in den Galaxien bei. Nach dem Abstoßen der Hülle kontrahiert der Stern und heizt sich auf (der heißeste bekannte Stern in dieser Entwicklungsphase – KPD 0005+5106 – hat  $T_{\text{eff}} \approx 200\,000\text{ K}$ , Wassermann et al., 2010). Im HRD wird dies durch eine Phase konstanter Leuchtkraft und steigender Effektivtemperatur beschrieben. Ab einer Effektivtemperatur von etwa 25 000 K kann der Kern des Sterns das zuvor abgestoßene Material zum Leuchten anregen, so dass es als so genannter Planetarischer Nebel (PN, Abb. 1.5) zu beobachten ist.

Mit der Zeit gehen die Brennstoffe der Fusionsprozesse endgültig zu Neige. Der Stern hat die Abkühlsequenz der Weißen Zwerge erreicht. In diesem Stadium haben die Sterne ca. Erdgröße und typischerweise  $0.6 M_{\odot}$ . Sterne, die sich kurz vor diesem Endstadium befinden, sind heiße, kompakte Objekte. In dieser Phase sind Elemente, die durch die nuklearen Reaktionsprozesse gebildet wurden und durch Konvektion oder Diffusion während früherer Entwicklungsphasen an die Oberfläche des Sterns gelangt sind, in der Atmosphäre des Sterns detektierbar. Mit Hilfe der modernen Spektralanalyse, angewandt auf die Spektren dieser Sterne, können Rückschlüsse auf die Prozesse gezogen werden, die in den vorhergehenden Phasen im Sterninneren abgelaufen sind. Sie ist besonders wichtig für die Erstellung und Überprüfung von Sternentwicklungstheorien.

Diese Sterne haben ihr Flussmaximum aufgrund ihrer hohen Effektivtemperaturen im (extremen) UV-Bereich. Für verlässliche Analysen dieser Objekte sollten Spektren aus dem gesamten beobachtbaren Wellenlängenbereich verwendet werden, vor allem aber vom extremen ultravioletten (EUV-) bis zum optischen Bereich. Grundlage für eine Analyse dieser Objekte sind außerdem, vor allem angesichts der vielen feinen Linien der aus der Kernfusion resultierenden Elemente, hochaufgelöste Beobachtungen mit einem hohen Signal-zu-Rausch-Verhältnis. Diese Daten stammen von einer Vielzahl erdgebundener und weltraumgestützter Teleskope. Aufgrund der Undurchlässigkeit der Erdatmosphäre für elektromagnetische Strahlung bestimmter Wellenlängenbereiche (Abb. 1.6<sup>12</sup>) können optische Spektrographen auf der Erdoberfläche betrieben werden, während Röntgen-, EUV-, UV-, und IR-Instrumente außerhalb der Atmosphäre bzw. in großen Höhen arbeiten müssen.

Mittlerweile werden die mit aktuellen Instrumenten beobachteten Daten in Datenbanken gespeichert, die entweder über die Webseite des Observatoriums oder über allgemeine Datenbanken wie beispielsweise das Mikulski Archive for Space Telescopes (MAST<sup>13</sup>) zugänglich sind. Dort sind auch einige ältere Beobachtungen zu finden. Unnötige Mehrfachbeobachtungen können dadurch vermieden und Beobachtungsanträge auf sinnvolle Daten beschränkt werden. Außerdem kann der gesamte zugängliche Wellenlängenbereich der Objekte mit archivierten, öffentlich zugänglichen Daten (eventuell ergänzend zu eigenen Beobachtungen) einfacher analysiert werden als wenn für jede Beobachtung ein eigener Be-

<sup>11</sup>Die verbleibende Sternhülle hat eine Masse  $< 10^{-4} M_{\odot}$ .

<sup>12</sup>[http://de.wikipedia.org/wiki/Datei:Atmospheric\\_electromagnetic\\_opacity-de.svg](http://de.wikipedia.org/wiki/Datei:Atmospheric_electromagnetic_opacity-de.svg)

<sup>13</sup><http://archive.stsci.edu/index.html>



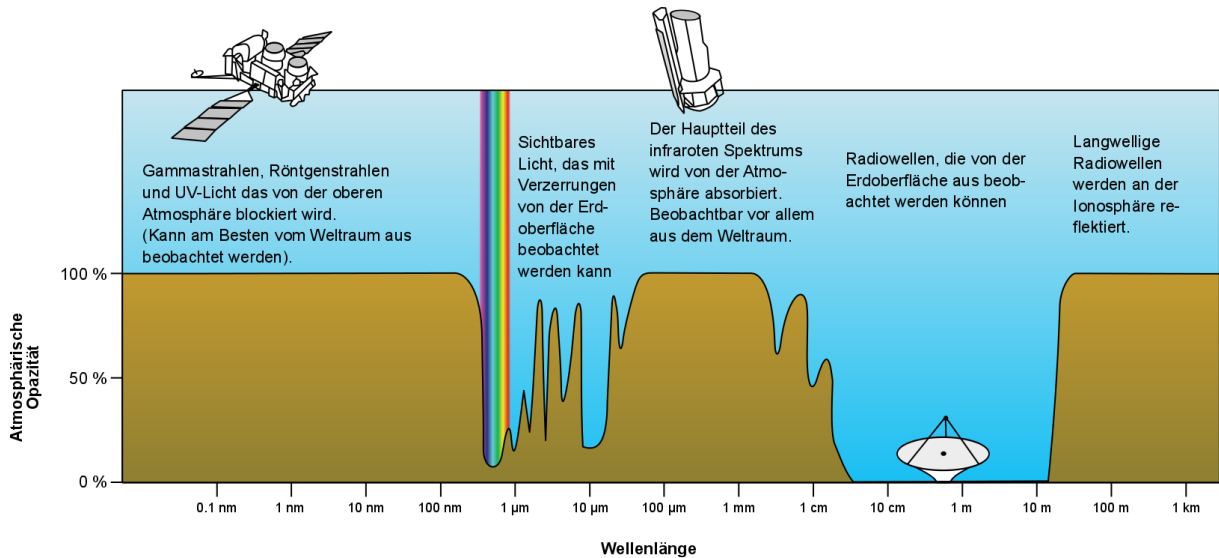


Abbildung 1.6.: Durchlässigkeit der Erdatmosphäre in den verschiedenen Wellenlängenbereichen. Quelle: NASA, modifiziert (siehe Text).

obachtungsantrag gestellt und akzeptiert werden müsste.

Für die Auswertung der Beobachtungen ist die Modellierung der Sternatmosphäre nötig. Dies ist seit Mitte des letzten Jahrhunderts möglich. Es wird dabei grundsätzlich zwischen zwei Arten von Modellatmosphären unterschieden, LTE- und NLTE-Programmen.

- **LTE:** Da die Atmosphäre eines Sterns kein abgeschlossenes System ist, herrscht kein thermodynamisches Gleichgewicht. Sind die freien Weglängen der Photonen klein, so kann ein lokales thermodynamisches Gleichgewicht (local thermodynamic equilibrium, LTE) angenommen werden. Für Atmosphären mit geringen Temperaturen und hohen Dichten (also z.B. für alte Weiße Zwerge oder Hauptreihensterne mit Spektraltyp B oder später) ist dies im Allgemeinen der Fall.
- **NLTE:** Sind die mittleren freien Weglängen der Photonen groß, gilt die LTE-Annahme nicht mehr (non-LTE oder NLTE, Abb. 1.7). NLTE-Effekte treten besonders im Bereich hoher Energien auf. Deshalb sollten für die Analyse hochaufgelöster Spektren in diesem Bereich NLTE-Modelle verwendet werden.

Zusätzlich muss beachtet werden, dass bei niedrigen  $\log g$ - und hohen  $T_{\text{eff}}$ -Werten der Strahlungsdruck den hydrostatischen Druck überwiegt und Windeffekte berücksichtigt werden müssen. Die Grenze, ab welcher der Strahlungsdruck dominiert (Eddingtonlimit) hängt von der chemischen Zusammensetzung der Atmosphäre ab.

Für die Modellierung aller planparallelen Sternatmosphären wird die Strahlungstransportgleichung

$$\mu \frac{\partial I_{\nu}(\mu)}{\partial \tau_{\nu}} = I_{\nu}(\mu) - S_{\nu}$$

zusammen mit den Gleichungen für das statistische Gleichgewicht und zusätzlichen Nebenbedingungen gelöst.  $I_{\nu}$  steht hierbei für die spezifische Intensität der Strahlung,  $\tau_{\nu}$  für die optische Tiefe,  $S_{\nu}$  für die Quellfunktion und  $\mu$  für den Kosinus des Winkels der Strahlungsrichtung zur Beobachtungsrichtung. Die formale Lösung der Strahlungstransportgleichung ist

$$J_{\nu} = \Lambda S_{\nu},$$

wobei  $J_{\nu}$  über Winkelintegration aus  $I_{\nu}$  erhalten wird und für die mittlere Intensität steht.

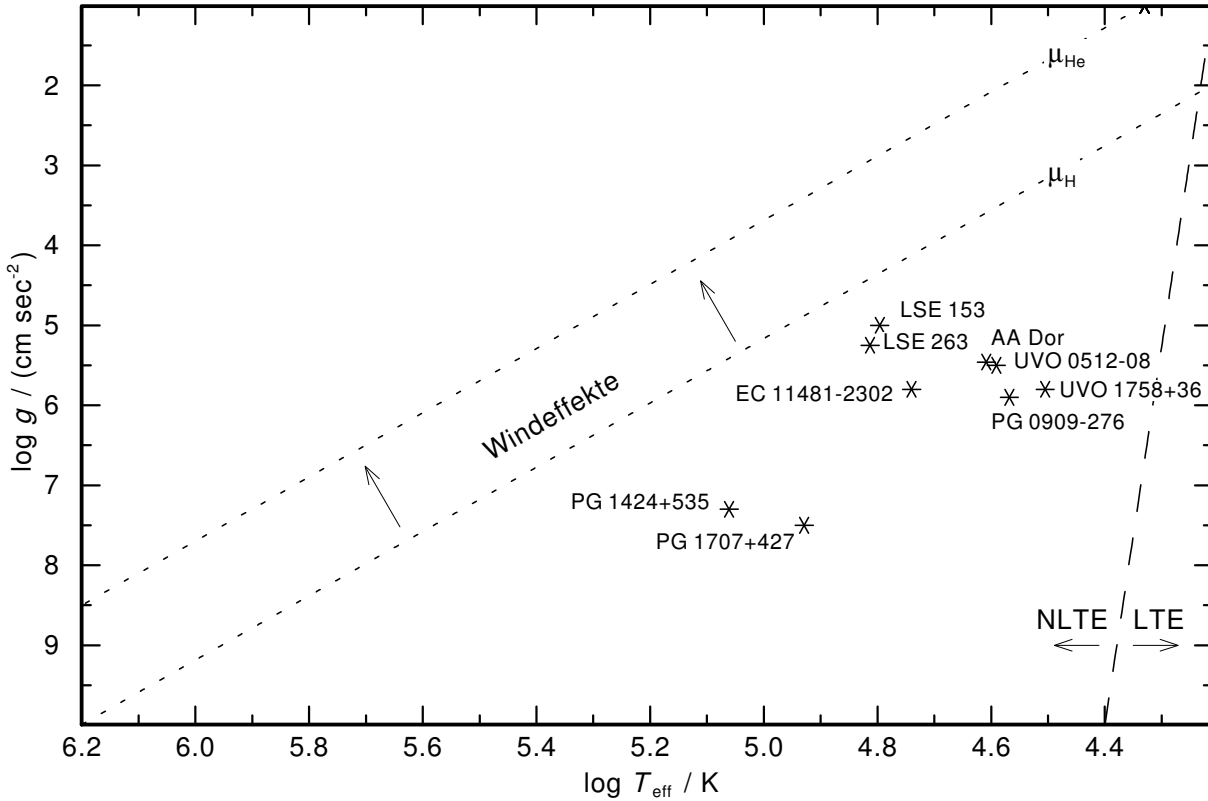


Abbildung 1.7.: Übersicht über den ungefähren Gültigkeitsbereich von LTE- und NLTE-Modellen. Das Eddingtonlimit ist für reine H- (mittleres Atomgewicht  $\mu_{\text{H}}$ ) und reine He-Atmosphären ( $\mu_{\text{He}}$ ) gestrichelt dargestellt. Die in dieser Arbeit untersuchten Objekte sind ebenfalls eingezeichnet.

Die oben erwähnten Nebenbedingungen sind Gleichungen für das radiative Gleichgewicht

$$\int_0^{\infty} \kappa_{\nu} (J_{\nu} - S_{\nu}) d\nu = 0,$$

und das hydrostatische Gleichgewicht

$$\frac{dP}{dm} = g,$$

sowie für Teilchenzahl- und Ladungserhaltung. Dabei steht  $P$  für den Druck,  $m$  für die Säulenmasse,  $g$  für die Oberflächenschwerebeschleunigung und  $\kappa_{\nu}$  für den Absorptionskoeffizienten.

Jede dieser Gleichungen gibt Aufschluss über einzelne Größen der Atmosphäre wie z.B. die Temperaturschichtung. Der Unterschied zwischen LTE- und NLTE-Modellen ist die Lösung der Strahlungstransportgleichung (im LTE-Fall gilt  $S = B(T)$ ) und die Berechnung der Besetzungszahlen der Modellniveaus sowie die im NLTE-Fall daraus resultierende Kopplung des Gleichungssystems, was das zur Lösung benötigte Programm sehr komplex macht. Während für LTE-Modelle die Saha-Boltzmann-Gleichung zur Bestimmung der Besetzungszahlen verwendet wird, werden im NLTE-Fall die Ratengleichungen

$$n_i \sum_{i \neq j} P_{ij} - \sum_{j \neq i} n_j P_{ji} = 0$$

für die statistischen Gleichungen herangezogen, wobei  $n_i$  für die Besetzungszahl des Niveaus  $i$  steht und  $P_{ij}$  die Ratenkoeffizienten für die Übergänge zwischen den atomaren Niveaus  $i$  und  $j$  darstellen. Im Gegensatz zum LTE-Fall hängen die Besetzungszahlen im NLTE zusätzlich noch vom Strahlungsfeld

ab. Für NLTE-Modelle entsteht so ein gekoppeltes, hochgradig nicht-lineares Gleichungssystem, dessen Lösung deutlich aufwändiger und zeitintensiver als die der LTE-Gleichungen ist.

Das erste NLTE-Modell, für dessen Berechnung dieses Gleichungssystem vollständig linearisiert wurde, wurde von Auer & Mihalas (1969) gerechnet. Im Vergleich zu LTE-Modellen konnten aber weniger Elemente (zuerst nur H, dann H und He) und (bis zu 18) Levels berücksichtigt werden. Erst mit der Einführung weiterer Lösungsverfahren wie des Accelerated Lambda Iteration (ALI)-Verfahrens (Canon, 1973; Scharmer, 1981; Werner & Husfeld, 1985), wurde die Berücksichtigung weiterer Elemente und insgesamt mehr Niveaus möglich. Damit konnten erstmals die Effekte leichter Metalle<sup>14</sup> auf die Atmosphäre untersucht werden (z.B. Werner, 1988). Mittlerweile können mehr als 2000 NLTE-Levels in der Berechnung berücksichtigt und der Einfluss weiterer Elemente auf die Atmosphäre getestet werden. Bei der Untersuchung der Effekte der Eisengruppenelemente (Ca, Sc, Ti, V, Cr, Mn, Fe, Co und Ni) kann dabei allerdings nicht wie bei den anderen Elementen verfahren werden. Grund dafür ist, dass sie eine Elektronenkonfiguration analog zu Argon haben, wobei zusätzlich die 3d und 4s Schalen (zum Teil) gefüllt sind. Die jeweiligen Elektronenbahnen überlagern sich und es entsteht eine Bandstruktur, die zu einer Vielzahl von Levels mit sehr ähnlicher Energie und Millionen von zugehörigen Übergängen führt. Heute ist es nicht möglich, diese Anzahl an NLTE-Levels in der Modellatmosphärenberechnung zu berücksichtigen und sie können deshalb nicht analog zu den klassischen Modellatomen behandelt werden. Um sie trotzdem in NLTE-Rechnungen zu berücksichtigen, wurde ein statistisches Verfahren eingeführt (Anderson, 1985), mit dem die Zahl der NLTE-Levels und die der zugehörigen Übergänge deutlich reduziert werden (Kapitel 2.1). Dieses Verfahren ermöglichte die Untersuchung der Auswirkungen der Eisengruppenelemente auf eine Sternatmosphäre – die so genannten Line-Blanketing-Effekte (z.B. Dreizler & Werner, 1993; Haas et al., 1996). Sie wirken vor allem auf drei Bereiche der Atmosphäre.

- Durch die zusätzlichen Linien, die sich – je nach Effektivtemperatur des Sterns und somit dominanter Ionisationsstufen der Elemente – hauptsächlich im Röntgen-, EUV- oder UV-Bereich befinden, wird der Fluss, der in diesem Bereich aus der Atmosphäre austritt, gemindert (flux blocking).
- Da der Gesamtfluss erhalten bleiben muss, erwärmen sich die tieferen Schichten (back warming).
- Durch Resonanzlinien, die sich weit in die äußere Atmosphäre erstrecken, wird diese gekühlt (die emittierten Photonen können den Stern verlassen und nehmen ihre Energie mit – surface cooling).

## 1.2. Das Virtuelle Observatorium

In den letzten Jahren bzw. Jahrzehnten ist eine Entwicklung der Astronomie in eine neue Richtung zu beobachten. Der Fortschritt im Bereich elektronischer Detektoren und Speichermedien mit immer größerer Kapazität und Effizienz führt seit einiger Zeit zu einer jährlichen Verdopplung der astronomischen Datenmengen. Dieser Trend wird zum einen durch die steigende Zahl von Teleskopen und deren Instrumente, durch die höhere räumliche, spektrale und zeitliche Auflösung der Beobachtungen (und damit Größe der Beobachtungsdateien), durch den Schritt von einzelnen Beobachtungen zu Multi-Objekt-Spektroskopie und durch die Einführung von Survey-Teleskopen unterstützt bzw. verstärkt. Anstelle von wenigen Beobachtungen werden so einige tausend pro Nacht aufgenommen, die jeweils mehr Speicherplatz benötigen als vor einigen Jahren die Beobachtungsdaten einer ganzen Nacht und durch die kontinuierliche Beobachtung des ganzen Himmels sind spezielle Datenbanken erforderlich, zu denen der notwendige Datentransfer gewährleistet sein muss.

Dies wird sich auch in Zukunft weiter fortsetzen. Dabei bringen uns schon bereits abgeschlossene und aktuelle Projekte an die Grenzen unserer Möglichkeiten, die Daten zeitnah auszuwerten. Zum

<sup>14</sup>In der Astronomie wird grundsätzlich zwischen Wasserstoff (X), Helium (Y) und Metallen (Z) unterschieden. Weitere Unterscheidungen sind dann leichte Metalle (Kohlenstoff, Stickstoff und Sauerstoff) und schwere Metalle (die Eisengruppe).

Beispiel benötigen die Teilprojekte des Sloan Digital Sky Survey SDSS-I (2000–2005) und SDSS-II (2005–2008) ca. 150 TB Speicher und umfassen ca. 200 Millionen Objekte. Die aktuelle Veröffentlichung des SDSS Data Release 9 umfasst ca. eine Milliarde Objekte. Das 2010 in Betrieb genommene Panoramic Survey Telescope & Rapid Response System (Pan-STARRS) wird im Vollbetrieb einige TB pro Nacht produzieren, wovon allerdings ein Teil von einer Software direkt aussortiert und gelöscht wird. Dabei darf die Software nur eine Minute pro Aufnahme benötigen, um alle Aufnahmen bearbeiten zu können. Das geplante Radioteleskop Square Kilometer Array (SKA), dessen Antennen in Australien und Südafrika aufgestellt werden, wird ab ca. 2020 jeden Tag etwa ein Exabyte an Daten produzieren. Der Transfer dieser Datenmenge übersteigt das aktuelle, tägliche Datentransfervolumen über das Internet in Westeuropa<sup>15</sup> etwa um das dreifache. Aktuell wird in einem Gemeinschaftsprojekt von IBM und dem niederländischen Institut für Radioastronomie (ASTRON<sup>16</sup>) erforscht, wie diese Datenmenge transportiert, verarbeitet und gespeichert werden kann<sup>17</sup>.

Dieser Trend wird auch durch Projekte für theoretische Daten verstärkt. Die Millennium-Simulation<sup>18</sup> (2005,  $10^{10}$  Teilchen, 1.4 Millionen CPU-Stunden, 25 TB Speicherplatz, siehe Kapitel 1) ist bereits wieder übertroffen worden. Millennium XXL<sup>19</sup>, eine 2010 durchgeführte Folgesimulation des Millennium-Runs berücksichtigte  $3 \times 10^{11}$  Teilchen. Sie wurde am Jülich Supercomputing Center auf 12 888 Kernen realisiert und benötigte etwa 2.86 Millionen CPU-Stunden, weshalb die Simulation innerhalb von etwa zehn Tagen durchgeführt werden konnte (Angulo et al., 2012). Die gesamten resultierenden Datenprodukte benötigen 100 TB Speicher.

Insgesamt ist durch die jüngste und zukünftige Produktion astronomischer Daten eine Umstrukturierung der Arbeitsweise zwingend erforderlich. Die Art der Analyse muss sich ändern. Viele Prozesse und Verfahren wie z.B. die Reduktion von beobachteten Daten, eine erste Klassifizierung oder Analyse müssen automatisiert werden. Die Daten dürfen nicht mehr verloren gehen, sondern müssen jedem zugänglich sein. Analysewerkzeuge sollten intuitiv zu bedienen sein und bereit stehen, sobald die Daten produziert werden. Um das realisieren zu können, ist es notwendig, bereits existierende Programme und Verfahren zu verwenden und gegebenenfalls zu erweitern anstatt diese für jedes Beobachtungsprogramm neu zu erfinden. Dies erhöht auch die Kompatibilität mit bereits existierenden Datenbanken oder Programmen zur Analyse der Daten.

Als Reaktion auf diese Entwicklung wurde das Virtuelle Observatorium (VO) eingeführt. Aufgabe des VO ist die Umsetzung der eben beschriebenen Ideen. Das VO hat es sich zum Ziel gemacht, Datenverlust zu verhindern, astronomische Daten weltweit zugänglich zu machen und den Umgang mit diesen zu erleichtern. Dafür wurden Standards entwickelt, die eine international einheitliche Beschreibung und Handhabung der Daten gewährleisten. Durch die Einführung von Metadaten<sup>20</sup> können z.B. Datenbanken eingerichtet werden, die mit Hilfe eines globalen Verfahrens durchsuchbar sind. Astronomen ist es möglich, eigene Datenbanken anzulegen, die international in größere Datenbanken oder Suchmaschinen integrierbar sind. Mit der Bereitstellung von Speicherplatz durch VO-Organisationen konnten auch einige Datenbanken übernommen werden. Speicherung möglichst vieler Daten ist jedoch nicht das eigentliche Ziel des VO. Ein einheitlicher Umgang mit den Daten, die Vermeidung von unnötigen Doppelbeobachtungen und Doppelentwicklungen von Programmen sowie ein freier Zugang zu diesen Programmen spielt eine noch größere Rolle. Analysewerkzeuge und Dienste wie Programme zur Darstellung und Weiterverarbeitung oder Suchmaschinen wurden und werden deshalb mit den neu definierten Standards entwickelt. Diese Werkzeuge sowie die Datenarchive stehen jedem Astronomen über das Internet zur Verfügung. Zusätzlich bietet das VO eine Anlaufstelle für Astronomen, die Hilfe bei der Verwendung

<sup>15</sup>z.B. [http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white\\_paper\\_c11-481360.pdf](http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-481360.pdf), abgerufen: 02.12.2012

<sup>16</sup><http://www.astron.nl/>

<sup>17</sup><http://www-03.ibm.com/press/us/en/pressrelease/37361.wss>, abgerufen: 2.12.2012

<sup>18</sup><http://www.mpa-garching.mpg.de/galform/virgo/millennium/>

<sup>19</sup><http://galformod.mpa-garching.mpg.de/dev/mxxlbrowser/>

<sup>20</sup>Metadaten enthalten standardisierte Informationen über den jeweiligen astronomischen Datensatz, wie z.B. das Beobachtungsinstrument, den Objektnamen oder den Wellenlängenbereich.

oder Sicherung der Daten und Werkzeuge benötigen oder die mehr über einen Dienst erfahren wollen.

Zukünftige *VO*-Aktivitäten müssen insbesondere darauf abzielen, dass die Software einsatzfähig ist, sobald die Teleskope den Betrieb starten. Die Verarbeitung der Daten muss dann ebenfalls weitgehend automatisch funktionieren und sollte möglichst am Speicherort der Daten geschehen. Aufgrund der Größe der Datenmengen muss außerdem beachtet werden, dass nur sinnvoll eingeschränkte Daten gesichert werden.

Die Auswirkungen der *VO*-Aktivitäten haben die Arbeit heutiger Astronomen bereits deutlich verändert. Früher musste zur Analyse eines Objekts zuerst ein Beobachtungsantrag gestellt werden, fundiert mit Vorhersagen aus theoretischen Rechnungen um die Wichtigkeit und Erreichbarkeit der Ziele des Antrags zu untermauern. Dann ist der Astronom selbst zum Teleskop gereist, hat mit Hilfe von Aufsucherkarten seine Objekte beobachtet und die Daten anschließend mit nach Hause genommen. Wieder am Arbeitsplatz angekommen, konnte der Astronom mit der endgültigen Datenreduktion beginnen. Neue theoretische Modelle wurden berechnet und die Analyse durchgeführt. Die Daten waren im Besitz des Beobachters und ansonsten meistens niemandem zugänglich. Aufgrund der geringen Lebensdauer der damaligen Speichermedien (z.B. Magnetbänder), waren die Daten unter Umständen innerhalb weniger Jahre verloren.

Heute wird zuerst in den Datenbanken geprüft, welche Spektren zum gewünschten Objekt verfügbar sind. Oft können bereits automatisch reduzierte direkt heruntergeladen und mit theoretischen Modellen verglichen werden. Die beobachteten Daten sind permanent und (in der Regel nach einem Jahr) jedem zugänglich. Falls doch ein eigener Beobachtungsantrag notwendig sein sollte, muss der lange Weg zum Teleskop meist nicht unternommen werden, da die Beobachtung oft vom Teleskoppersonal („service mode“) ausgeführt wird.

Auch auf Seite der theoretischen Daten hat sich die Arbeit deutlich gewandelt. Früher galt beispielsweise die NLTE-Spektralanalyse als Spezialgebiet der Astronomie. Viele Astronomen hatten keinen Zugang zu einem Modellatmosphärencode oder haben sich nicht an die Einarbeitung gewagt. Die einzige Möglichkeit ohne Sternatmosphärencode an ein Modell zu kommen war, einen Experten zu bitten es für ihn zu berechnen. Mittlerweile sind viele theoretische Datensätze immer höherer Qualität und größerer Auswahl an berücksichtigten Elementen über das Internet frei zugänglich. Sie können heruntergeladen und für Analysen unterschiedlicher Objekte verwendet werden. Dadurch wird eine Klassifizierung von Spektren und eine erste schnelle Voranalyse mit wenig Aufwand ermöglicht und die Rechenzeit kann für Detailanalysen genutzt werden.

Die *VO*-Organisationen wurden erst einige Jahre nach der Entstehung des *VO*-Konzepts geschaffen. Die eben beschriebene Veränderung der Arbeit eines Astronomen aufgrund von *VO*-Aktivitäten war unter anderem deshalb so erfolgreich, weil die Ideen von allen, nicht nur *VO*-Angestellten, umgesetzt wurden und werden und weil damit bereits lange vor diesem letzten „Jahrzehnt des *VOs*“ – in dem die Realisierung der Aufgaben deutlich beschleunigt wurde – begonnen wurde. Bereits in den 1970er Jahren gab es die ersten Datenbanken. Wichtige Beispiele für *VO*-Verwirklichungen vor dem Jahr 2000 sind die SIMBAD<sup>21</sup>-Datenbank (1979), ALADIN<sup>22</sup> (1999) oder das Reduktions- und Analysesoftware-Paket IRAF<sup>23</sup> (1984). Auch Datenformate wie z.B. das weit verbreitete FITS-Format<sup>24</sup> (1981) wurden entwickelt.

Die ersten *VO*-Organisationen mit bezahlten Mitarbeitern, die sich hauptberuflich darum kümmern, die Ideen des *VO* zu verwirklichen, wurden Anfang des 21. Jahrhunderts gegründet. Zuerst gab es einige nationale *VOs*, die sich dann zu größeren Gemeinschaften wie dem European Virtual Observatory (EURO-*VO*<sup>25</sup>, als Nachfolger des Astrophysical Virtual Observatory, AVO, 2001, Titelbild), dem US-

---

<sup>21</sup><http://simbad.u-strasbg.fr/simbad/>

<sup>22</sup><http://aladin.u-strasbg.fr/aladin.gml>

<sup>23</sup><http://iraf.noao.edu/>

<sup>24</sup><http://fits.gsfc.nasa.gov/>

<sup>25</sup><http://www.euro-vo.org/pub/>



Abbildung 1.8.: Mitgliedsorganisationen der IVOA.

amerikanischen National Virtual Observatory (NVO<sup>26</sup>, 2001) oder der International Virtual Observatory Alliance (IVOA<sup>27</sup>, 2002) zusammengeschlossen haben. Im Laufe der Zeit haben immer mehr Länder eigene VOs eingerichtet, die ebenfalls den internationalen Vereinigungen beigetreten sind. 2012 hatte beispielsweise die IVOA 19 Mitgliedsorganisationen (Abb. 1.8).

Heutzutage können alle Werkzeuge und Hilfsmittel für astronomische Daten, die nach VO-Standards entwickelt wurden, von jeder Person auf jedem Kontinent registriert und somit veröffentlicht werden. Diese Hilfsmittel können dann über diese Registries (wie z.B. das des NVO<sup>28</sup>) gefunden und verwendet werden. Durch die Einrichtung dieser Registries kann jeder – nicht nur bezahlte Mitarbeiter von VO-Organisationen – zum VO beitragen, was wichtig ist, um mit der steigenden Datenmenge mithalten zu können.

Das German Astrophysical Virtual Observatory (GAVO) ist der deutsche Beitrag zum VO. Es wurde 2002 gegründet und ist Teil der IVOA. Die vier GAVO-Projekte, die es bisher gab, sind vom Bundesministerium für Bildung und Forschung gefördert worden:

- GAVO I wurde von 2002 bis 2005 verwirklicht.
- Die Realisierung des GAVO II-Projekts erfolgte von 2006–2008. In diesem Projekt gab es zum ersten Mal eine Tübinger Beteiligung, wobei ein Dienst, der SEDs über das Internet bereitstellt, entwickelt wurde (Theoretical Stellar Spectra Access, *TheoSSA*<sup>29</sup>).
- Das GAVO III-Projekt wurde von 2008–2011 durchgeführt.
- Momentan wird das GAVO-2011–2014-Projekt umgesetzt. Hierbei werden von Tübinger Seite aus Werkzeuge zur Visualisierung der Spektren aus der *TheoSSA*-Datenbank und der Beobachtung (*JANALYSIS*), für einen  $\chi^2$ -Fit der theoretischen Spektren an die Beobachtung (*TCHI*) sowie für

<sup>26</sup><http://www.us-vo.org/>

<sup>27</sup><http://www.ivoa.net/>

<sup>28</sup><http://nvo.stsci.edu/vor10/index.aspx>

<sup>29</sup><http://dc.g-vo.org/theossa>

eine automatische Klassifizierung von Spektren aus Multi-Objekt-Beobachtungen (*TMOS*) entwickelt.

Ein Teil dieser Arbeit wird im Rahmen eines DFG-Projekts (Projektmittel WE 1312/41-1) parallel zu *GAVO III* und *GAVO-2011–2014* ausgeführt. Momentan sind fünf Institute direkt an *GAVO-2011–2014* beteiligt, das Max-Planck-Institut für extraterrestrische Physik (MPE, Garching), das Institut für Informatik der Technischen Universität München (TUM), das Leibniz-Institut für Astrophysik Potsdam (AIP), das Institut für Astronomie und Astrophysik Tübingen (IAAT) und das Zentrum für Astronomie Heidelberg (ZAH). Etwa 15 Mitarbeiter beschäftigen sich mit mehreren Projekten wie der Einführung und Erweiterung von *VO*-Standards, -Tools und -Protokollen, der Erweiterung des *GAVO*-Datenzentrums, des Transfers von Daten in das *VO*-Format und an Diensten, die eine Spektralanalyse über das Internet ermöglichen.

### 1.3. Aufbau und Ziele der Arbeit

Die vorliegende Arbeit beschäftigt sich mit *GAVO*-Diensten, die zur Spektralanalyse heißer, kompakter Objekte verwendet werden können und aktuell auf in Tübingen entwickelten Programmen zur Analyse solcher Objekte basieren. Im Rahmen dieser Arbeit wurden sowohl bestehende Dienste erweitert, als auch neue Dienste entwickelt, die eine präzisere Spektralanalyse mit *VO*-Mitteln bzw. -Werkzeugen und ohne Kenntnis der (Modellatmosphären-)Programme ermöglichen. Dabei ist anzumerken, dass die Dienste in Zusammenarbeit mit den *GAVO*- und *AstroGrid-D*-Teams entstanden sind und auch während dieser Arbeit mehrere Mitarbeiter an den Diensten gearbeitet haben.

Die Dienste bzw. ihre zugrunde liegenden Programme wurden für mehrere Vertreter unterschiedlicher Spektralklassen von heißen, kompakten Objekten getestet und ihr Einfluss auf das Ergebnis der Spektralanalyse bzw. ihre Bedeutung für eine Spektralanalyse mit *VO*-Diensten geprüft. Die Arbeit ist deshalb folgendermaßen aufgebaut.

1. Nach dieser Einleitung wird das bestehende *IrOnIc*-Programm (Rauch & Deetjen, 2003) zur Berechnung von „gesampten“ radiativen gebunden-gebunden- und gebunden-frei-Querschnitten und Erzeugung von Modellatomen für Eisengruppenelemente beschrieben (Kapitel 2). Dieses Programm wird unabhängig von der Modellierung der Sternatmosphäre in einem separaten Schritt vor deren Berechnung ausgeführt und kann deshalb für Sternatmosphärenprogramme unterschiedlicher Art verwendet werden. Die Berücksichtigung von Eisengruppenelementen bei der Modellierung von Sternatmosphären kann starke Auswirkungen auf die Ergebnisse der Analyse haben und die Bereitstellung der dafür notwendigen Daten ist wichtig, um eine realistische Spektralanalyse mit *VO*-Diensten zu ermöglichen. Das *IrOnIc*-Programm bzw. die daraus resultierenden Querschnittsdaten und Modellatome werden deshalb im Rahmen dieser Arbeit öffentlich zugänglich gemacht. Die Berechnung der Querschnitte kann – je nach Feinheit des verwendeten Frequenzgitters und gewählten Ionen – mehrere Tage dauern. Es werden deshalb einige Tests durchgeführt, mit denen standardmäßig verwendete Input-Daten und Berechnungsverfahren untersucht werden, um eine mögliche Beschleunigung und Verbesserung des Programms zu prüfen. Anschließend wird ein neuer *VO*-Dienst (genannt *TIRO*) entwickelt, der die mit *IrOnIc* erstellten Querschnitte und Modellatome zur Verfügung stellt. Dieser Dienst und seine Funktionsweise werden dokumentiert und alle zugehörigen Prozeduren sind im Anhang gelistet.
2. Die bereits bestehenden *VO*-Dienste, die von der Tübinger Arbeitsgruppe entwickelt wurden, werden in Kapitel 3 beschrieben. Momentan basieren sie auf dem Tübingen NLTE Model-Atmosphäre Package (*TMAP*<sup>30</sup>). Der daraus entstandene *TheoSSA*<sup>31</sup>-Dienst ermöglicht dem *VO*-Nutzer anhand von bereits berechneten SEDs eine Klassifizierung oder erste Analyse von beobachteten Sternspektren bzw. liefert einen verlässlichen Input für Photoionisationsmodelle. Für genauere Analysen kann der *TMAW*<sup>32</sup>-Dienst zur automatischen Berechnung synthetischer Spektren verwendet werden. In dieser Arbeit wird *TMAW* im Rahmen der Einführung einer Qualitätskontrolle für die Tübinger *VO*-Dienste weiterentwickelt und getestet. Die Funktionsweise des Dienstes sowie die zugehörigen Programme und Prozeduren werden dokumentiert. Die Tübinger Modellatomdatenbank (*TMAD*<sup>33</sup>), die dem *VO*-Nutzer aktuelle und detaillierte Modellatome für eine präzise Spektralanalyse zur Verfügung stellt, wird ebenfalls ausgebaut. Diese Änderungen und Erweiterungen werden auch in Kapitel 3 beschrieben.
3. Für eine Anwendung der eben beschriebenen Dienste in der Spektralanalyse (Kapitel 4) werden einzelne Objekte ausgesucht, die zu unterschiedlichen Spektralklassen gehören und verschiedene

<sup>30</sup><http://astro.uni-tuebingen.de/~TMAP>

<sup>31</sup><http://dc.g-vo.org/theossa>

<sup>32</sup><http://astro.uni-tuebingen.de/~TMAW>

<sup>33</sup><http://astro.uni-tuebingen.de/~TMAD>



$T_{\text{eff}}$ ,  $\log g$  und chemische Zusammensetzungen haben. Diese Sterne haben sich nicht nach dem oben beschriebenen Entwicklungsszenario entwickelt, sondern sind nach der RGB- oder AGB-Phase von diesem abgewichen. Ihre Atmosphären sind deshalb meist wasserstoffarm und die Ergebnisse ihrer Analyse sind besonders interessant für die Prüfung und Weiterentwicklung neuer Entwicklungstheorien und um Aufschlüsse über den Aufbau und die Prozesse in allen Sternen zu erhalten.

- Zu den ausgewählten Objekten gehören heiße unterleuchtkräftige Sterne (subdwarf-Sterne, sdB, sdOB und sdO abgekürzt, Kapitel 4.1 und 4.3), die im Kern Helium verbrennen und sich (meist) vom Stadium der Roten Riesen (ohne die AGB-Phase) direkt in Richtung der Weißen Zwerge entwickeln. Für einige der ausgewählten sd(O)B-Sterne wurde ein stark abgeflachter Flussverlauf im UV-Bereich beobachtet, für dessen Erklärung stark übersolare Eisengruppenhäufigkeiten vermutet werden.
- Objekte aus der Klasse der PG 1159-Sterne werden analysiert (Kapitel 4.2), deren äußere Wasserstoffhülle durch einen späten thermischen Puls in tiefere Schichten gemischt und dort (teilweise) verbrannt wurde. Diese Objekte bieten direkten Einblick in die zuvor unzugängliche Zwischenschale. Sie entwickeln sich weiter zu DO-Weißen Zwergen, deren Spektren von Heliumlinien dominiert sind.
- Ein Vertreter der DO-Weißen Zwerge, bei dem eine hohe Eisengruppenhäufigkeit vermutet wurde, wurde ebenfalls für die Analyse des Einflusses der Eisengruppenelemente auf den Flussverlauf dieses Objekts gewählt (Kapitel 4.4).

Die sdO- und PG 1159-Objekte werden zuerst mit *TMAP* analysiert und die resultierenden Modellspektren anschließend mit dem *TMAW*-Dienst nachgerechnet und verglichen, um die Genauigkeit der *TMAW*-Rechnungen abzuschätzen. Für die Anwendung des neuen *TIRO*-Dienstes werden in die Modelle dieser Objekte Eisengruppenelemente eingefügt, um ihre Auswirkungen auf die Atmosphäre und das Ergebnis der Analyse abschätzen zu können.

Die sd(O)B-Objekte und der DO-Weiße Zwerg zeigen ungewöhnlich stark abgesenkte Flussverläufe, deren Ursache bisher nicht erklärt werden konnte. Die Berücksichtigung von Eisengruppenelementen konnte bisher nicht zur Lösung dieser Diskrepanz beitragen, aufgrund einer Erweiterung der Atomdaten muss der Einfluss der Eisengruppenelemente jedoch neu untersucht werden.

Für den DO-Weißen Zwerg zeigt sich, dass die Eisengruppenelemente nicht die alleinige Ursache für den ungewöhnlichen Flussverlauf sein können. Als weitere mögliche Erklärung für den beobachteten Flussverlauf wird der Einfluss des interstellaren Mediums auf das beobachtete Spektrum untersucht. Dafür wird ein neues Programm zur Modellierung interstellarer Absorption entwickelt. Basierend auf diesem Programm wird ein *VO*-Dienst erzeugt, der eine Korrektur durch interstellare Absorption im EUV-Bereich an ein vom Nutzer hochgeladenes Spektrum anbringt.



## 2. Querschnitte und Modellatome für Eisengruppenelemente über das VO

Seit der Entwicklung des ersten NLTE-Modellatmosphärencodes Ende der 1960er Jahre hat sich die NLTE-Spektralanalyse deutlich gewandelt. Durch neue numerische Verfahren konnten immer mehr Elemente in die Sternatmosphärenrechnungen einbezogen werden. Dabei stellte sich zuerst heraus, dass der Einfluss der leichten Metalle auf die Sternatmosphäre nicht zu vernachlässigen ist. Mit der Einführung eines statistischen Verfahrens können seit 1992 auch Eisengruppenelemente (Ca, Sc, Ti, V, Cr, Mn, Fe, Co und Ni) berücksichtigt werden. Die Aufnahme von Eisengruppenelementen in Modellatmosphärenrechnungen kann ebenfalls starke Auswirkungen auf die Atmosphärenstruktur haben. Sie können die bereits in der Einleitung beschriebenen Line-Blanketing-Effekte verursachen. Je nach Parameter der Sternatmosphäre kann das Ergebnis einer Analyse bei Vernachlässigung der Eisengruppenelemente z.B. für  $T_{\text{eff}}$  um bis zu 12% abweichen (siehe auch Kapitel 4).

Die Linien der Eisengruppenelemente befinden sich (für die Ionisationsstufen IV-VIII) hauptsächlich im (E)UV-Bereich (Abb. 2.1). Die Berücksichtigung dieser Linien kann zu einer starken Absenkung des Flusses bei bestimmten (E)UV-Wellenlängen führen. Bei der Analyse heißer, kompakter Objekte, die ihr Flussmaximum in diesem Bereich haben, ist es deshalb wichtig, die Eisengruppenelemente zu berücksichtigen. Besonders nützlich sind diese Linien zur Temperaturbestimmung, da im Vergleich zu anderen Elementen meist mehr Ionisationsstufen zur Verfügung stehen und diese sensibel auf Temperaturänderung reagieren (Abb. 2.2). Aber selbst wenn nur optische Spektren betrachtet werden, in denen aufgrund der hohen Ionisationen keine Linien der Eisengruppenelemente zu sehen sind, hat die veränderte Atmosphärenstruktur Auswirkungen auf die Linienprofile anderer Elemente und somit auf die gesamte Analyse.

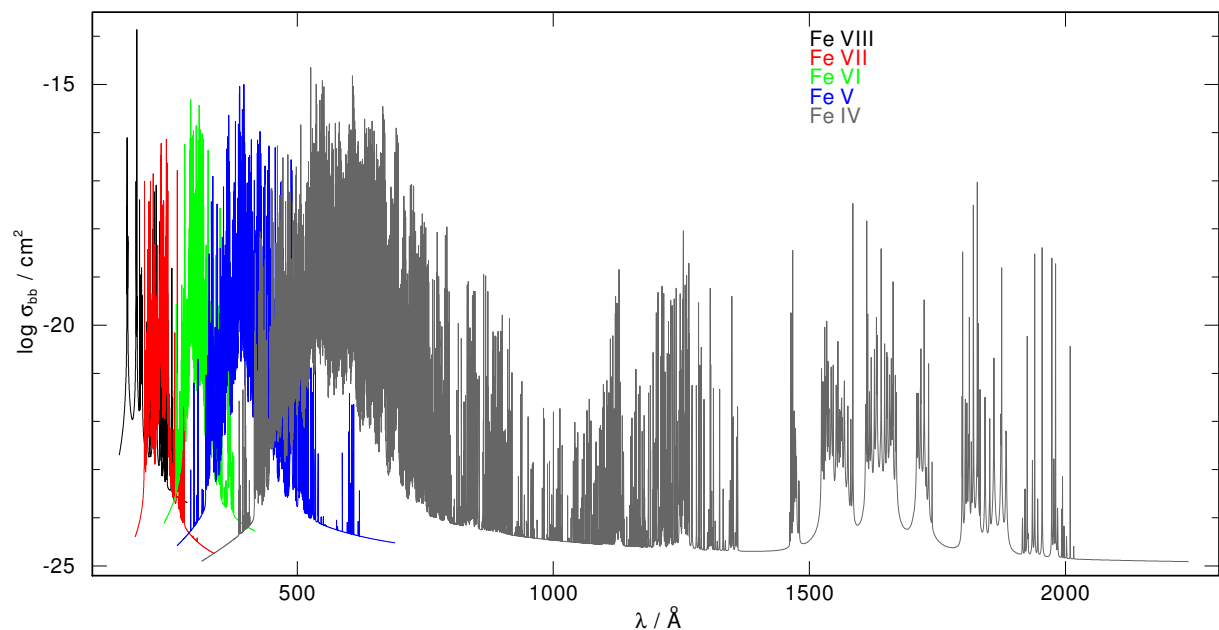


Abbildung 2.1.: Die Lage von radiativen gebunden-gebunden-Querschnitten von verschiedenen Fe-Ionen vom „Grundzustands“-Superlevel zum nächst höheren.

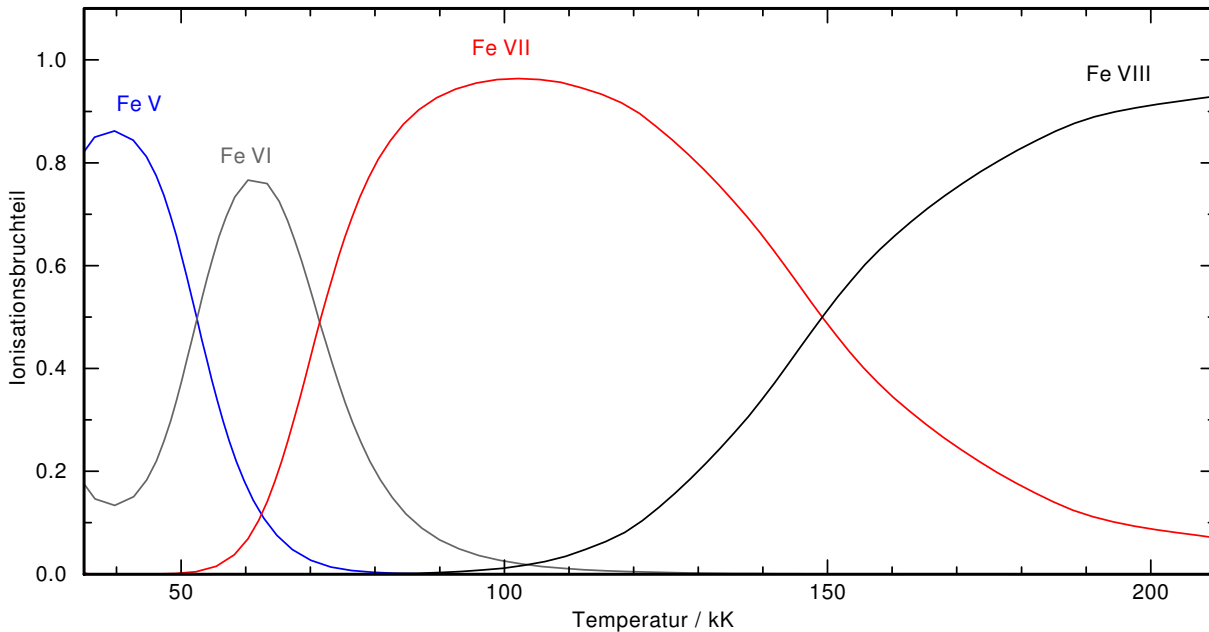


Abbildung 2.2.: Ionisationsbruchteile der Fe-Ionen V - VIII bei verschiedenen Temperaturen.

Die Eisengruppenelemente können nicht nur erfolgreich zur Erklärung bis dahin unidentifizierter Linien verwendet werden. Aufgrund ihrer großen Zahl an Linien können sie auch starke Absenkungen des Flussverlaufs in bestimmten Wellenlängenbereichen verursachen, wodurch der bis dahin unerklärbare Flussverlauf einiger Objekte modelliert werden konnte (z.B. Lanz et al., 1996). Da die zugrunde liegenden Atomdaten für die Eisengruppenelemente jedoch nicht vollständig sind, kann nicht ausgeschlossen werden, dass die Eisengruppenelemente die Ursache für noch nicht modellierbare Flussverläufe sind. Mit der im Jahr 2009 begonnenen Erweiterung der Atomdaten für Eisengruppenelemente (Kurucz, 2009) ist eine erneute Untersuchung ihrer Auswirkungen auf den Flussverlauf von Modellspektren (dringend) erforderlich (Kapitel 4.3).

Die Behandlung der Eisengruppenelemente bei der Modellierung der Sternatmosphäre unterscheidet sich von der anderer Elemente. Wie bereits in Kapitel 1 beschrieben, können die Eisengruppenelemente aufgrund der enormen Anzahl an realen Levels und zugehörigen Übergängen nicht wie „klassische“ Atome behandelt werden. Es wird stattdessen ein statistisches Verfahren angewandt. Dieses Verfahren ist bereits implementiert und wird erfolgreich in der NLTE-Spektralanalyse verwendet. Die Querschnitte der radiativen gebunden-gebunden- (RBB) und gebunden-frei-Übergänge (RBF) sowie die Modellatome der Eisengruppenelemente werden mit diesem Verfahren in einem separaten Schritt vor der Modellierung der Sternatmosphäre berechnet bzw. erstellt. Im Rahmen dieser Arbeit wird ein VO-Dienst entwickelt, der die Erzeugung der RBB- und RBF-Querschnitte sowie der zugehörigen Modellatome nach Vorgaben des VO-Nutzers mit Hilfe dieses Verfahrens über das Internet ermöglicht. Mit diesem neuen VO-Dienst können detaillierte Querschnittsdaten für jedes beliebige Modellatmosphärenprogramm (nicht nur im NLTE-Bereich) erstellt und die Eisengruppenelemente bei der Modellierung jeder Sternatmosphäre berücksichtigt werden.

Im Folgenden wird zuerst beschrieben, wie dieses statistische Verfahren für die Berücksichtigung der Eisengruppenelemente in der Spektralanalyse funktioniert (Kapitel 2.1) und wie mit dem in Tübingen verwendeten Programm Querschnittsdaten und Modellatome erzeugt werden (Kapitel 2.1.1). Dann wird unsere Quelle der Atomdaten für Eisengruppenelemente und die Änderungen aufgrund ihrer Aktualisierung kurz vorgestellt (Kapitel 2.2). Das bisherige Verfahren wird in mehreren Punkten verändert, um schneller realistischere Ergebnisse zu bekommen (Kapitel 2.3). Abschließend wird das neue Verfahren, das die Querschnitte und Modellatome dem VO-Nutzer zur Verfügung stellt, beschrieben (Kapitel 2.4).

## 2.1. *IronIc* – Statistische Behandlung der Eisengruppenelemente

Das von der Tübinger Arbeitsgruppe verwendete statistische Verfahren zur Reduzierung der großen Level- und Linienzahl der Eisengruppenelemente in den NLTE-Modellatmosphärenrechnungen wurde ursprünglich von Anderson (1985, 1989, 1992) für kühle (sonnenähnliche) Sterne entwickelt. Später wurde es von Dreizler & Werner (1992, 1993) in *TMAP* implementiert. Deetjen (1999) schrieb diese Version in eine etwas schnellere um und nannte sie Iron Opacity and Interface (*IronIc*, Rauch & Deetjen, 2003). *IronIc* besteht wie *TMAP* aus einer Sammlung von FORTRAN-Programmen und wird über Unix Shell-Skripte bedient. Im zugrunde liegenden statistischen Verfahren wird der Energiebereich zwischen Grundzustands- und Ionisationsenergie eines Ions der Eisengruppenelemente in mehrere Bänder unterteilt. Alle realen Levels (Index  $l$ ) innerhalb eines Bands tragen zur Energie ( $E_L$ ) und zum statistischen Gewicht ( $G_L$ ) dieses so genannten Superlevels (Index  $L$ ) bei über

$$E_L = \sum_{l \in L} E_l g_l^* / \sum_{l \in L} g_l^*,$$

$$G_L = e^{E_L/kT^*} \times \sum_{l \in L} g_l^*,$$

$$g_l^* = a_s g_l e^{-E_l/kT^*}.$$

Hier steht  $T^*$  für die Temperatur, bei der die betrachtete Ionisationsstufe dominant ist,  $a_s$  für das Verhältnis der Häufigkeit des jeweiligen Elements  $s$  zur Eisenhäufigkeit (bzw. der des häufigsten Elements) und  $k$  für die Boltzmannkonstante. Mit diesem statistischen Ansatz werden alle realen Levels zu typischerweise sieben Modellniveaus<sup>1</sup> zusammengefasst, die – im Gegensatz zu den realen Levels – als NLTE-Levels behandelt werden und die über die verschiedenen Übergänge gekoppelt sind.

Um Querschnittsdaten für die Übergänge zwischen den Superlevels zu erhalten, werden die Querschnitte der realen Übergänge berechnet und zu Bandübergangsquerschnitten gewichtet aufaddiert. Falls die Parität ignoriert wird, sind auch Übergänge innerhalb eines Bandes erlaubt. Diese werden nur als Hintergrundopazität berücksichtigt, nicht bei der Lösung der statistischen Gleichungen. Die Querschnitte der einzelnen Übergänge werden folgendermaßen berechnet:

- Die **radiativen gebunden-gebunden-Übergänge (RBB)** zwischen den Bändern  $L$  und  $U$  werden mit

$$\sigma_{LU} = \frac{\pi e^2}{m_e c} \sum_{(l \in L; u \in U)} g_l^* f_{lu} \Phi(\nu_{lu} - \nu) / \sum_{l \in L} g_l^*$$

berechnet.  $\frac{\pi e^2}{m_e c}$  stellt hierbei den Querschnitt eines klassischen Oszillators dar,  $f_{lu}$  die Oszillationsstärke des Übergangs und  $\Phi(\nu_{lu} - \nu)$  das Linienprofil, das durch die Voigtfunktion beschrieben wird (Kapitel 2.3.2). Mit dieser Berechnungsmethode tragen alle realen Übergänge mit ihrer richtigen Wellenlänge und Stärke zur Gesamtopazität bei. Die Querschnitte werden für zwei Elektronendichten ( $0$  und  $10^{16} \text{ cm}^{-3}$ ) berechnet und später von *TMAP* auf die Elektronendichten in der Modellatmosphäre interpoliert.

- Falls einzelne **radiative gebunden-frei-Übergänge (RBF)** im Opacity Project berechnet wurden, können diese Querschnitte verwendet und aufaddiert werden. Ansonsten werden wasserstoffähnliche Querschnitte angenommen.
- **Gebunden-frei Stoß-Übergänge (CBF)** und **gebunden-gebunden Stoß-Übergänge (CBB)** werden wie bei den klassischen Modellatomen von *TMAP* nach der Seaton-Formel (Seaton, 1962) bzw. van Regemorter-Formel (van Regemorter, 1962) bestimmt.

<sup>1</sup>Zum Vergleich: Die detaillierte Behandlung beispielsweise des Fe VI-Ions würde die Berücksichtigung von 24 112 Levels voraussetzen.

### 2.1.1. Bedienung des *IrOnIc*-Programms

Um mit dem ursprünglichen *IrOnIc*-Programm RBB- und RBF-Querschnittsdateien und Modellatome zu erstellen, sind drei verschiedene Programmdurchläufe nötig.

- Im ersten Durchlauf wird  $T^*$ , die Temperatur, bei der die jeweilige Ionisationsstufe dominant ist, für jede gewünschte Ionisationsstufe mit Hilfe der Saha-Gleichung bestimmt.  $T^*$  wird zur Berechnung der statistischen Gewichte der Superlevels und bei der Berechnung der Querschnitte verwendet.
- Im nächsten Schritt werden vorläufige Daten erzeugt, um ein Modellatom generieren zu können. Zu diesen Atomdaten wird anschließend vom *IrOnIc*-Anwender ein Frequenzgitter erstellt.
- Dieses Frequenzgitter wird im dritten Durchlauf verwendet, um detaillierte RBB-Querschnitte für jeden Frequenzpunkt zu berechnen. Das Frequenzgitter durfte in der ursprünglichen *IrOnIc*-Version nach diesem Durchlauf nicht mehr geändert werden, sondern wurde genau so von *TMAP* verwendet.

Die Parameter des *IrOnIc*-Programms (z.B. die einzelnen Ionen mit ihrer jeweiligen Bandstruktur) werden für jeden Durchlauf in einer Datei (meist `ironic_Element.in` genannt, siehe Anhang) gespeichert. Für die Steuerung des Programms wird ein Unix-Shell-Skript (meist `ironic_Element.job`, siehe Anhang) verwendet. Diese beiden Dateien (im Folgenden Steuerdateien genannt) müssen vor jedem Durchlauf geringfügig angepasst werden.

Die Ausführung der drei Berechnungsschritte bedarf – abhängig von den gewählten Ionen und der Zahl der Frequenzpunkte – etwa ein bis zwei Tage Rechenzeit, wobei der dritte Schritt die meiste Zeit benötigt. Nach der Berechnung der Daten für die Superlevels und Superlinien wird mit diesen ein Modellatom in *TMAP*-Format erzeugt. Darin sind alle Energien und statistischen Gewichte der Superlevels, sowie die zur Berechnung der CBF- und CBB-Querschnitte nötigen Daten enthalten. Die Querschnitte der RBF- und RBB-Übergänge sind in speziellen Dateien gespeichert, auf die im Modellatom verwiesen wird. Die so erzeugten Modellatome und die zugehörigen Querschnittsdateien werden zur Modellierung der Sternatmosphäre an den Modellatmosphärenencode übergeben.

## 2.2. Die Quelle für Eisengruppendaten: Linienlisten von Kurucz

Die Genauigkeit der Atomdaten hat großen Einfluss auf die Qualität und Zuverlässigkeit der berechneten Modellatmosphäre. Für die „klassischen“ Modellatome wurde in Tübingen deshalb die *TMAD*-Atomdatenbank erstellt (Kapitel 3.2). Für *IrOnIc* und die damit berechneten Eisengruppenmodellatome werden die Linienlisten von Kurucz verwendet (Kurucz, 2009, 2011). Diese Linienlisten werden seit 1965 für unterschiedliche Elemente und verschiedene Ionisationsstufen aufgebaut. Für die Eisengruppenelemente waren 1988 alle Listen für die Ionisationsstufen I bis IX erstellt. Zu diesem Zeitpunkt enthielten sie insgesamt 42 Millionen Linien (Kurucz, 2009), quantenmechanisch berechnet und, falls verfügbar, durch die im Labor bestimmten ersetzt.

Kurucz's ursprüngliche Motivation für diese Listen war, das Spektrum der Sonne mit ihrer Hilfe so zu modellieren, dass alle Merkmale reproduziert werden können. Dieses Ziel wurde mit den Listen von 1988 nicht erreicht. Ein Grund dafür war, dass die Listen nicht vollständig sind. Deshalb fing Kurucz 2009 an, diese zu erweitern und zu verbessern (in Tabelle 2.1 und Abb. 2.3 ist gut zu erkennen, wie viele neue Linien nun mitberechnet werden können), wobei er auch das Format seiner Dateien änderte. Mittlerweile enthalten die Listen etwa 250 Millionen Linien für die Eisengruppenelemente (Stand: Dezember 2012).

Die Listen können direkt von seiner Webseite<sup>2</sup> heruntergeladen werden, wo sie in Verzeichnissen (xxyy), benannt nach der Ordnungszahl des jeweiligen Elements (xx) und seiner Ladung (yy), eingeordnet sind. Zusätzlich zu den Linienlisten sind weitere Dateien verfügbar, die Daten enthalten, die zur

<sup>2</sup><http://kurucz.harvard.edu/atoms.html>

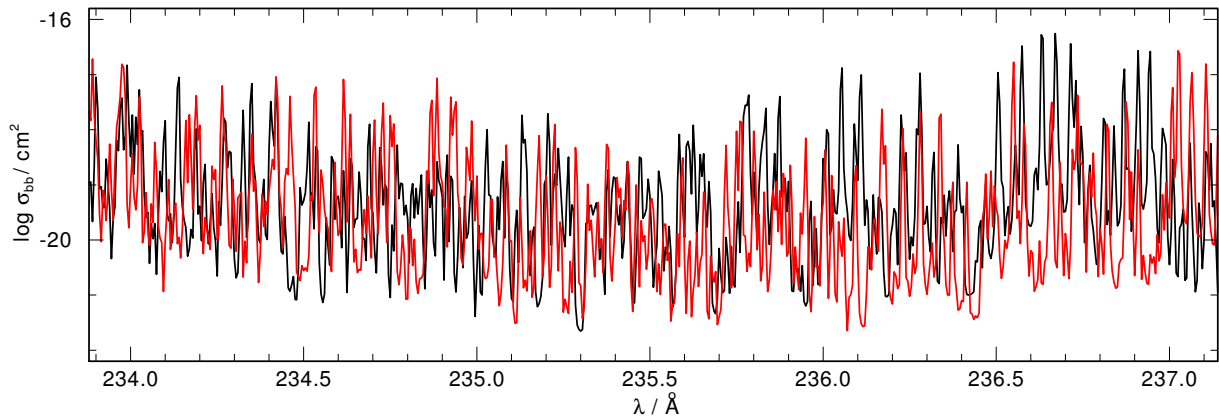


Abbildung 2.3.: Vergleich zweier RBB-Querschnitte (Linientemperatur 48 750 K, Elektronendichte 0), die mit den alten (schwarz) und den neuen Atomdaten (rot) erzeugt wurden. Dieses Beispiel zeigt die RBB-Querschnitte von Fe VI vom in dieser Rechnung gewählten Band zwei zum Band sechs.

Berechnung der Listen notwendig waren. Ihre Bedeutung wird im Kopfteil der Webseite erklärt. *IrOnIc* verwendet die GFxxyy.GAM-, GFxxyy.LIN- und GFxxyy.POS-Dateien. Sie beinhalten Werte für die Levels und die zugehörigen Übergänge. Sind Dateien im neuen Format (nach 2009) vorhanden, so werden diese für *IrOnIc* ins alte Format umgewandelt, um konsistente Eingabewerte zu garantieren.

**Die GFxxyy.GAM-Dateien** beinhalten eine Kopfzeile, in der die Zahl der Linien, die in der zugehörigen GFxxyy.LIN-Datei enthalten sind, die Anzahl der Levels mit gerader und ungerader Parität und die Ionisationsenergie des Ions angegeben ist. Dann folgt eine Tabelle (Tabelle 2.2), in der die einzelnen Energieniveaus beschrieben sind, angefangen mit allen Levels gerader Parität. Die einzelnen Einträge dieser Tabelle enthalten die Ordnungszahl und Ladung, die Parität (EVE oder ODD), die fortlaufende Nummer der Levels, die Energie des Levels in Kayser, den Drehimpuls  $J$  und weitere Komponenten, die zur Berechnung der Levels nötig waren, aber nicht von *IrOnIc* verwendet werden.

**Die GFxxyy.LIN-Dateien** enthalten Angaben über die Linien, die zu den Übergängen zwischen Levels der GFxxyy.GAM-Dateien gehören. Jede einzelne Zeile steht dabei für einen Linienübergang. Die Linien sind nach Wellenlänge (in nm) sortiert. Falls die Werte größer als 200 nm sind, stehen sie für Luftwellenlängen, ansonsten für Vakuumwellenlängen. Die Tabelle der GFxxyy.LIN-Dateien enthält außer der Wellenlänge den log gf-Wert des Übergangs, die laufende Nummer des Energieniveaus mit gerader Parität in der zugehörigen GFxxyy.GAM-Datei, sowie die des Niveaus mit ungerader Parität, gefolgt von einer 1 (falls die Wellenlänge der Linie nur berechnet und nicht gemessen ist) oder 0 (wenn sie im Labor bestimmt wurde). Auf seiner Webseite bietet Kurucz die Programme EXPAND3007.FOR und EXPANDFORBIDDEN3007.FOR an. Diese können verwendet werden, um die GFxxyy.LIN-Tabellen so zu erweitern, dass die Informationen aus den GFxxyy.GAM-Dateien direkt beim jeweiligen Eintrag der GFxxyy.LIN-Dateien stehen.

**Die GFxxyy.POS-Dateien** sind den GFxxyy.LIN-Dateien sehr ähnlich. Auch sie enthalten eine Linientabelle in vergleichbarem Format (die zugehörigen Energieniveaus sind dabei nicht als laufende Nummer der GFxxyy.GAM-Datei angegeben, sondern ausgeschrieben mit Energie und Drehimpuls  $J$ ). Es werden allerdings nur Linien berücksichtigt, deren zugehörige Energieniveaus im Labor bestimmt wurden (also die, die in der GFxxyy.LIN-Datei mit einer 0 gekennzeichnet sind). POS-Übergänge werden daher auch als Linien mit „guter Wellenlänge“ bezeichnet.

Tabelle 2.1.: Statistik der Eisengruppenelemente in der Kurucz-Datenbank verglichen mit dem Stand vor 2009 (für einen besseren Überblick sind nur Werte bei einer Veränderung angegeben).  
Stand: 19.12.2012

	Levels	POS-Linien	POS-Linien (vor 2009)	LIN-Linien	LIN-Linien (vor 2009)
Ca I	1159		22336		45753
Ca II	2885	2433	946	55732	4227
Ca III	801	2897	2280	58087	11740
Ca IV	2281		410		82110
Ca V	3052		121		330004
Ca VI	2499		111		217927
Ca VII	1588		72		125560
Ca VIII	1468		187		30156
Ca IX	565		641		22803
Sc I	4332	15546	7274	737992	191253
Sc II	1153	3436	3387	116264	49811
Sc III	240		226		1578
Sc IV	558		1204		16985
Sc V	3311	2180	36	645343	130563
Sc VI	3052		119		456400
Sc VII	2499		124		256997
Sc VIII	1588		72		136916
Sc IX	1468		95		30587
Ti I	13978	32821	12723	4754246	867399
Ti II	4414	8188	4557	835006	264867
Ti III	10961	4090	3071	499510	23742
Ti IV	401		153		5079
Ti V	783		445		37610
Ti VI	2281		77		155919
Ti VII	3052		143		356808
Ti VIII	2499		128		230705
Ti IX	1588		76		136448
V I	29719	23342	16649	7043452	1156790
V II	14162	18389	9455	3932806	925330
V III	4412	9892	8439	966458	284003
V IV	1221		1245		61630
V V	401		401		8427
V VI	783		430		39525
V VII	2281		36		160652
V VIII	3052		92		443343
V IX	2499		111		231753
Cr I	37502	35315	16597	2582957	434773
Cr II	29657	58998	36345	7858507	1304043
Cr III	11986	23150	4767	5535931	990951
Cr IV	4740		2443		366851
Cr V	1221		249		73222
Cr VI	401		318		10886
Cr VII	783		538		39668
Cr VIII	2281		35		164228
Cr IX	3052		80		454312
Mn I	11952	16798	9880	1481464	327741
Mn II	39506	31437	19840	4523283	878996
Mn III	29596	17294	12812	10525061	1589314



Tabelle 2.1.: Fortsetzung

	Levels	POS-Linien	POS-Linien (vor 2009)	LIN-Linien	LIN-Linien (vor 2009)
Mn IV	11986		1243		1033926
Mn V	4740		908		450293
Mn VI	1221		1351		79068
Mn VII	401		65		14024
Mn VIII	783		122		39770
Mn IX	1919		39		5
Fe I	37505	93508	18143	6029023	789176
Fe II	39423	123541	45815	7834247	1264969
Fe III	39540	37199	23059	9548612	1604934
Fe IV	27978	8408	7897	14617228	1776984
Fe V	14086	11417	3670	7785320	1008385
Fe VI	24112	3534	1100	9072714	475750
Fe VII	17485	2326	942	2916991	90250
Fe VIII	3955	233	62	220177	14561
Fe IX	784		138		39346
Co I	24005	15432	9876	3770844	546130
Co II	38019	23355	6132	10050728	1361114
Co III	37995	9356	4081	11515111	2198940
Co IV	13184		8404		1569347
Co V	13705		7751		2032402
Co VI	11986		3634		1089039
Co VII	4740		437		562192
Co VIII	1221		294		88976
Co IX	401		57		15185
Ni I	10061	9663	3949	731627	149925
Ni II	21699	55590	29036	3645991	404556
Ni III	38019	21251	9556	11120348	1309729
Ni IV	37860	5659	5471	15152635	1918070
Ni V	37446	10637	9857	15622436	1971819
Ni VI	29498	12219	7953	17971635	2211919
Ni VII	47610	3502	230	28327975	967466
Ni VIII	24112	758	204	12208124	602486
Ni IX	12139	253	82	2671445	79627
Ni X	3955	135		285029	

Wie in Tabelle 2.1 zu sehen ist, existieren deutlich mehr theoretische als im Labor bestimmte Linien – vor allem für höhere Ionisationsstufen. Das Verhältnis der Anzahl von LIN- zu POS-Linien liegt etwa bei 300.

**Für *IrOnIc*** wurden bereits während der Implementierung des Programms die verfügbaren LIN-, GAM- und POS-Dateien heruntergeladen. Da seit 2009 einige der Dateien erweitert wurden, wurden bereits vor dieser Arbeit einige der alten Dateien durch neue ersetzt. Dieser Prozess war allerdings nicht abgeschlossen und im Laufe der Arbeit kamen neue Erweiterungen hinzu. Deshalb wurden die Atomdaten seit Beginn der Arbeit kontrolliert und gegebenenfalls nach aktuellen Kriterien (siehe unten) korrigiert oder aktualisiert. Außerdem wurde ein Cron-Job eingeführt, der einmal pro Woche automatisch prüft, ob neue Atomdaten vorhanden sind und den *IrOnIc*-Administrator per Email darüber benachrichtigt.

Seit der Erweiterung der Atomdaten wurden nicht nur die Formate der Dateien geändert. Manche Dateien wurden aufgeteilt in bis zu drei neue Dateien, deren Namen zusätzlich zum alten Namen einen Buchstaben „w“, „y“ oder „z“ enthalten. „z“ kennzeichnet dabei die Dateien, die im Labor bestimmte

Tabelle 2.2.: Beispiel für zwei Energieniveaus mit dem selben Namen, Parität und Drehimpuls, aber unterschiedlicher Energie. Diese Beispiel wurde der GF2701.GAM-Datei entnommen.

```
27.01EVE 0.0 30 -127527.971 5(4F)5D 33 0.000 1.16E+08 3.59E-13 2.89E-31-0.264
27.01EVE 0.0 31 -127895.480 5(4F)5D 33 0.000 2.11E+08-9.34E-14 2.00E-31 0.023
```

Werte enthalten. Für die *IrOnIc*-Rechnungen werden diese Dateien wieder zu einer großen Datei vereint. Außerdem treten seit der Erweiterung in einzelnen Dateien Zeilen auf, die nur aus Nullen bestehen. Diese werden für *IrOnIc* gelöscht.

GAM-Dateien im neuen Format enthalten nicht mehr die kompletten Namen der Energieniveaus, sondern Abkürzungen, die im Kopfteil der Datei aufgelistet sind. Es wurde geprüft, ob es dadurch zu inkonsistenten Benennungen kommt. Dieser Test zeigte, dass es Levels gibt, die den selben Namen, Parität und Drehimpuls  $J$  haben, aber unterschiedliche Energien (ein Beispiel ist in Tabelle 2.2 dargestellt, für Fe IV z.B. gibt es über 3000 solcher Levels). In den meisten Fällen handelt es sich dabei um Levels mit negativer (also berechneter) Energie und um Autoionisationsniveaus, die von *IrOnIc* ignoriert werden. Da *IrOnIc* nicht die Namen der Levels, sondern nur ihre Energien verwendet, sind auch für die anderen Levels, bei denen dieser Sachverhalt auftritt, keine Probleme durch diese ungenaue Bezeichnung zu erwarten.

## 2.3. Die *IrOnIc*- und *TIRO*-Parameter

Um dem *VO*-Nutzer detaillierte Querschnittsdaten und Modellatome für Eisengruppenelemente zur Verfügung zu stellen, die Berechnung der Querschnitte zu beschleunigen und die Bedienung des *IrOnIc*-Programms zu vereinfachen, wird die Handhabung von *IrOnIc* im Rahmen dieser Arbeit verändert. Dafür wird der Tübingen Iron-Group Opacity- (*TIRO*-)Dienst entwickelt, mit dem der Nutzer über das Internet Zugang zu den gewünschten Daten der Eisengruppenelemente bekommt. Das ermöglicht eine noch genauere Analyse von Sternen mit Hilfe von *VO*-Diensten als bisher mit den bereits bestehenden Diensten *TMAW*, *TMAD* und *TheoSSA* (Kapitel 3) möglich war. Dem *TIRO*-Anwender wird die Bedienung des *IrOnIc*-Codes zusätzlich erleichtert, indem die Zahl der Parameter (im Folgenden beschrieben) reduziert wird. Die Parameter, die in der neuen Version fest gesetzt sind, wurden bereits bei der Verwendung der alten Version des Programms kaum bzw. nicht variiert. Um diese Parameter in der neuen Version nach bestem Kenntnisstand zu setzen, wurden sowohl Tests aus der Literatur zu Rate gezogen als auch der Einfluss einiger Parameter auf die Querschnitte bzw. Modellatmosphäre selbst getestet, was im Folgenden beschrieben wird. Es wurde außerdem ein Fehler im *IrOnIc*-Programm korrigiert, durch den einem Energieniveau ein falsches statistisches Gewicht zugeordnet wurde, falls für einen Linienübergang die Energie des Niveaus mit gerader Parität höher als die des Niveaus mit ungerader Parität ist.

### 2.3.1. Energiebänder und Superlevels

Die Zahl der Energiebänder pro Ion, in die die realen Levels eingeteilt werden, wurde bereits ausführlich von Haas (1997) und Hoffmann (2005) getestet. Bei der Bänderwahl gilt zu beachten, dass der Energiebereich, den ein Band abdeckt sowohl klein genug sein muss, um ein Boltzmann-Gleichgewicht zwischen den realen Levels zu garantieren, als auch groß genug, um ausreichend viele reale Levels zu beinhalten, also um möglichst wenige NLTE-Levels zu erhalten. Die Entwickler des Verfahrens, Anderson (1989) und Dreizler & Werner (1992), hatten für ihre Tests jeweils sieben Bänder pro Ion verwendet. Hubeny & Lanz (1995) implementierten die Idee von Anderson (1989) ebenfalls in ihren NLTE-Code *TLUSTY*<sup>3</sup> (Hubeny, 1988; Lanz & Hubeny, 2003; Hubeny & Lanz, 2011). Sie berücksichtigten zusätzlich noch die Parität der Levels und verwendeten 20 bis 30 Bänder pro Ion für ihre Rechnungen.

<sup>3</sup><http://nova.astro.umd.edu/>

Haas (1997) berechnete für seine Tests der Bänderzahl pro Ion Querschnitte und *TMAP*-Modelle mit Sternparametern, die typisch für sdO-Sterne sind. Er verglich die RBB-Querschnitte und Modellatmosphären von generischen Ionen<sup>4</sup>, die sieben und 20 Bänder enthielten. Das Ergebnis war, dass sieben Bänder für eine zuverlässige Spektralanalyse ausreichen, da die auftretenden Unterschiede statistisch sind und bei der Sternatmosphärenberechnung im Mittel vernachlässigbare Auswirkungen haben. Aufgrund dieser Tests werden für den neuen *TIRO*-Dienst sieben Bänder festgelegt. Bei NLTE-Rechnungen sind die Besetzungszahlen höherer Levels verglichen mit dem LTE-Fall kleiner, da weniger Elektronenstöße dafür sorgen, dass die Ionen angeregt bleiben. Niedrigere Levels sind im NLTE-Fall deshalb stärker zu berücksichtigen als höhere. Höhere reale Levels erfüllen untereinander die LTE-Bedingungen besser als niedrigere. Deshalb werden die Bänder des *TIRO*-Dienstes so gewählt, dass die Zahl der realen Levels eines Ions logarithmisch mit der Bandnummer zunimmt. Die resultierenden NLTE-Superlevels beinhalten dann für den Grundzustand wenige reale Levels, für das höchste NLTE-Superlevel die meisten realen Levels.

### 2.3.2. Algorithmen für die Berechnung der Voigtfunktion

Spektrallinien werden durch mehrere Mechanismen verbreitert. Am wichtigsten sind dabei die natürliche Linienbreite und Druckverbreiterung, die durch Lorentzprofile dargestellt werden, und die thermische Verbreiterung und Mikroturbulenz, die durch Gaussprofile repräsentiert werden. Die Faltung eines Lorentz- und eines Gaussprofils ergibt insgesamt ein Voigtprofil, welches durch die auf  $\sqrt{\pi}$  normierte Voigtfunktion

$$H(\alpha, v) = \frac{\alpha}{\pi} \int_{-\infty}^{\infty} \frac{e^{-x^2}}{(v-x)^2 + \alpha^2} dx$$

ausgedrückt wird, wobei  $v = (\nu - \nu_0) / \Delta\nu_D$  der gewichtete Abstand zum Linienzentrum,  $\alpha = \gamma / 4\pi \Delta\nu_D$  der Dämpfungsparameter und  $\Delta\nu_D = \nu_0 * (v_{th}/c)$  die Dopplerbreite ist.

Analytisch ist die Voigtfunktion nicht darstellbar. Statt dessen werden Näherungsformeln verwendet, um sie zu beschreiben. In der Literatur sind mehrere solcher Näherungsroutinen zu finden, die bei gleichen Eingabewerten unterschiedliche Rechenzeiten benötigen. Ein Test mit dem Profiler der Portland Group *pgprof*<sup>5</sup> am *IrOnIc*-Code zeigt, dass die Berechnung der Voigtprofile 89% der gesamten Rechenzeit einnimmt. Die Wahl der Voigtoutine bestimmt somit grundlegend die Wartezeit auf die benötigten Querschnitte.

Da sich die Rechnerarchitektur und die Compiler seit der Erstellung des *IrOnIc*-Codes verändert haben, wurden Voigtoutinen aus der Literatur getestet, um die schnellste Näherungsroutine zu finden und so den Code eventuell zu beschleunigen. Die getesteten Routinen sind von Schreier (1992), Hellwege (1982), Peytremann (1974), Humlíček (1982), Humlíček (1979), Armstrong (1967), Behara & Jeffery (2006, priv. Mitt.), Hellwege (1965), Garcia et al. (2009) und eine von Curtis & Osborne (1966) mit Werten von Hui et al. (1978). Da die Voigtfunktion und somit auch ihre numerischen Umsetzungen von zwei Parametern abhängen, dem Dämpfungsparameter  $\alpha$  und dem mit der Dopplerbreite gewichteten Abstand zum Linienzentrum  $v$ , wurde zuerst das Intervall ermittelt, das der Parameter  $\alpha$  überdeckt. Dann wurden für Werte von  $\alpha$  zwischen 0.00001 und 100 in Schritten mit Faktor zehn die Voigtprofile für ein künstliches Frequenzgitter mit 100 000 Punkten und 1000 künstlichen Linienzentren mit den einzelnen Routinen 100 mal berechnet (Abb. 2.5). Als schnellste Voigtoutine stellte sich die von Hellwege (1965) heraus, die im Gegensatz zu vielen anderen Implementierungen, keine komplexen Zahlen verwendet. Je nach Wert des Parameters  $\alpha$  folgen die Routinen von Hellwege (1982), Peytremann (1974), Behara & Jeffery (2006) und Humlíček (1982).

Die Routinen von Curtis & Osborne (1966), Garcia et al. (2009) und Armstrong (1967) liefern nicht in

<sup>4</sup>Bei generischen Modellatomen werden alle realen Levels aller Eisengruppenionen der selben Ionisationsstufe in eine gemeinsame Bandstruktur eingeordnet.

<sup>5</sup><http://www.pgroup.com/products/pgprof.htm>

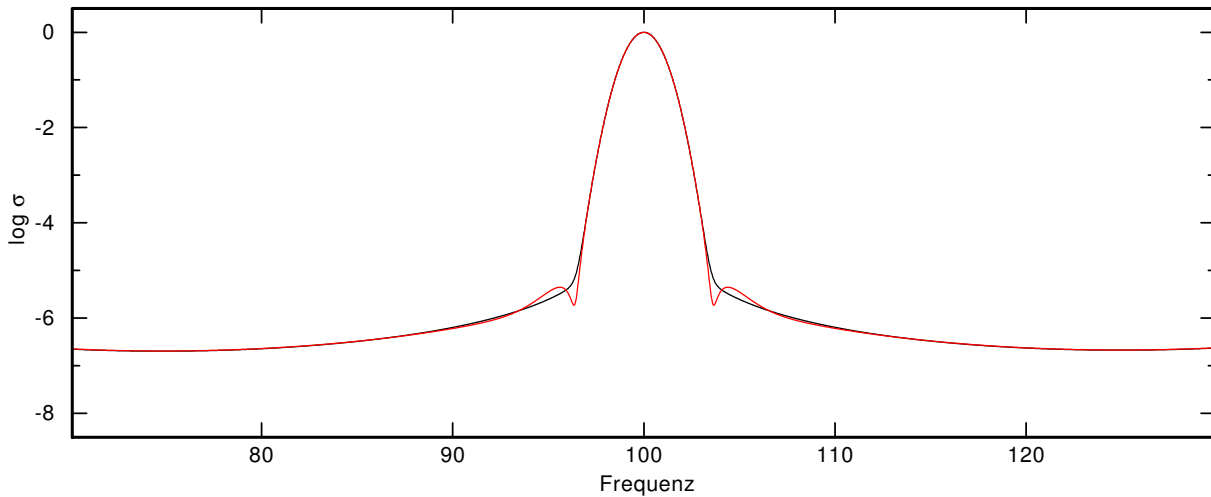


Abbildung 2.4.: Linienprofile berechnet mit den Routinen von Hellwege (1982, schwarz) und Curtis & Osborne (1966, rot) für  $\alpha = 10^{-4}$ .

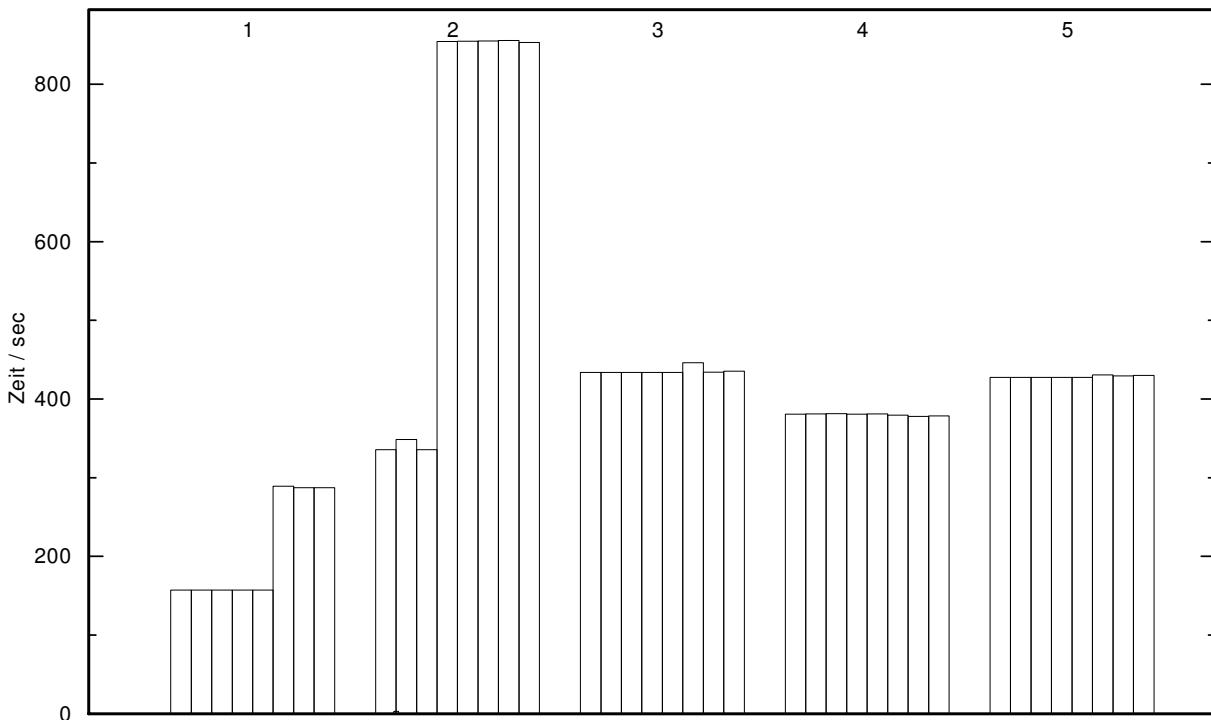


Abbildung 2.5.: Vergleich der Rechenzeiten der schnellsten getesteten Voigt-routinen. 1 steht hier für die Routine von Hellwege (1965), 2 für Hellwege (1982), 3 für Peytremann (1974), 4 für Humlíček (1982) und 5 für Behara & Jeffery (2006). Die Balken stehen für die verschiedenen Werte für die Dämpfungsbreite  $\alpha$  (von links nach rechts:  $10^{-5} - 10^2$ ,  $\Delta = 10$ )

allen getesteten Bereichen zuverlässige Ergebnisse und wurden deshalb von weiteren Tests ausgeschlossen (z.B. Abb. 2.4).

In der ursprünglichen *IrOnIc*-Version ist die Routine von Hellwege (1982) implementiert. Da die Routinen von Hellwege (1965), (Peytremann, 1974), (Humlíček, 1982) und (Behara & Jeffery, 2006) sich als – je nach Parameterbereich – schnellere Routinen herausstellten, wurden diese ebenfalls in die *IrOnIc*-Version implementiert. Für jede dieser Routinen wurde eine *IrOnIc*-Berechnung von Querschnitten für

Fe VI mit sieben Bändern mit einem Frequenzgitter mit 314 992 Punkten durchgeführt. Beim Vergleich der Rechenzeiten ergab sich, dass auch hier die Routine von Hellwege (1965) die schnellste ist (5182 sec). Danach folgen die Routinen von Peytremann (1974, 5219 sec), Behara & Jeffery (2006, 5301 sec), Hellwege (1982, 5500 sec) und Humlíček (1982, 5713 sec). In die neue *IrOnIc*-Version wird deshalb die Routine von Hellwege (1965) implementiert.

### 2.3.3. Zusätzliche Ionen

Das *IrOnIc*-Programm wurde ursprünglich für die Berechnung von Querschnitten für heiße, kompakte Objekte entwickelt, bei deren Analyse niedrige Ionisationsstufen keine Rolle spielen. Im ersten Berechnungsschritt von *IrOnIc* wird die Temperatur ermittelt, bei der die jeweils betrachtete Ionisationsstufe dominant ist. Dafür wird die Saha-Gleichung in einer Form verwendet, die Informationen über die jeweils vorherige und folgende Ionisationsstufe benötigt. Diese Temperatur wird für vier verschiedene Elektronendichten berechnen und anschließend gemittelt. Für die Ionisationsstufe I ist eine Berechnung dieser Temperatur über die Saha-Gleichung so nicht möglich.

Der neue *TIRO*-Dienst ist nicht mehr auf heiße, kompakte Objekte beschränkt, sondern liefert Querschnitte, die ebenfalls zur Analyse kühler Sterne verwendet werden können. Dafür muss die eben beschriebene Temperatur auch für nicht ionisierte Atome berechnet werden können. Der neue *TIRO*-Dienst wurde deshalb um eine Extrapolation erweitert, bei der aus den bekannten Werten höherer Ionisationsstufen der Wert für die Stufe I extrapoliert wird. Mit dieser Änderung ist *IrOnIc* bzw. *TIRO* für alle Objekte verwendbar, zu deren Analyse Ionen bis zur Ionisationsstufe IX wichtig sind.

### 2.3.4. Generische Modellatome – Die Wahl der Ionisationsenergie

Bei der Erzeugung generischer Modellatome werden alle realen Levels aller berücksichtigten Eisen- und Nickel-Elemente einer Ionisationsstufe in eine gemeinsame Bandstruktur eingeordnet. Das resultierende generische Modellatom hat den Vorteil, dass die Opazitäten mehrerer Eisen- und Nickel-Elemente mit wenigen zusätzlichen NLTE-Levels berücksichtigt werden können. Für die Erzeugung eines generischen Modellatoms muss erfüllt sein, dass das Ionisationsverhalten der einzelnen Elemente ähnlich ist. Haas (1997) zeigte, dass diese Annahme durchaus gültig ist, solange nur die Line-Blanketing-Effekte modelliert werden sollen. Für detaillierte Häufigkeitsbestimmungen ist dieses Verfahren nicht geeignet.

Vor dieser Arbeit hatte *IrOnIc* als Ionisationsenergie des generischen Modellatoms die Ionisationsenergie des ersten aufgeführten Elements verwendet (siehe Listing A.8, Eintrag nach Schlüsselwort „ELEMENT“). Wie in Abb. 2.6 zu sehen ist, kann es durchaus vorkommen, dass Elemente mit höherer Ladung eine höhere Ionisationsenergie haben. Alle Levels zwischen der Ionisationsenergie des generischen Modellatoms und der eigentlich richtigen werden dann abgeschnitten. Zum Beispiel fehlen in einem generischen Modellatom aus Ca und Sc für die Ionisationsstufe IV in der alten *IrOnIc*-Version Linien von Sc. In einem generischen TiFe-Modellatom für die selbe Ionisationsstufe werden Eisenlinien ignoriert.

Die Auswirkung der fehlenden Linien auf die Querschnitte eines generischen Ions wurden anhand eines generischen Modellatoms aus Ca und Sc mit den Ionisationsstufen III-IV mit sieben bis auf die Ionisationsenergie (592 732 Kayser bzw. 542 600 Kayser für die Berücksichtigung nach der neuen bzw. alten Methode für die Ionisationsstufe IV) identischen Bändern getestet. Als Linientemperatur wurde 45 000 K gewählt und das verwendete Frequenzgitter besteht aus 198 000 Punkten. Der Test wurde für ein generisches Modellatom aus Sc und Ti mit einer Ionisationsenergie von 221 736 Kayser bzw. 199 677 Kayser für die Ionisationsstufe III und 592 732 Kayser bzw. 348 973 Kayser für die Ionisationsstufe IV wiederholt. Er ergab, dass den RBB-Querschnitten von Übergängen zum höchsten Superlevel, bei denen die „falsche“ Ionisationsenergie berücksichtigt wird, einige Linien fehlen (Abb. 2.7). Da diese zu Levels höherer Energie gehören, sind sie bei anderen Wellenlängen als die restlichen Linien des Übergangs zu finden.

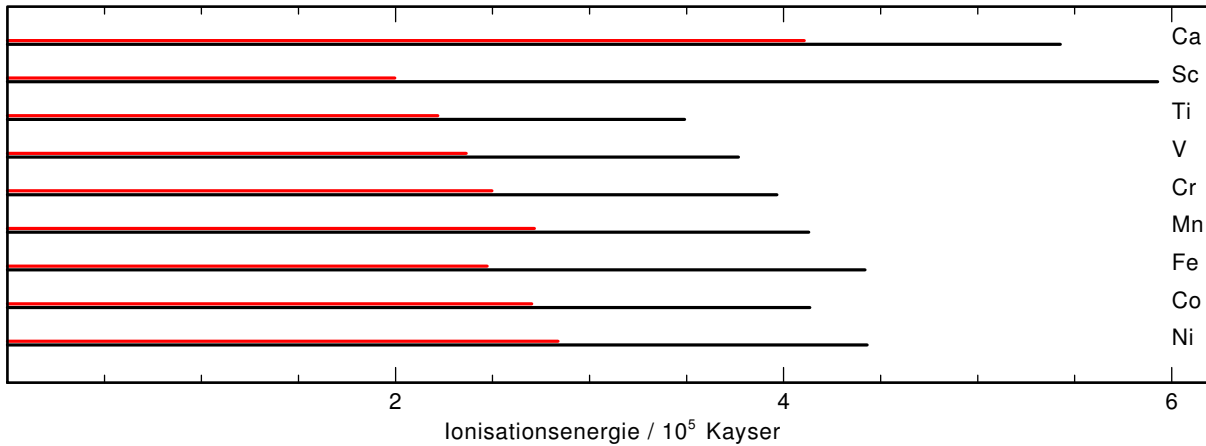


Abbildung 2.6.: Die Ionisationsenergien der Eisengruppenelemente für die Ionisationsstufen III (rot) und IV (schwarz).

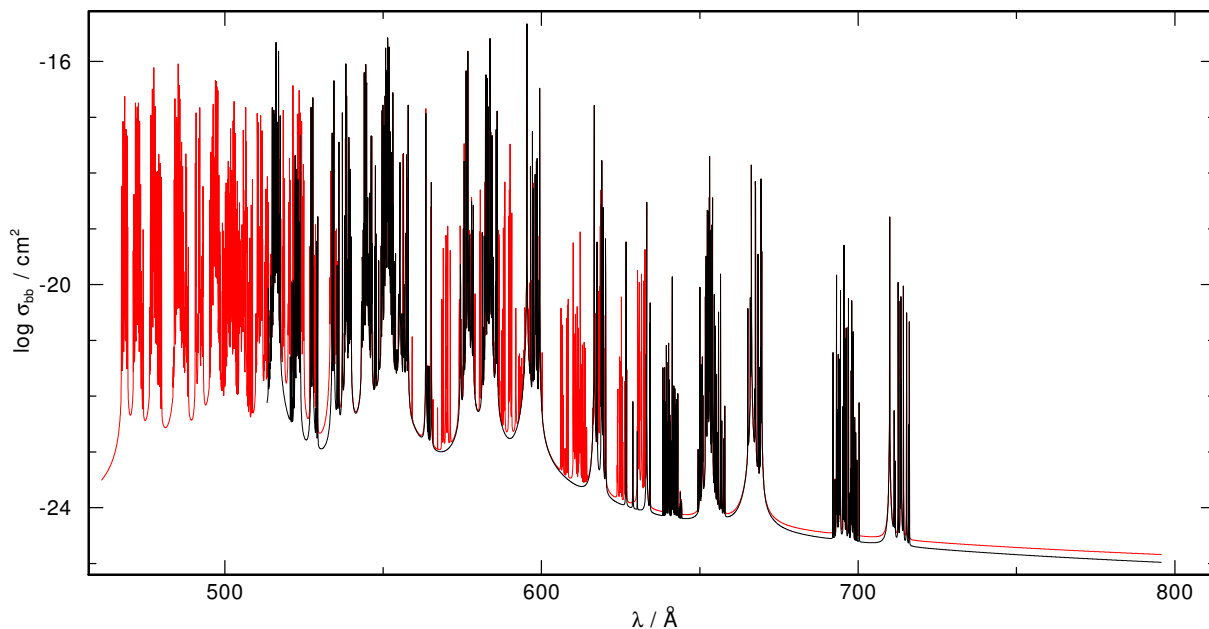


Abbildung 2.7.: Radiative gebunden-gebunden-Querschnitte für ein generisches Modellatom (Sc+Ti), berechnet mit der Ionisationsenergie des ersten Elements (schwarz, siehe Text) und mit der höchsten Ionisationsenergie aller berücksichtigten Elemente (rot). Die Übergänge zwischen dem gewählten Band eins und sieben der Ionisationsstufe III sind dargestellt.

Die RBF-Querschnitte ändern sich bei unterschiedlicher Ionisationsenergie des Ions ebenfalls. Wenn eine höhere Ionisationsenergie berücksichtigt wird, tragen ebenfalls mehr einzelne Querschnitte zur Summe bei.

Im neuen *TIRO*-Dienst wird nicht mehr die Ionisationsenergie des ersten angegebenen Elements eines generischen Modellatoms, sondern die höchste aller berücksichtigten Elemente verwendet. Dadurch werden weitere Opazitätsquellen einbezogen und das generische Modellatom ist realistischer.

### 2.3.5. Linientemperatur und Zahl der Frequenzpunkte

Die Breite von Spektrallinien ist abhängig von der Dopplerbreite. Diese wird in der Regel für eine Temperatur berechnet, die bei  $3/4$  der Effektivtemperatur (der so genannten Linientemperatur  $T_L$ , die der

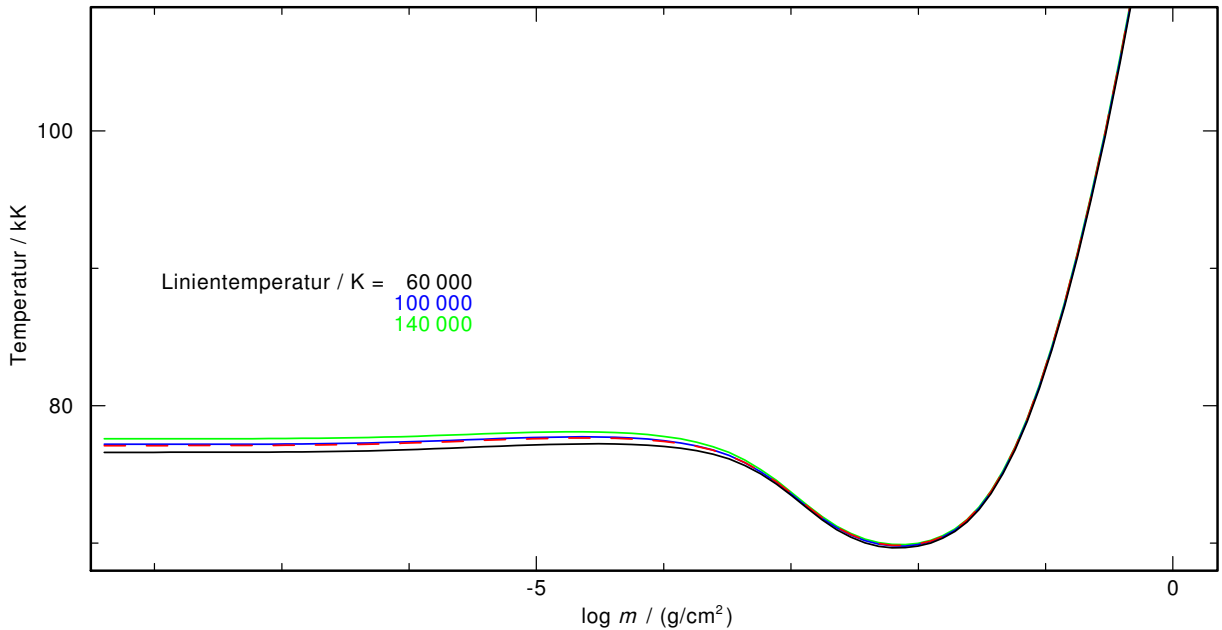


Abbildung 2.8.: Temperaturstrukturen der Modellatmosphären mit H und Fe bei der Verwendung unterschiedlicher Linientemperaturen. Gestrichelt dargestellt ist das Modell mit den auf  $T_L = 100\,000$  K interpolierten Querschnitten.

mittleren Temperatur des Linienentstehungsgebiets entspricht) liegt (Rauch & Werner, 1988). Für Objekte mit unterschiedlichen Effektivtemperaturen müssen die Querschnitte neu berechnet werden. Sollte der Einfluss der Linientemperatur jedoch gering sein, so könnte diese Neuberechnung vermieden werden. Deshalb wurde ein Test durchgeführt, für den Querschnitte bei drei verschiedenen  $T_L$  (60 000 K, 100 000 K und 140 000 K) berechnet wurden. Anhand dieser Querschnitte wurde jeweils eine Modellatmosphärenrechnung für die Elemente H und Fe mit solaren Häufigkeiten,  $T_{\text{eff}} = 100\,000$  K und  $\log g = 7.0$  und einem Frequenzgitter mit etwa 22 000 Punkten erzeugt. Die Modelle wurden so lange gerechnet, bis die niedrigsten absoluten Korrekturen  $10^{-6}$  unterschritten, was einen Faktor hundert niedriger als das von *TMAP* verwendete Konvergenzkriterium ist. Die resultierenden Temperaturstrukturen unterscheiden sich maximal um 1034 K in der äußeren Atmosphäre (Abb. 2.8). Die Kontinua der Flüsse sind in den Bereichen, die nicht von Linien dominiert sind, nahezu identisch. Eine Abweichung der Linienstärken ist jedoch zu sehen (Abb. 2.9). Eine einheitlich gewählte Linientemperatur kann somit (falls  $3/4 T_{\text{eff}}$  nicht der  $T_L$  der Querschnitte entspricht) nicht für eine zuverlässige Spektralanalyse verwendet werden. Eine Interpolation über ein angemessenes Intervall zwischen Querschnitten ist jedoch eine Option, die zu großer Zeitersparnis führt. Um die Abweichungen durch eine Interpolation abzuschätzen wurde zwischen den Querschnitten für 140 000 K und 60 000 K auf 100 000 K interpoliert. Mit den interpolierten Querschnitten wurde eine Modellatmosphäre berechnet und das Ergebnis mit der für  $T_L = 100\,000$  K berechneten Modellatmosphäre verglichen (Abb. 2.8). Der maximale Unterschied in der Temperaturstruktur liegt bei 93 K. Eine Interpolation ist also durchaus eine sehr gute Näherung. Um die Abweichungen noch weiter zu reduzieren wird für den neuen *TIRO*-Dienst eine Datenbank aufgebaut für unterschiedliche Werte der  $T_L$ , die sich um 20 000 K anstelle der 40 000 K dieses Tests unterscheiden.

Mit dem Modell mit  $T_L = 100\,000$  K wurde in einem weiteren Test der Einfluss der Zahl der Frequenzpunkte untersucht. Dafür wurden drei Frequenzgitter erzeugt, die – zusätzlich zu einem groben Grundfrequenzgitter – zwischen  $95 \text{ \AA}$  und  $400 \text{ \AA}$  alle  $0.100$ ,  $0.010$  bzw.  $0.001 \text{ \AA}$  Frequenzpunkte haben. Mit diesen Frequenzgittern wurden analog zum vorherigen Test Querschnitte und HFe-Modellatmosphären berechnet. Die resultierenden Temperaturstrukturen unterscheiden sich maximal um 2100 K in der äußeren Atmosphäre ( $\log m < -3$ , Abb. 2.10), der Unterschied zwischen dem Modell mit  $0.010 \text{ \AA}$ - und

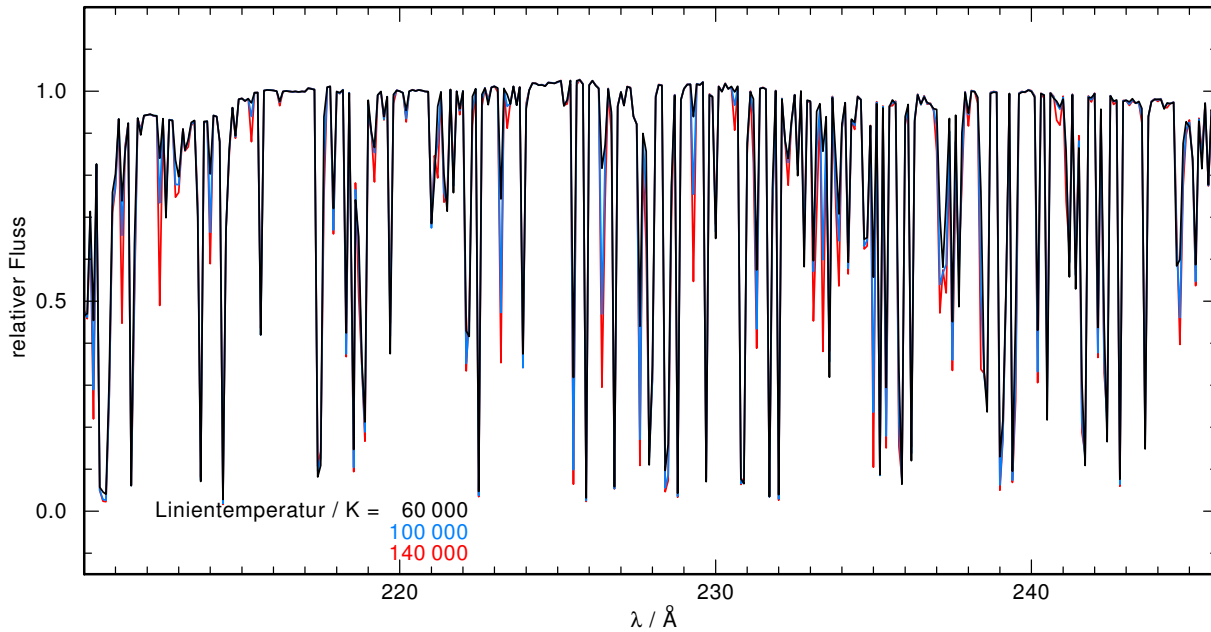


Abbildung 2.9.: EUV-Fluss von HFe-Modellen in Abhängigkeit von  $T_L$ .

0.001 Å-Frequenzpunktabstand ist jedoch kaum zu erkennen. Der EUV-Flussverlauf der Modelle zeigt keine signifikanten Unterschiede im Kontinuumsverlauf des Modells mit 0.010 Å und mit 0.001 Å Frequenzpunktabstand (Abb. 2.11). Im Fall des 0.100 Å-Modells ist der Flussverlauf allerdings deutlich verändert. Die Auswirkungen der reduzierten Frequenzpunktzahl sind für das Modell mit 0.100 Å-Frequenzpunktabstand insgesamt am stärksten. Beim Modell mit 0.010 Å-Frequenzpunktabstand sind vor allem Auswirkungen auf die Linienstärken zu erkennen. Für die Berechnung der Querschnitte (und Modellatmosphäre) sollte deshalb ein sehr feines Gitter verwendet werden. *IronIc* verwendet in der neuen Version ein Frequenzgitter, das eine Million Punkte enthält, die logarithmisch äquidistant verteilt sind.

### 2.3.6. Das Intervall zur Berechnung der Voigtprofile

Wie bereits dargestellt (Kapitel 2.3.2), benötigt die Berechnung der Voigtprofile der RBB-Querschnitte die meiste Rechenzeit des *IronIc*-Programms. Dabei werden die Profile nicht über das gesamte Frequenzgitter berechnet, sondern nur für Punkte mit einem gewissen Abstand zum Linienzentrum ( $(\nu - \nu_0) < v_{\max}$ , wobei  $\nu_0$  die Zentralfrequenz der betrachteten Linie und  $v_{\max}$  der Abstand des Frequenzpunkts zur Linienmitte in Dopplerbreiten ist). In der alten *IronIc*-Version war  $v_{\max}$  zwar variabel, wurde jedoch in der Regel auf  $10^{15}$  gesetzt. Da beim Kompilieren des Programms bis zu dieser Arbeit eine falsche Kompileroption verwendet wurde, trat ein Fehler beim Einlesen von  $v_{\max}$  auf und anstelle des gewünschten Werts wurde ein deutlich kleinerer ( $1.2 \times 10^9$ ) verwendet. Wird mit der richtigen Option kompiliert, kann der korrekte Wert eingelesen werden. Die Größe dieses Werts hat sowohl Einfluss auf die Genauigkeit der resultierenden, gesampelten Querschnitte als auch auf die Rechenzeit. Deshalb wurde der Einfluss dieses Werts in einer Testrechnung genauer untersucht. Dafür wurden Querschnitte für ein Fe VII-Modellatom mit  $T_L = 100\,000$  K, einem Frequenzgitter mit 1 000 000 Punkten und sieben Superlevels berechnet. Der Wert von  $v_{\max}$  wurde dabei ausgehend von 100 jeweils um einen Faktor zehn erhöht bis der ursprüngliche Wert erreicht war.

Dieser Test zeigt, dass die Berechnungszeiten bis zu  $v_{\max} = 1 \times 10^8$  ansteigen (Tabelle 2.3). Ab diesem Wert ist kein eindeutiger Trend mehr festzustellen. Die Abweichungen der resultierenden Querschnittswerte zu den mit maximalem  $v_{\max}$  berechneten nehmen bis zu  $v_{\max} = 10^8$  ab, wobei ab  $v_{\max} = 10^6$  nur noch wenige RBB-Dateien Unterschiede zeigen. Abb. 2.12 zeigt, dass bereits die Querschnitte berechnet



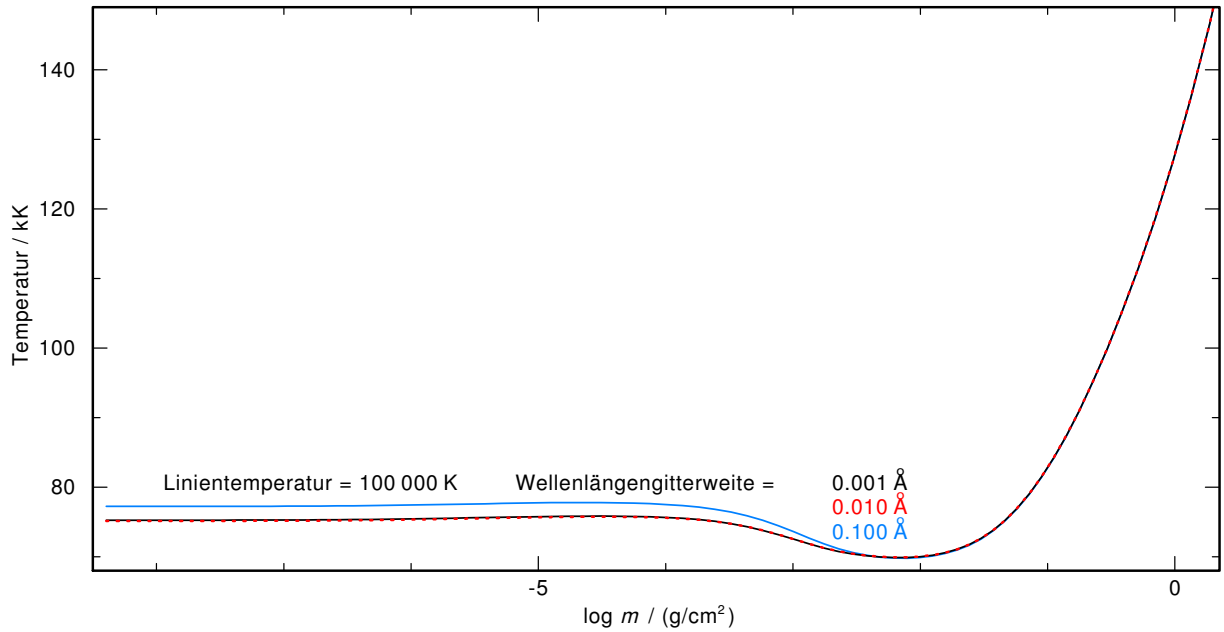


Abbildung 2.10.: Temperaturstrukturen der HFe-Modellatmosphären bei der Verwendung dreier unterschiedlicher Frequenzgitter.

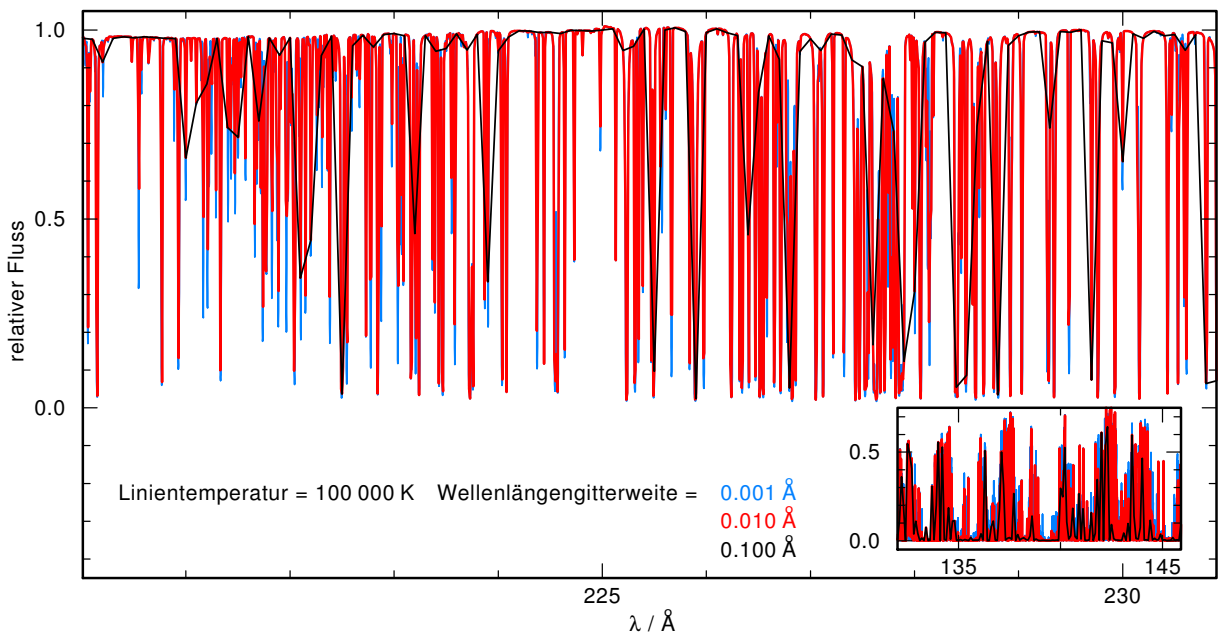


Abbildung 2.11.: Ausschnitt des EUV-Flusses der HFe-Modellatmosphäre um das Flussmaximum und im Bereich der größten Abweichungen (kleine Abbildung) bei der Verwendung dreier unterschiedlicher Frequenzgitter.

mit  $v_{\max} = 10^5$  nur innerhalb der Linienstärke der Darstellung abweichen, während für  $v_{\max} = 10^3$  deutliche Abweichungen zu erkennen sind. Für das getestete Ion wäre  $v_{\max} = 10^9$  groß genug. Da die Rechenzeiten für größere Werte von  $v_{\max} > 10^8$  nicht mehr eindeutig ansteigen, wirkt sich ein größerer Wert nicht negativ auf die Rechenzeiten aus. Um  $v_{\max}$  für alle Elemente und Ionen einheitlich zu setzen, wird der minimale Wert ermittelt, bei dem alle Frequenzpunkte des Gitters zur Berechnung eines Voigtprofils berücksichtigt werden. Es gilt  $\nu - \nu_0 < v_{\max} \Delta\nu_D$ , wobei  $\nu - \nu_0 < 10^{18}$  Hz. Ein minimaler

Tabelle 2.3.: Berechnungsdauer der gebunden-gebunden-Querschnitte von Eisengruppenelementen für unterschiedliche  $v_{\max}$ -Werte.

$\log v_{\max}$	Rechenzeit / sec	max. Abweichung in %
2	1019.4	>100
3	1061.9	>100
4	1492.1	>100
5	12690.2	5.54
6	13041.2	$1.87 \times 10^{-3}$
7	13079.5	$1.15 \times 10^{-6}$
8	13128.4	
9	13113.7	
10	13121.3	
11	13123.5	
12	13056.4	
13	13107.2	
14	13124.6	
15	13107.7	

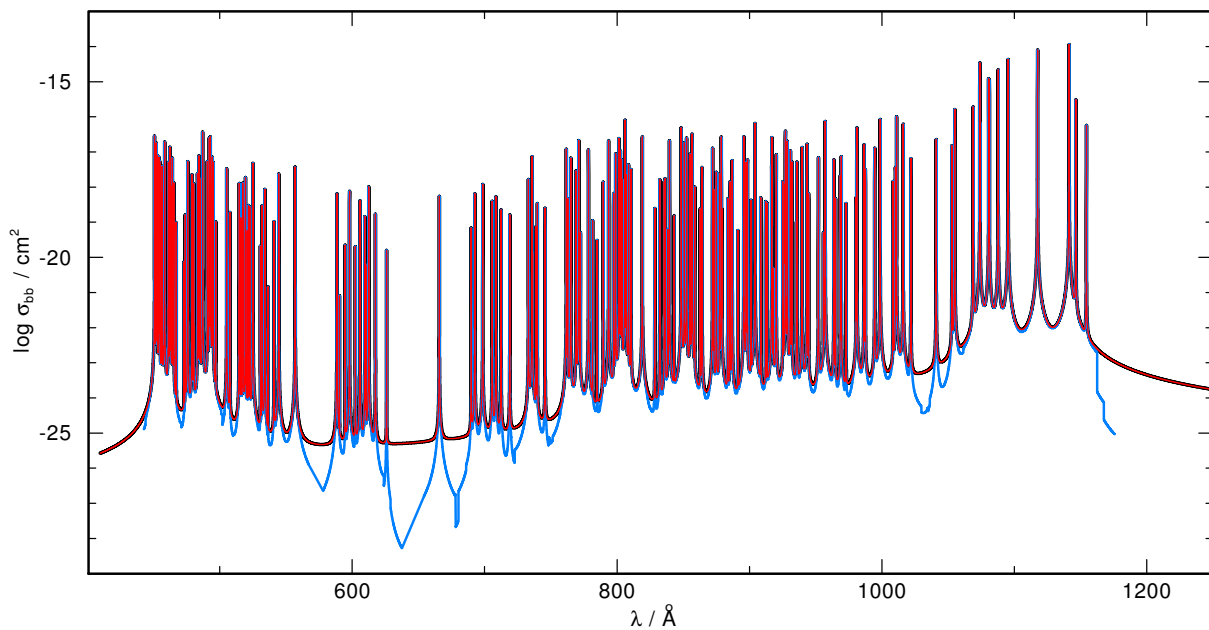


Abbildung 2.12.: Vergleich von Querschnitten (Elektronendichte 0) von Fe VII von Band zwei nach vier mit  $(\nu - \nu_0)/\Delta\nu_D < 10^{15}$  (schwarz), 100 000 (rot) und 1000 (blau).

$v_{\max}$ -Wert, bei dem alle Frequenzpunkte berücksichtigt werden, ist also  $10^{18}/\Delta\nu_D$ , für eine minimale Dopplerbreite ( $\approx 10^6$  bei  $T_L = 10\,000$  K). Also werden mit  $v_{\max} = 10^{12}$  die Voigtprofile auf dem gesamten Frequenzgitter berechnet. Da die Abweichungen aber bereits für  $v_{\max} = 10^9$  klein genug sind, wird in der neuen *IronIc*-Version  $v_{\max}$  auf  $10^9$  gesetzt.

## 2.4. Der TIRO-Dienst

Nach diesen Tests, durch die die Berechnung der RBB- und RBF-Querschnitte der Eisengruppenelemente verbessert und beschleunigt werden kann, wird *IrOnIc* dem VO-Nutzer als *TIRO*-Dienst zugänglich gemacht. Diese Umstellung erfolgte in drei Schritten.

Der erste Ansatz zur Umstrukturierung der Bedienung des *IrOnIc*-Codes war, die drei zur Erzeugung der Querschnitte nötigen Programmdurchläufe in einen einzigen Schritt umzuwandeln. Dafür wurde ein Verfahren eingeführt, bei dem das *IrOnIc*-Programm vom VO-Nutzer über eine vom Tübinger GAVO-Team entwickelte Webseite<sup>6</sup> bedient werden kann (Abb. 2.13). Von dieser Webseite wird eine Datei erzeugt, in der die vom Nutzer angeforderten Parameter (Linientemperatur, gewünschte Elemente und Häufigkeiten, Art der Ionen und der zugrunde liegenden Atomdaten) gespeichert werden.

Abbildung 2.13.: *TIRO*-Webseite.

Die neue *IrOnIc*-Version wird mit Hilfe von Unix-Skripten (in `/home/TIRO/scripts`) und FORTRAN-Programmen (in `/home/TIRO/FORTRAN`) gesteuert. Ein Überblick über die einzelnen Schritte, die vom neuen *TIRO*-Dienst durchgeführt werden, ist in Abb. 2.14 dargestellt.

Für den Start der *IrOnIc*-Berechnung ist die Unix-Prozedur `checkrun.bat` zuständig. Sie wird alle fünf Minuten als Cron-Job auf dem *GAVO* @IAAT-Server gestartet und prüft zuerst, ob genug Rechenkapazität frei ist und ob kein anderer *TIRO*-Request zur selben Zeit abgearbeitet wird. Sind alle Anforderungen erfüllt, so wird die Prozedur `run_ions.bat` gestartet. Durch diese wird die älteste, noch nicht bearbeitete Request-Datei eingelesen. Ein Unterverzeichnis, benannt nach dem Datum des Requests, wird im Verzeichnis `/home/TIRO/requests` erstellt, in dem alle zur Steuerung nötigen, sowie die resultierenden Dateien erzeugt werden. Nach dem Einlesen des Requests wird der Nutzer per Email benachrichtigt, dass mit der Bearbeitung seines Requests begonnen wurde. Mit `create_infile.Linux_x64` werden dann zuerst die zur Berechnung von  $T^*$  nötigen Steuerdateien, die die vom Nutzer gewünschten Parameter enthalten, für jedes gewählte Element erstellt. Dann wird *IrOnIc* mit den eben erzeugten Steuerdateien ausgeführt. Die resultierenden  $T^*$ -Werte werden von

<sup>6</sup><http://astro.uni-tuebingen.de/~TIRO>

`create_infile_single.Linux_x64` eingelesen und neue Steuerdateien – dieses Mal für jede Ionisationsstufe jedes Elements – werden zusammen mit `create_jobfile_single.Linux_x64` erzeugt. Mit der Prozedur `startjob.bat` werden diese Steuerdateien und damit die Berechnung der Querschnitte und Erzeugung der Modellatome gestartet. Sind die Berechnungen für alle Ionisationsstufen aller Elemente fertig, werden die Atomdaten mit den beiden bereits vor dieser Arbeit existierenden Prozeduren `ironic_combine_web.job` zusammengesetzt und die Querschnittsdateien mit `ChangeIrOnIcHeaders` in das von *TMAP* benötigte Format gebracht. Die resultierenden Atomdaten und Querschnitte werden in ein komprimiertes tar-Archiv gepackt und der Nutzer wird informiert, über welche URL er diese herunterladen kann. Ist während der Berechnung ein Fehler aufgetreten, die Prozedur unvorhergesehen abgebrochen oder wurden keine Querschnittsdateien erzeugt, wird der *TIRO*-Administrator per Email informiert.

Der frühere zweite Berechnungsschritt wird bei diesem Verfahren nicht mehr durchgeführt und viel Arbeit, die zuvor von Hand erledigt wurde (wie z.B. der Start der einzelnen Programmdateien für jedes Element auf einem anderen Rechner oder die Änderung der Steuerdateien), wird nun automatisch ausgeführt.

Ein zweiter Schritt zur Umstrukturierung war, die *IrOnIc*-Berechnungen unabhängig von den einzelnen Frequenzgittern der Modellatmosphärenberechnung zu machen. Vor dieser Arbeit wurden die Querschnitte für das Frequenzgitter berechnet, welches anschließend für die Modellrechnung verwendet wurde. Eine Änderung des Frequenzgitters, z.B. weil ein neues Element in die Modellrechnung mit aufgenommen wurde, weil mehr NLTE-Levels eines bereits vorhandenen Elements berücksichtigt werden sollten oder weil die Frequenzpunkte enger gewählt wurden, hatte eine Neuberechnung aller Querschnittsdaten zur Folge. Mit der neuen Handhabung des *IrOnIc*-Programms werden die Querschnitte auf einem sehr feinen, in  $\log \nu$  äquidistanten Frequenzgitter mit etwa 1 000 000 Punkten berechnet. Basierend auf diesem Frequenzgitter wurde eine Datenbank für Querschnitte angelegt, die Dateien für generische Modellatome mit den Elementen Ca-Ni und mit den Elementen Ca, Sc, Ti, V, Cr, Mn und Co, jeweils mit solaren relativen Häufigkeitsverhältnissen enthält. Zusätzlich wurden Querschnitte für alle Eisengruppenelemente individuell erzeugt. Diese Berechnungen wurden jeweils für  $T_L$  zwischen 20 000 K und 200 000 K (im Abstand von jeweils 20 000 K) für LIN-Daten durchgeführt.

Beim Absenden eines Requests über die *TIRO*-Webseite wird nun geprüft, ob die gewünschten Querschnittsdaten neu erzeugt werden müssen oder ob zwischen den Werten der Datenbank interpoliert werden kann. Diese Interpolation erfolgt sowohl auf die gewählte Linientemperatur als auch auf das Frequenzgitter. Dafür wurde die *TIRO*-Webseite im Rahmen dieser Arbeit um einen Frequenzgitter-Upload erweitert. Alternativ dazu kann der Nutzer ein Frequenzgitter für einen selbst gewählten Wellenlängenbereich mit gewünschter Feinheit erzeugen lassen. Die Interpolation der Querschnitte erfolgt auf das vom Nutzer hochgeladene oder ausgewählte Frequenzgitter. Auf diese Art werden einige Neuberechnungen, die während beinahe jeder Analyse nötig sind, vermieden. Können die gewünschten Querschnitte nicht durch Interpolation erzeugt werden, so werden sie neu berechnet. Dafür wird allerdings nicht das Frequenzgitter mit etwa 1 000 000 Punkten verwendet, sondern das vom Nutzer gewählte und auf ca. 200 000 Punkte beschränkte.

Für die Umsetzung wurde die Prozedur `run_inter.sh` erzeugt. Analog zu `run_ions.bat` liest sie zuerst die vom Nutzer gewählten Daten ein und erzeugt ein nach dem Datum benanntes Arbeitsverzeichnis. Dann wird das vom Nutzer hochgeladene Frequenzgitter in dieses Verzeichnis verschoben oder ein neues Frequenzgitter mit dem SETF2-Programm von *TMAP* oder mit `fgrid_user.Linux_x64` erzeugt. Falls dabei ein Fehler auftritt, wird der *TIRO*-Administrator per Email informiert. Dann wird die Prozedur `interpol.sh` gestartet, die eine mögliche Interpolation zwischen Querschnitten aus der Datenbank auf die gewünschten Parameter prüft und gegebenenfalls durchführt. Ist keine Interpolation möglich, werden die Querschnitte analog zu `run_ions.bat` auf dem vom Nutzer gewählten Frequenzgitter erzeugt und die resultierenden Daten in ein mit `wget` zugängliches Verzeichnis verschoben.

Die Prozedur `interpol.sh` bestimmt zuerst den Temperaturbereich, in dem die gewählte Linientemperatur liegt. Ist dieser Bereich von der Datenbank abgedeckt, wird geprüft, ob das gewünschte

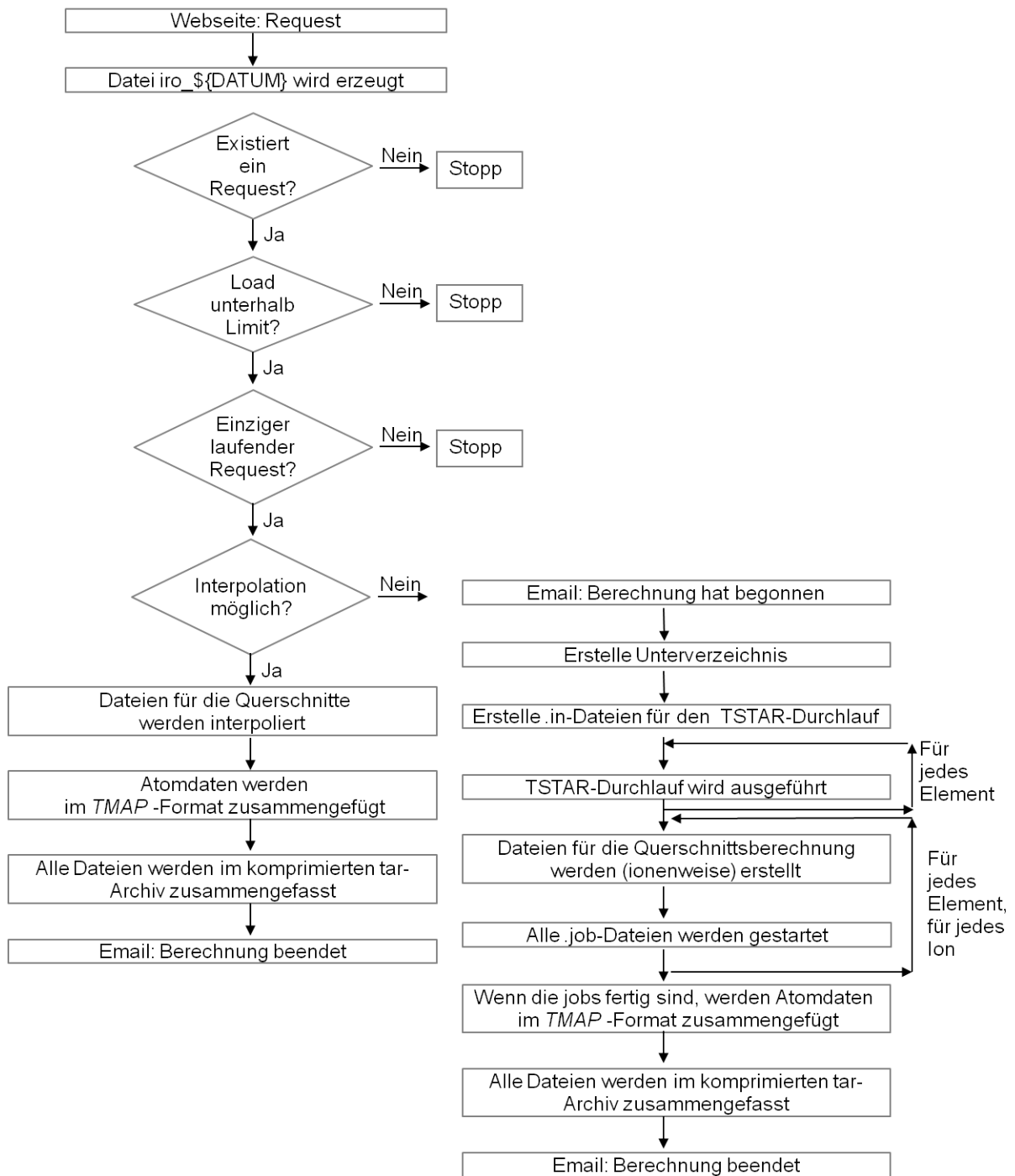


Abbildung 2.14.: Schematische Übersicht des *TIRO*-Ablaufs.

Modellatom alle Elemente individuell, alle Elemente generisch oder alle Elemente außer Fe und Ni generisch berücksichtigen soll und ob das Modellatom auf LIN-Linien basiert sein soll. Für generische Modellatome wird außerdem mit `check.Linux_x64` geprüft, ob die gewählten Häufigkeitsverhältnisse solar sind. Für die Interpolation wird das Programm `Interpol.Linux_x64` verwendet. Analog zur Prozedur `run_ions.bat` werden dann die Atomdaten zusammengefasst, das Format der Querschnitte geändert und die Daten in ein mit `wget` zugängliches Verzeichnis verschoben, über dessen URL der Nutzer per Email informiert wird.

Die Standardprozedur zur Berechnung der Querschnitte ist nun nicht mehr `run_ions.bat`, sondern

run\_inter.sh. Dem *TIRO*-Administrator steht allerdings die Prozedur run\_ions.bat weiterhin zur Verfügung, um beispielsweise die Datenbank zu erweitern.

Der dritte Schritt war, die Berechnung der Querschnitte selbst zu beschleunigen. Mit der Steuerung über die Webseite wurde auch eingeführt, dass die Querschnitte für die einzelnen Ionen – und nicht nur, wie zuvor, die einzelnen Elemente – parallel berechnet werden. Für einen Test, ob die parallele Berechnung der Ionen einen Unterschied zur seriellen bewirkt, wurde ein Eisenmodellatom für die Ionisationsstufen IV-VI mit jeweils sieben Bändern auf einem Frequenzgitter mit etwa 1 000 000 Punkten auf beide Arten berechnet. Zwischen den resultierenden Dateien konnte kein Unterschied festgestellt werden. In diesem Test brauchte die parallele Berechnung 159 734 Sekunden, die serielle 343 648 Sekunden. Das neue Verfahren macht die Berechnung also um etwa einen Faktor zwei schneller.

Um das Programm weiter zu beschleunigen, wurde, wie bereits beschrieben, die schnellste Voigtfunktion (Hellwege, 1965) implementiert. Dies führte im ersten Schnelligkeitstest zu einer Beschleunigung um etwa 10%.

Wie zum Teil bereits erwähnt, verfügt der neue *TIRO*-Dienst über mehrere Kontrollfunktionen, die den *TIRO*-Administrator über nicht funktionierende Rechnungen bzw. eventuell auftretende Fehler bei der Berechnung informiert. Die möglichen Fehlerquellen, auf die hauptsächlich geachtet wird, sind:

- Fehlerhafte Steuerdateien. Diese werden automatisch erstellt und können durch falsche Eingaben des *TIRO*-Nutzers verursacht werden. Um solche Fehler zu vermeiden, wurden (zum Teil von Rauch) in die Webseite einzelne Abfragen eingebaut, die das Absenden eines Requests nur für sinnvoll ausgefüllte Eingabefelder und hochzuladende Dateien ermöglichen.
- Das hochgeladene, bzw. nach Nutzervorgaben durch *TIRO* erzeugte Frequenzgitter.
- Fehler bei der Ausführung der Interpolationsroutine.
- Die Berechnung der Querschnitte bzw. die Ausführung des *IrOnIc*-Programms.
- Die Atomdaten. Wird z.B. eine Datei durch eine von Kurucz neu berechnete ersetzt, können Formatfehler auftreten.
- Die resultierenden Dateien. Sind diese nicht vorhanden, kann es sein, dass sich wichtige Einstellungen auf dem Rechner geändert haben oder das *IrOnIc*-FORTRAN-Programm nicht mehr zugänglich oder ausführbar ist.

Insgesamt ist der *TIRO*-Dienst so auf einen reibungslosen Ablauf vorbereitet und der *TIRO*-Administrator wird direkt auf eventuell auftretende Schwierigkeiten hingewiesen. Eine lange Fehlersuche wird so ebenso wie eine aufwändige Kontrolle von Hand vermieden.

Mit diesem Dienst wurde ein wichtiger Beitrag zur Spektralanalyse über das *VO* geleistet. Wie wichtig die Berücksichtigung der Eisengruppenelemente in der Spektralanalyse ist, wird noch einmal in Kapitel 4 deutlich gemacht.

## 3. Das Tübinger GAVO-Projekt und seine Weiterentwicklung

Die grundlegenden Gleichungen eines Sternatmosphärencodes sind, wie in Kapitel 1 beschrieben, die Strahlungstransportgleichung, die statistischen Gleichungen und die Gleichungen für die Nebenbedingungen. Ihre Lösung ist im NLTE-Fall sehr komplex und erfordert ausgeklügelte Programme, die meist nur wenigen Astronomen zur Verfügung stehen. Um realistische Modelle für eine Sternatmosphäre zu erhalten, sind außer den physikalischen Gleichungen auch möglichst große bzw. vollständige, präzise Modellatome von großer Wichtigkeit. Dies bezieht sich nicht nur auf die Anzahl der Elemente. Es sollten auch ausreichend viele Ionisationsstufen, atomare Levels und Übergänge berücksichtigt werden, um alle Effekte auf die Atmosphäre modellieren zu können. Zuverlässige Modelle sollten außerdem viele Iterationsschritte durchlaufen haben. Vor allem für Sternparameter im NLTE-Bereich sind daher ausreichende Rechnerressourcen und Rechenzeiten für die Erstellung vertrauenswürdiger Modelle nötig. Aus diesem Grund galt die NLTE-Spektralanalyse lange Zeit als Gebiet der Spezialisten.

Für die Modellierung von NLTE-Spektren werden weltweit mehrere, zum Teil unterschiedliche Programme verwendet. Die Tübinger Arbeitsgruppe verwendet das selbst entwickelte Programm *TMAP*. Ein weiterer Modellatmosphärencode ist *TLUSTY*<sup>1</sup>, bei dem die selben Gleichungen gelöst werden, aber z.B. andere Atomdaten zugrunde liegen und ein anderes Konvergenzkriterium verwendet wird. Mit dem NLTE-Code *PHOENIX*<sup>2</sup> können beispielsweise expandierende Atmosphären zusätzlich zu den statischen simuliert werden. Für Neueinsteiger in dieses Gebiet der Spektralanalyse ist es unter Umständen schwer, sich in die Programme einzuarbeiten oder sie überhaupt zu bekommen. Astronomen, die sich nicht direkt mit der Spektralanalyse beschäftigen, Spektren aber trotzdem für einen Teil ihrer Arbeit benötigen, schreckt der große (Zeit-)Aufwand für die Berechnung zuverlässiger Modellatmosphären oft ab. So wurden z.B. für die Modellierung Planetarischer Nebel lange Zeit Schwarzkörperflüsse anstelle realistischer, theoretischer Spektren als Input für die Photoionisationsmodelle verwendet.

Um die NLTE-Spektralanalyse für jedermann möglich zu machen, wurde der Dienst *TheoSSA* entwickelt, der in der Anfangsphase auf dem Tübinger Programm zur Modellierung von Sternatmosphären (Kapitel 3.1) basiert. Mit *TheoSSA* können zum ersten Mal sowohl Hobbyastronomen ihre selbstaufgenommenen Spektren analysieren, als auch Spezialisten eine schnelle, erste Analyse durchführen, ohne dabei viel Rechenzeit investieren zu müssen. *TheoSSA* ist intuitiv zu bedienen, erfordert keinerlei Hintergrundkenntnisse und liefert schnell Ergebnisse. Damit der Nutzer abschätzen kann, wofür genau der Dienst geeignet ist, wurden eigene Tests und eine Qualitätskontrolle eingeführt (Kapitel 3.5, 4).

Im Folgenden werden die verschiedenen Möglichkeiten beschrieben, mit denen die Spektralanalyse über das Internet im Rahmen des Tübinger GAVO-Projekts realisiert wurde. Anschließend wird auf die Einführung einer Qualitätskontrolle für diese Dienste eingegangen und die Benchmarktests vorgestellt.

### 3.1. *TMAP* – Das Tübinger Modellatmosphärenpaket

Das Tübinger NLTE Model-Atmosphere Package (*TMAP*, Werner, 1986; Werner et al., 2003; Rauch & Deetjen, 2003) ist eine Sammlung von FORTRAN-Programmen, die in den 1980er Jahren entwickelt wurden und seither ständig verbessert und erweitert werden. Mit diesen Programmen wird das in Kapitel 1 beschriebene gekoppelte, hochgradig nicht-lineare Gleichungssystem für die Strahlungstransport-

<sup>1</sup><http://nova.astro.umd.edu/>

<sup>2</sup><http://www.hs.uni-hamburg.de/EN/For/ThA/phoenix/>

gleichung und die statistischen Gleichungen zusammen mit den Gleichungen für die Nebenbedingungen gelöst. Zusätzliche Annahmen des Codes sind chemische Homogenität der Atmosphäre und planparallele Geometrie. Alternativ können auch geschichtete Atmosphären oder sphärische Geometrie angenommen werden.

Für die Lösung des Gleichungssystems wird das Verfahren der beschleunigten Lambda-Iteration (accelerated lambda iteration, ALI, Cannon, 1973; Scharmer, 1981; Werner & Husfeld, 1985) verwendet. Bei diesem Verfahren wird in die formale Lösung der Strahlungstransportgleichung  $J = \Lambda S$  ein genäherter  $\Lambda$ -Operator  $\Lambda^*$  eingefügt

$$J^n = \Lambda^* S^n + (\Lambda - \Lambda^*) S^{n-1},$$

wobei der Korrekturterm  $(\Lambda - \Lambda^*) S^{n-1}$  aus dem vorherigen Rechenschritt  $n - 1$  bekannt ist und (in dieser Arbeit) ein diagonalen  $\Lambda^*$ -Operator verwendet wird. Die Gleichung für das Strahlungsfeld hängt damit nicht mehr von der aktuellen Quellfunktion ab und kann unabhängig von den Gleichungen der Nebenbedingungen (innerer Iterationszyklus) in einem äußeren Zyklus gelöst werden. Dieser äußere Iterationszyklus wird so lange wiederholt bis Konvergenz ( $S^n = S^{n-1}$ ) erreicht ist. Der innere Zyklus wird mit dem Newton-Raphson-Verfahren gelöst, wobei in jedem Schritt die Jacobi-Matrix aufgestellt und invertiert werden muss. Alternativ dazu kann die Kantorovich-Variante dieses Verfahrens verwendet werden, wobei die Jacobi-Matrix nicht mehr in jeder Iteration<sup>3</sup> invertiert wird, was eine enorme Beschleunigung der Berechnung zur Folge hat.

*TMAP* kann für zuverlässige, präzise Spektralanalysen von Objekten mit Parametern aus dem gesamten NLTE-Bereich (Abb. 1.7) verwendet werden. Mittlerweile werden nicht nur Weiße Zwerge oder deren Vorläufer modelliert (typischerweise liegen die Parameter im Bereich  $20\,000 \lesssim T_{\text{eff}} \lesssim 200\,000\text{K}$  und  $4 \lesssim \log g \lesssim 9$ ), sondern auch andere Objekte, wie z.B. unterleuchtkräftige Sterne, Weiße Zwerge in Novae oder Neutronensterne (mit  $T_{\text{eff}}$  bis zu einigen Millionen K und mit bis zu  $\log g = 15$ ).

Für die Berechnung einer Modellatmosphäre können Elemente von H bis Xe mit bis zu 2000 Levels und 4000 Linien berücksichtigt werden. Die Daten dafür (Levelenergien, Querschnitte für Übergänge, etc.) werden der speziell für *TMAP* angelegten Modellatomdatenbank *TMAD* (Kapitel 3.2) entnommen. Diese Datenbank wird ständig überarbeitet, wodurch die Modellatome, die fundamentaler Input für alle Modellatmosphärenrechnungen sind, auf dem neuesten Stand sind.

### 3.2. *TMAD* – Eine Modellatomdatenbank

Die Genauigkeit der von einem Sternatmosphärenprogramm verwendeten Atomdaten hat große Auswirkungen auf die Atmosphärenstruktur und auf die Präzision der Wellenlänge und Stärke der Absorptions- bzw. Emissionslinien des resultierenden Spektrums. Die Ergebnisse bzw. Zuverlässigkeit der Spektralanalyse insgesamt wird also stark von der Qualität der eingehenden Atomdaten bestimmt. Deshalb ist es besonders wichtig, diese Daten immer auf dem neuesten Stand zu halten und alle für die Atmosphärenstruktur wichtigen chemischen Elemente in die Berechnungen mit einzubeziehen.

Die Tübinger Modellatomdatenbank *TMAD* wird aus diesem Grund regelmäßig aktualisiert und erweitert. Die enthaltenen Daten haben ein Format, das von *TMAP* verwendet wird und sind auf den Standardquellen wie z.B. NIST<sup>4</sup> oder dem Opacity Project<sup>5</sup> (Seaton et al., 1994) aufgebaut. Um diese Atomdaten allen zur Verfügung zu stellen, die an Spektralanalysen interessiert sind, wurde die *TMAD*-Datenbank im Rahmen des *GAVO*-Projekts veröffentlicht (Abb. 3.1). Seither sind alle verfügbaren Atomdaten frei über das Internet zugänglich und können direkt heruntergeladen werden.

*TMAD* enthält für jedes Element grundsätzlich zwei Arten von Modellatomen (Abb. 3.2), die einen sind für die Atmosphärenberechnung, die anderen für die Berechnung synthetischer Spektren vorgese-

<sup>3</sup>Die Standardeinstellung sieht eine Invertierung der Jacobi-Matrix jede fünfte Iteration vor, der *TMAP*-Nutzer kann dies aber individuell wählen.

<sup>4</sup><http://www.nist.gov/pml/data/asd.cfm>

<sup>5</sup><http://cdsweb.u-strasbg.fr/topbase/TheOP.html>



Abbildung 3.1.: Im oberen Teil der *TMAD*-Webseite ist ein Überblick über die verfügbaren Elemente sowie einleitende Informationen über die Datenbank dargestellt.

hen. Für die letzteren wird zusätzlich die Feinstrukturaufspaltung der Levels berücksichtigt. Die Modellatome können individuell aus den einzelnen Modellionen der Elemente aufgebaut bzw. angepasst werden. Für die Atmosphärenberechnung sind zusätzlich Standardmodellatome für bestimmte Temperaturbereiche verfügbar, die aus den jeweils für diesen Bereich wichtigen Ionen bestehen. Je nach Fachkenntnis kann der *VO*-Nutzer diese direkt verwenden oder eigene zusammenstellen.

Im Rahmen dieser Arbeit wurden Grotriandiagramme für die Ionen der Elemente C, N, O, Ne und Mg erstellt (Abb. 3.3), um dem Benutzer einen Überblick über die Modellatome von *TMAD* zu geben. Durch diese Grotriandiagramme wird schnell deutlich, wie komplex die Modellatome sind, wie viele NLTE-Levels und wie viele Linienübergänge in den Modellatomen berücksichtigt werden.

*TMAD* beinhaltet Modellatome für die Elemente H, He, C, N, O, F, Ne, Na, Mg, Si, P, S, Ar, Ca, Ge, Kr und Xe (Abb. 3.1). Im Rahmen dieser Arbeit wurde die Datenbank um Modellionen für die Atmosphärenberechnung für

- Na III-IV
- Ar II-V

erweitert. Für Modellionen, in denen die Feinstrukturaufspaltungen berücksichtigt ist, wurden Daten für

- Ne II-V, IX-X
- Mg I-XII
- Si II-IV
- Ar IV, VI-VIII

zusammengestellt. Dies ermöglicht die Bestimmung der Häufigkeiten der Elemente. Durch die zusätzlich vorhandenen Levels und Linien für die Atmosphärenberechnung werden außerdem die Modelle realistischer.

## Neon

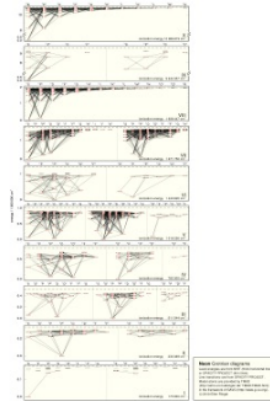
A complete model atom (NE I - XI) is provided (not processed by ATOMS2 - please read the header for the sources of the atomic data). For model-atmosphere calculations, this model atom has to be individually adjusted depending on  $T_{\text{eff}}$  and  $\log g$  (for further details and the use of ATOMS2, please have a look into the *TMAP User's Guide*). For  $\log g = 7$  (9 at  $T_{\text{eff}} = 500$  kK), examples for model ions in an approximate  $T_{\text{eff}}$  range are given (not processed by ATOMS2). These are, e.g., automatically selected for model-atmosphere calculations via *TMAW*, the *TMAP WWW* interface. The example plots (H+Ne models) show the dominant ionization stages for selected  $T_{\text{eff}}$ . (These H+Ne models were calculated for test reasons only and, thus, may not be fully converged in the outer atmospheres.)

The Ne VIII model ion was used in Werner, Rauch & Kruk 2007, *A&A*, 474, 591.

Ne model atoms for <i>TMAP</i> model-atmosphere calculation		
ions	valid for $\approx T_{\text{eff}} / \text{kK}$	example plots for ionization fractions at $T_{\text{eff}} / \text{kK}$
NE I - XI	to be adjusted	
NE I - III	< 20	20
NE I - IV	20 - 25	20
NE I - V	25 - 35	30
NE I - VI	35 - 55	40 50
NE II - VII	55 - 75	60 70
NE III - VIII	75 - 130	100
NE IV - IX	130 - 180	150
NE V - X	180 - 250	200
NE VI - X	250 - 400	300
NE VII - XI	> 400	500

Ne model ions for the calculation of the emergent spectrum (incl. fine-structure splitting)	
NE I	
NE II	
NE III	
NE IV	
NE V	
NE VI	
NE VII	
NE VIII	
NE IX	
NE X	



Ne Grotrian diagram

Abbildung 3.2.: Dieser Ausschnitt der *TMAP*-Webseite zeigt die verfügbaren Modellatome für Neon für die Berechnung der Atmosphäre sowie für synthetische Spektren. Rechts sind die Grotrian diagramme der einzelnen Neonionen dargestellt (eine hochau aufgelöste Version ist auf der Webseite verlinkt).

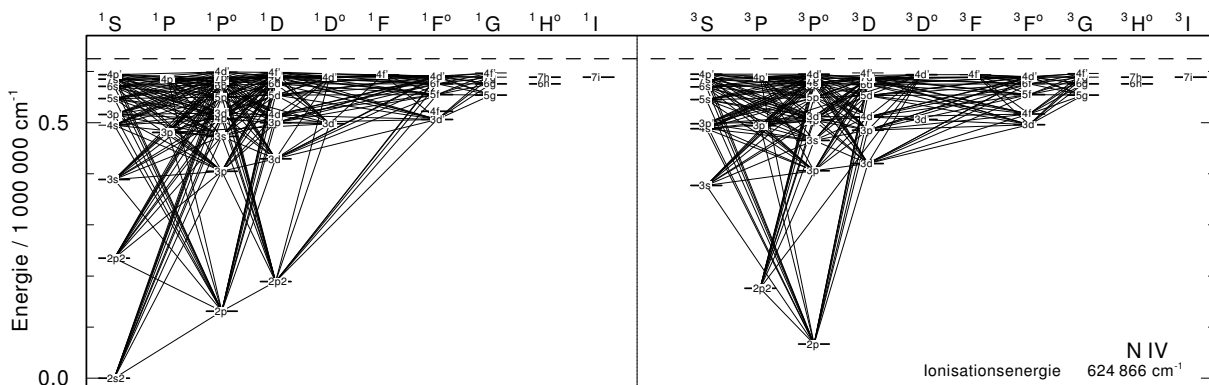


Abbildung 3.3.: Grotrian diagramm von N IV mit den Singulett- (links) und den Triplettzuständen (rechts). Die waagerechten Striche mit Levelbezeichnung stellen die Levels dar, die Linien zwischen den Levels die Übergänge.

Der Aufbau der *TMAP*-Modellatome wird durch Schlüsselworte bestimmt. Jeder einzelne der im Folgenden beschriebenen Bereiche wird mit einem eindeutigen Schlüsselwort eingeleitet. Abgeschlossen werden die Bereiche durch eine „0“ oder durch den Beginn des nächsten Bereichs.

- **Die Levels:** Um ein neues Modellatom zusammen zu stellen, werden Elektronenkonfiguration, Drehimpuls und die Energien der Levels aus den Standarddatenbanken wie NIST, dem Opacity Project, Chianti<sup>6</sup> oder der Kentucky-Datenbank<sup>7</sup> entnommen. Die Elektronenkonfigurationen und Drehimpulse werden zu einem A10-String zusammengefügt (der *TMAD*-Levelname „C402S 12S“ steht für das Element „C“ mit Ionisationsstufe „4“, Hauptquantenzahl „02“, Drehimpuls „S“, Gesamtdrehimpuls „1“ (nur bei Modellatomen, die die Feinstrukturaufspaltung berücksichtigen) und Elektronensystem „2“). Diese bilden zusammen mit dem Namen des Parent Levels (Grundzustand der nächsthöheren Ionisationsstufe, ebenfalls dargestellt als A10-String), der Ionisationsenergie des Levels in Hz und des statistischen Gewichts den ersten Teil der Modellatome. Die hier beschriebenen Informationen werden nach dem Schlüsselwort **L** oder **LTE** aufgelistet, was den Beginn der Bereiche der NLTE- bzw. LTE-Levels anzeigt.
- **Gebunden-gebunden-Übergänge:** Das Schlüsselwort für die radiativen gebunden-gebunden-Übergänge ist **RBB**. Für diese Übergänge werden hauptsächlich Opacity Project-Daten verwendet. Die beiden A10-Strings für den Namen des unteren und oberen Levels, zwischen denen der Übergang stattfindet, stehen am Anfang jedes Eintrags. Dahinter werden die Formel, mit der *TMAP* später den Übergang berechnen wird, und die Input-Daten für diese Formel gelistet, gefolgt von der Wellenlänge des Übergangs in Å.
- Die zugehörigen **gebunden-gebunden Stoß-Übergänge** (Schlüsselwort **CBB**) unterscheiden sich von den RBB-Einträgen in den von *TMAP* verwendeten Formeln zu ihrer Berechnung und deren Input (siehe <http://astro.uni-tuebingen.de/~TMAP/UserGuide/UserGuide.pdf>). Außerdem wird für diese Übergänge keine Wellenlänge angegeben.
- **Gebunden-frei-Übergänge:** Die radiativen und Stoß-Übergänge zwischen gebundenen Levels und dem Kontinuum werden durch die Schlüsselworte **RBF** bzw. **CBF** eingeleitet. Wie bei den RBB-Übergängen werden die beiden Levels, zwischen denen der Übergang stattfindet, sowie die Formel zur Berechnung des Querschnitts und die dafür nötigen Inputs angegeben. Das zweite Level ist hierbei nicht aus dem selben Ion, sondern aus dem um eins höher ionisierten.
- **Frei-frei-Übergänge und Stoßanregung:** Informationen zu **CBX**- (Stoßanregung zu LTE-Levels) und **RFF**-Übergängen sind im selben Stil wie die der anderen Übergänge angegeben und vervollständigen das Atomdatenfile.

Wurde ein neues Modellion für die Berechnung synthetischer Spektren erstellt, müssen die *TMAP*-Routinen so angepasst werden, dass die Besetzungszahlen der Levels aus der Modellrechnung (ohne Berücksichtigung der Feinstrukturaufspaltung) im richtigen Verhältnis den entsprechenden aufgespaltenen Levels zugeordnet werden können. Diese Anpassungen wurden für die oben angegebenen Ionen durchgeführt.

### 3.3. *TheoSSA* – Eine Datenbank für synthetische Sternspektren

Mit *TMAP* sind präzise und detaillierte Spektralanalysen möglich. Deshalb ist immer wieder von außerhalb der Tübinger Arbeitsgruppe angefragt worden, ob SEDs für bestimmte Objekte gerechnet und zur Verfügung gestellt oder Analysen für ausgewählte Sterne durchgeführt werden können.

Mit der Zunahme der astronomischen Daten stieg auch die Zahl der Nachfragen nach synthetischen Spektren. Die Bearbeitung jeder einzelnen Anfrage wurde zum Teil sehr zeitintensiv, was auch daran liegt, dass für jede Analyse Gitter von Spektren mit z.B. unterschiedlichen  $T_{\text{eff}}$ - und  $\log g$ -Werten anstelle einzelner SEDs nötig sind. Ein weiterer Grund für die steigende Nachfrage ist, dass vor allem

<sup>6</sup><http://www.chiantidatabase.org/>

<sup>7</sup><http://www.pa.uky.edu/~peter/atomic/>

The screenshot shows the TheoSSA Web Interface. At the top left is the GAVO logo (German Astrophysical Virtual Observatory). The main title is 'TheoSSA Web Interface'. Below the title, there is a brief description: 'TheoSSA provides spectral energy distributions based on model atmosphere calculations. Currently, we serve results obtained using the Tübingen NLTE Model Atmosphere Package for hot compact stars.' On the left side, there is a navigation menu with options like 'Help', 'Service info', 'Related', 'Compute custom SEDs', 'Metadata', and 'Try ADQL to query our data.'. The main content area contains a search form with the following fields:
 

- Effective Temperature [K]: between 100000 and 120000
- Log. Surface gravity [cm/s<sup>2</sup>]: between 6.0 and 7.0
- Mass loss rate [solMass/yr]: between [ ] and [ ]
- Mass Fraction 1: H dropdown, between 0.7 and 1.0
- Mass Fraction 2: ANY dropdown, between [ ] and [ ]
- Mass Fraction 3: ANY dropdown, between [ ] and [ ]
- Standard Stars: ANY dropdown
- Type: Radio buttons for VOTable and Text (Text is selected)
- Table: Sort by H dropdown, Limit to 100 items
- Output format: HTML dropdown, More output fields button

 A 'Go' button is at the bottom left of the form, and a '[Result link] ★' link is at the bottom right. Below the form, there is a grey box with text: 'When publishing research making use of this service, please acknowledge: "The TheoSSA service (http://dc.g-vo.org/theossa) used to retrieve theoretical spectra for this paper was constructed as part of the activities of the German Astrophysical Virtual Observatory."'

Abbildung 3.4.: TheoSSA-Webseite.

Astronomen, die synthetische Sternspektren als Input für ihr eigentliches Programm benötigen und denen deren Berechnung zu zeitintensiv ist, Schwarzkörperflüsse anstelle eines realistischen synthetischen Sternspektrums verwendeten. Mit der steigenden Genauigkeit von Programmen, die solche SEDs als Input verwenden, wird außerdem die Verwendung von Schwarzkörperspektren, die zwar einfach zu berechnen sind, aber eine schlechte Näherung für ein Sternspektrum darstellen, immer fragwürdiger und das Interesse an realistischen SEDs nimmt zu.

Diese Entwicklung macht deutlich, dass ein großer Bedarf an frei zugänglichen, bereits berechneten oder innerhalb kurzer Zeit „lieferbaren“ SEDs besteht. Eine Lösung für diese ungünstige Situation bot sich im Zusammenhang mit der Idee des Virtuellen Observatoriums. Im Rahmen des GAVO-Projekts wurde deshalb die Datenbank *TheoSSA* (Theoretical Stellar Spectra Access) eingerichtet. Sie bietet bereits berechnete, synthetische Spektren bzw. Modellgitter zum Download an. Die ursprüngliche Motivation für die Veröffentlichung von synthetischen Sternspektren war, realistische, einfach zugängliche ionisierende Spektren für Photoionisationsmodelle von Planetarischen Nebeln zur Verfügung zu stellen und auch Nicht-Spezialisten eine Spektralanalyse zu ermöglichen (Rauch, 1997, 2003, 2008).

Momentan ist die *TheoSSA*-Datenbank auf *TMAP*-Spektren basiert. Die Einbeziehung von Spektren, die mit anderen Modellatmosphärenencodes berechnet wurden, ist aber möglich und auf der Webseite bereits eingerichtet. In naher Zukunft sollen z.B. Modelle des *PoWR*-Codes<sup>8</sup>, mit dem expandierende Atmosphären simuliert werden können, über diese Datenbank zugänglich sein.

<sup>8</sup><http://www.astro.physik.uni-potsdam.de/~htodt>

**TheoSSA Web Interface**

**Parameters**

- Element 1: H
- Max Effective Temperature [K]: 120000
- Max Log. Surface gravity [cm/s<sup>2</sup>]: 7.0
- Max. M.F. 1: 1.0
- Min Effective Temperature [K]: 100000
- Min Log. Surface gravity [cm/s<sup>2</sup>]: 6.0
- Min. M.F. 1: 0.7
- Type: text/plain

**Result**  
Matched: 100

Send via SAMP | Quick Plot | Open in VOPlot

The query limit was reached. Increase it to retrieve more matches. Note that unsorted truncated queries are not reproducible (i.e., might return a different result set at a later time).

Product key	Band start [Å]	Band end [Å]	Eff. Temp. [K]	Log Grav. [cm/s <sup>2</sup> ]	Mass loss rate [solmass/yr]	H	He	C	N	O	F	Na	Mg	Al	Si	P	S	Cl	Ar	Ca	Fe	Co	Ni	Ge	Kr	Sn	Xi					
[Spectrum as txt] [in SpectroView]	5.00	2000.00	110000	6.0	0.0	0.7037	0.2794	0.003338	0.0009778	0.008871	5.28e-07	3.201e-05	0.0005366	4.732e-05	0.0007805	5.433e-06	0.0003548	4.937e-06	0.0001398	0.001259	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
[Spectrum as txt] [in SpectroView]	5.00	2000.00	120000	6.0	0.0	0.7037	0.2794	0.003338	0.0009778	0.008871	5.28e-07	3.201e-05	0.0005366	4.732e-05	0.0007805	5.433e-06	0.0003548	4.937e-06	0.0001398	0.001259	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
[Spectrum as txt] [in SpectroView]	5.00	2000.00	110000	7.0	0.0	0.7037	0.2794	0.003338	0.0009778	0.008871	5.28e-07	3.201e-05	0.0005366	4.732e-05	0.0007805	5.433e-06	0.0003548	4.937e-06	0.0001398	0.001259	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
[Spectrum as txt] [in SpectroView]	5.00	2000.00	100000	7.0	0.0	0.7037	0.2794	0.003338	0.0009778	0.008871	5.28e-07	3.201e-05	0.0005366	4.732e-05	0.0007805	5.433e-06	0.0003548	4.937e-06	0.0001398	0.001259	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
[Spectrum as txt] [in SpectroView]	5.00	2000.00	120000	7.0	0.0	0.7037	0.2794	0.003338	0.0009778	0.008871	5.28e-07	3.201e-05	0.0005366	4.732e-05	0.0007805	5.433e-06	0.0003548	4.937e-06	0.0001398	0.001259	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
[Spectrum as txt] [in SpectroView]	5.00	2000.00	100000	6.0	0.0	0.7037	0.2794	0.003338	0.0009778	0.008871	5.28e-07	3.201e-05	0.0005366	4.732e-05	0.0007805	5.433e-06	0.0003548	4.937e-06	0.0001398	0.001259	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

All abundances are given as mass fractions

In ADOL to query our data.

Please report errors and omissions to [theo@gausoftware.com](mailto:theo@gausoftware.com).  
[Privacy Policy](#) | [Terms of Use](#) | [Feedback](#)

Abbildung 3.5.: Auszug aus der Ergebnisliste einer TheoSSA-Suche.

Die Bedienung der Datenbank erfolgt nach VO-Regeln intuitiv – nur die fundamentalen Parameter ( $T_{\text{eff}}$ ,  $\log g$  und Häufigkeiten einzelner Elemente) werden auf der Webseite angegeben (Abb. 3.4). Als Ergebnis wird eine Liste aller verfügbaren SEDs in einem gewissen, voreingestellten Parameterbereich angezeigt (Abb. 3.5). Die Ergebnisse können entweder als Klartext oder im VOTable-Format<sup>9</sup> heruntergeladen werden, wobei die Metadaten im Kopfteil der einzelnen Dateien zu finden sind. Die jeweiligen SEDs werden in Tübingen gespeichert, während die zugehörigen Metadaten-Dateien in Heidelberg gesichert sind. Für eine erste Übersicht kann der Nutzer die Spektren auch mit Specview<sup>10</sup> darstellen.

*TheoSSA* nutzt den VO-Standard Simple Spectral Access Protocol (SSAP, SSA<sup>11</sup>) für eindimensionale Spektren und SEDs. Das bedeutet, dass die Informationen über die SEDs in dem Format des Standards SSA gespeichert werden, was es z.B. der Suchfunktion ermöglicht die gewünschten Daten zu finden und sie als VOTable darzustellen. Die Datensätze werden mit Hilfe der so genannten „SSA data models“ beschrieben. Diese haben unterschiedliche Komponenten (z.B. „DataID“), die jeweils wieder verschiedene Kategorien haben (z.B. „must“ oder „should“) und weiter in so genannte UTYPES unterteilt sind (z.B. „DataID.Title“ oder „DataID.Creator“). „Data model metadata“ beschreiben die Daten selbst. Von ihnen sind mehrere im Standard SSA definiert, eine davon ist in Abb. 3.6 zu sehen, die den Datensatz „identification metadata“ zeigt. In dieser Abbildung sind die verschiedenen UTYPES dieses data models aufgelistet und beschrieben und die jeweiligen Anforderungen sowie Standardeinstellungen werden angegeben. Alle in SSA existierenden data models sind in der zugehörigen VO-Dokumentation beschrieben<sup>8</sup>. Jeder, der seinen Datensatz veröffentlichen möchte, kann seine Informationen über die Daten anhand dieses Dokuments so umformen, dass sie dem SSA-Standard entsprechen und sie dann veröffentlichen.

<i>UTYPE</i>	<i>Description</i>	<i>Req</i>	<i>Default</i>
<b>DataID.Title</b>	Dataset title	MAN	
DataID.Creator	Creator name (string)	REC	
DataID.Collection	Collection shortname (string)	REC	
DataID.CreatorDID	Creator assigned dataset identifier	REC	
DataID.Date	Data processing/creation date	OPT	
DataID.Version	Version of creator-produced dataset	OPT	
DataID.Instrument	Instrument name	OPT	
DataID.Bandpass	Bandpass name, e.g., filter	OPT	
DataID.DataSource	Original source of data	REC	survey
DataID.CreationType	Dataset creation type	REC	archival

Abbildung 3.6.: Auszug aus der IVOA-Dokumentation für SSA, Version 0.97 (Quelle siehe Text). Diese Abbildung gibt eine Übersicht über die dataset identification-Metadaten. MAN steht für mandatory und bedeutet, dass dieser Eintrag auf jeden Fall ausgefüllt werden muss, REC für recommended, also empfohlen, und OPT für optional.

*TheoSSA* ist eine der ersten Datenbanken, die theoretische SEDs nach VO-Vorgaben zur Verfügung stellt. Die Spektren der *TheoSSA*-Datenbank bieten eine realistische Repräsentation des (Zentral)Sterns – im Gegensatz zu vorher verwendeten Schwarzkörperflüssen. Außerdem ist mit den enthaltenen Spektren eine erste, grobe Analyse heißer, kompakter Objekte möglich. Bei Analysen von Objekten, zu denen  $T_{\text{eff}}$  und  $\log g$  noch nicht bekannt sind, kann so eine erhebliche Zeitersparnis erzielt werden. Außerdem wird die Doppelberechnung vieler Spektren verhindert, da nicht mehr für jeden TMAP-Anwender nur seine eigenen Modelle, sondern auch die der anderen Benutzer über *TheoSSA* zugänglich sind. VO-Nutzer, die sich bis dahin nicht oder kaum mit der Analyse von beobachteten Spektren beschäftigt haben, haben mit diesem Dienst die Möglichkeit in die Spektralanalyse einzusteigen ohne großes Hintergrundwissen mit zu bringen.

<sup>9</sup><http://www.ivoa.net/Documents/VOTable/>

<sup>10</sup>[http://www.stsci.edu/institute/software\\_hardware/specview](http://www.stsci.edu/institute/software_hardware/specview)

<sup>11</sup><http://www.nrao.edu/~dtody/ssa/ssa-v097.pdf>

Nach der Einführung des *TheoSSA*-Dienstes wurde jedoch schnell deutlich, dass nicht alle Anfragen nach *TMAP*-Spektren durch *TheoSSA* abgedeckt werden. Die Nachfrage nach individuellen Spektren wurde größer, woraufhin zusätzlich ein automatischer Rechendienst (*TMAW*, Kapitel 3.4) eingerichtet wurde. Führt eine *TheoSSA*-Suche nicht zum gewünschten Ergebnis, so kann der Benutzer den *TMAW*-Dienst aufrufen. Mit diesem kann er SEDs mit individuell bestimmten Parametern berechnen lassen. Diese SEDs werden automatisch in die *TheoSSA*-Datenbank eingefügt, wodurch sich diese ständig erweitert. *TheoSSA* enthält deshalb sowohl Modelle, die von einzelnen *TMAP*-Anwendern berechnet wurden, als auch solche, die automatisch mit *TMAW* erzeugt wurden. Sie können über den Metadateneintrag „DataID.Creator“ unterschieden werden.

Mit den *TheoSSA*- und *TMAW*-Dienstes wird die NLTE-Spektralanalyse weltweit vereinfacht und auch Nicht-Spezialisten zugänglich gemacht. Allerdings werden auf diese Art die Erzeuger der Spektren nicht mehr als Koautoren von Veröffentlichungen genannt, sondern es wird allgemein im Acknowledgement zitiert: „The *TheoSSA* service (<http://dc.g-vo.org/theossa>) used to retrieve theoretical spectra for this paper was constructed as part of the activities of the German Astrophysical Virtual Observatory.“

### 3.4. *TMAW* – Berechnung individueller synthetischer Spektren

SEDs, die nicht in der *TheoSSA*-Datenbank zu finden sind, können mit dem web-basierten *TMAP*-Dienst (*TMAW*) berechnet werden. Wie *TheoSSA* wird auch *TMAW* über eine Internetseite bedient (Abb. 3.7). Dabei müssen die grundlegenden Parameter wie  $T_{\text{eff}}$ ,  $\log g$  und Elementhäufigkeiten sowie Kontaktdaten wie die Email-Adresse angegeben werden. Der Absendeprozess der Daten bereitet die automatische Berechnung einer oder mehrerer SEDs mit den gewünschten Parametern vor, indem diese in eine oder mehrere Dateien geschrieben werden. Diese Dateien werden automatisch (mit Hilfe eines Cron-Jobs) nacheinander abgearbeitet, d.h. die SEDs zu diesen Parametern berechnet. Wenn die Berechnung beendet ist, bekommt der Anwender per Email die URL mitgeteilt, von der er die Ergebnisse herunterladen kann. Die resultierenden SEDs werden, wie eben erwähnt, zusätzlich in die *TheoSSA*-Datenbank eingefügt, um unnötige Neuberechnungen zu verhindern.

In *TMAW*-Rechnungen konnten bisher die Elemente H, He, C, N und O berücksichtigt werden. Gegen Ende der Arbeit wurde *TMAW* in einem parallelen Projekt von N. Reindl um die Elemente Ne und Mg erweitert. Der Dienst kann auch weiterhin um zusätzliche Elemente ergänzt werden. Wie gut er bereits jetzt genutzt wird, zeigt Abb. 3.8. Es ist deutlich zu erkennen, dass die Nachfrage nach *TMAW*-Modellen zunimmt. Dies ist unter anderem dem steigenden Bekanntheitsgrad von *TMAW* zu verdanken (siehe z.B. Webcounter und Weltkarte (inklusive Link zu Detailangaben) mit Markierungen von wo aus *TMAW* aufgerufen wurde in Abb. 3.7, abgerufen: 17.01.2013).

#### 3.4.1. Hintergrundoperationen nach dem Absenden eines Requests

Der Nutzer des *TMAW*-Dienstes muss diesen gewissermaßen als Blackbox verwenden. Die Schritte, die nach dem Absenden des Requests durchgeführt werden, sind für ihn nicht direkt einsichtig und werden deshalb im Folgenden vorgestellt.

##### Erzeugen der Request Datei

Sobald ein Request über die *TMAW*-Webseite abgesendet wird, erzeugt die Prozedur `/www/uncgi-bin/TMAP_WWW_interface_script` eine Datei, die alle über die Webseite angegebenen Informationen enthält. Diese Request-Datei liegt in dem Verzeichnis `/home/TMAW/requests`, bis ihre Bearbeitung startet. Alle Prozeduren, die zur Berechnung der SEDs verwendet werden, sind Unix-Shell-Skripte und befinden sich im Verzeichnis `/home/TMAW/scripts`. Die Programme, die von den Prozeduren verwendet werden, sind in FORTRAN geschrieben und in `/home/TMAW/FORTRAN`

**GERMAN ASTROPHYSICAL VIRTUAL OBSERVATORY**  
**GAVO**  
 TMAP WWW Interface

**German Astrophysical Virtual Observatory**  
 TMAP WWW Interface

Home  
 About GAVO  
 Getting Started  
 GAVO Data Center  
 Documents  
 Internal

Search

Sponsored by  
 Federal Ministry of Education and Research

Member of the International Virtual Observatory Alliance

622 Visitors  
 24 Apr 2011 17:20:29

**TMAW Request**

Please specify effective temperature  $T_{\text{eff}}$ , surface gravity  $\log g$ , abundances for H, He, C, N, O, Ne, and Mg as well as your e-mail address.

A NLTE model atmosphere with your input parameters will be calculated by *TMAP* - the Tübingen NLTE Model-Atmosphere Package - and you will be informed about the progress by e-mail.

Personal Information

Last Name   
 First Name   
 Institute   
 E-mail

SED Parameters

Wavelength range for standard SED:  
 5 - 2000 Å  2000 - 3000 Å  3000 - 55000 Å

Wavelength range for an individual SED and a quicklook plot:  
 3500  -  7000  Å,  $\Delta\lambda \approx$   0.1  Å  
 Note: the maximum number of data points is about 50 000.

Before submitting, please verify the data you have filled in this form, especially your e-mail address.

Model-Grid Parameters

$T_{\text{eff}}$  [K]:  $20\,000\text{ K} \leq T_{\text{eff}} \leq 300\,000\text{ K}$

Minimum	Maximum	Grid spacing
<input type="text" value="100000"/>	<input type="text" value="100000"/>	<input type="text" value="20000"/>

$\log g$  [ $\text{cm/s}^2$ ]:  $4.0 \leq \log g \leq 9.9$

Minimum	Maximum	Grid spacing
<input type="text" value="7.0"/>	<input type="text" value="7.0"/>	<input type="text" value="0.5"/>

Abundances [mass fractions, preset with solar values (Asplund et al. 2009, ARAA 47, 481)]:

H:  He:   
 C:  N:  O:   
 Ne:  Mg:

This WWW interface is fully functional. A H+He+C+N+O+Ne+Mg SED calculation takes about two days. Numerical instabilities may occur due to the requested parameters. We will then check for these asap. Please do not hesitate to start any calculation - this helps us to further improve the TMAW procedure.

Please do not hesitate to contact us in case of any question.

© 2003 - 2013 German Astrophysical Virtual Observatory and the contributing authors

Impressum / Contacts

© 2003 - 2013 German Astrophysical Virtual Observatory and the contributing authors

WEB  
 This page has been visited 1671 times since June 2, 2008

Abbildung 3.7.: TMAW-Request-Webseite.

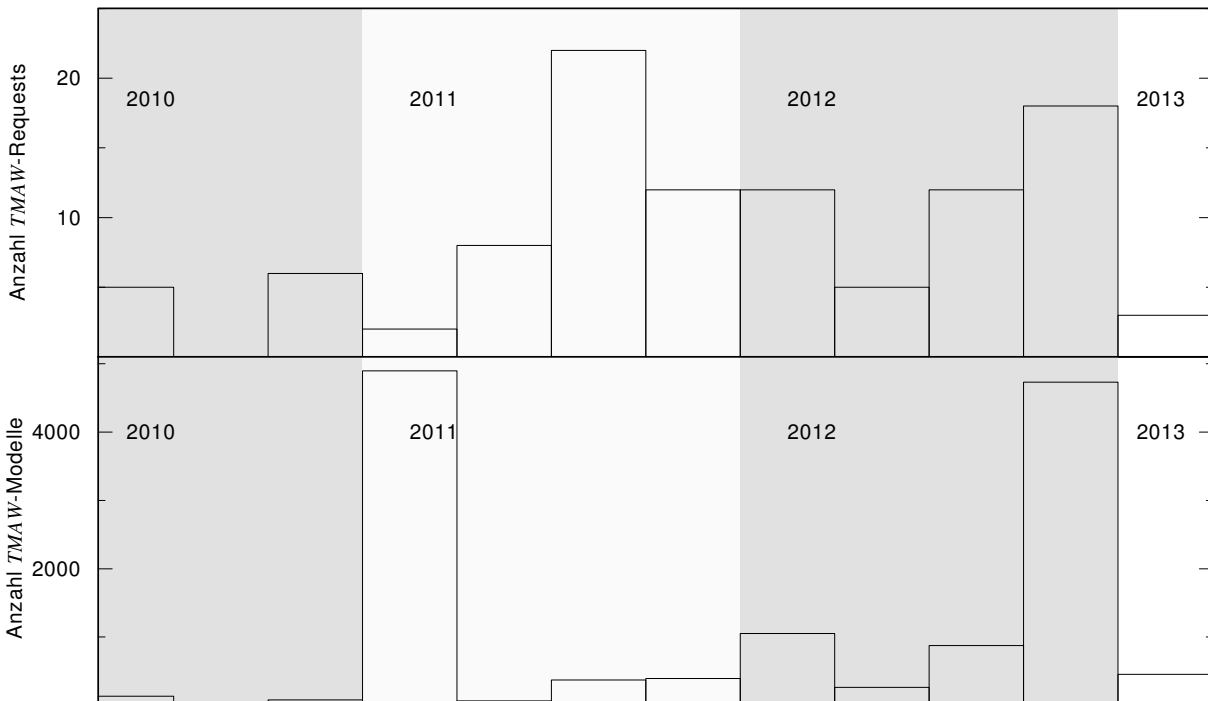


Abbildung 3.8.: Zahl der extern eingegangenen TMAW-Requests (oben) und der angeforderten Modelle (unten) pro Quartal seit Beginn dieser Arbeit. Stand: 23.01.2013



zu finden. Ihre ausführbaren Dateien befinden sich im Verzeichnis `/home/TMAW/bin` und stehen sowohl als 32-bit- als auch als 64-bit-Versionen (`.Linux_X86` bzw. `.Linux_x64` als Erweiterung) zur Verfügung. Alle zur Berechnung der SEDs verwendeten Prozeduren sind im Anhang dieser Arbeit zu finden.

**Eine Übersicht** über die einzelnen Schritte, die bei einer *TMAW*-Berechnung durchlaufen werden, ist in Abb. 3.9 und 3.10 dargestellt. Wie bei der Berechnung mit *TMAP* wird – nach einer Prüfung, ob genug Rechenkapazität verfügbar ist – zuerst ein Modellatom erstellt. Hierbei werden Modellatome aus der *TMAD*-Datenbank für den jeweiligen Temperaturbereich verwendet. Anhand dieser Modellatome werden Frequenzgitter erzeugt und die Parameter für die Programmdateien überprüft. Die eigentliche Berechnung des Sternatmosphärenmodells beginnt mit einem Startmodell (ein Kontinuumsmodell in grauer Näherung). Anschließend folgen verschiedene Schritte für die Erzeugung eines Kontinuumsmodells, dem später, in weiteren Schritten, Linien hinzugefügt werden. Zum Abschluss der Berechnung wird das synthetische Spektrum erzeugt und in ein komprimiertes tar-Archiv gepackt, das mit Hilfe eines `wget`-Kommandos heruntergeladen werden kann. All diese Schritte werden von verschiedenen Prozeduren ausgeführt, die im Folgenden genauer beschrieben werden.

### Die einzelnen *TMAW*-Schritte zur Modellatmosphärenberechnung

Das IAAT besitzt zur Zeit 16 Rechner mit Vierkernprozessoren, auf denen unter der Woche maximal jeweils zwei, am Wochenende maximal drei Requests gleichzeitig abgearbeitet werden. Zusätzlich steht ein Zwölfkernrechner mit Hyper-Threading-Technologie (*GAVO @IAAT-Server*) zur Verfügung, auf dem bis zu 24 Requests gleichzeitig berechnet werden können.

- `checkreq.sh`: Mit Hilfe eines Cron-Jobs wird auf dem *GAVO @IAAT-Server* jede Minute, auf den anderen jede Viertelstunde, die Prozedur `checkreq.sh` ausgeführt. Diese Prozedur prüft zuerst, ob zur Zeit eine Maintenance-Phase ist, in der keine Requests bearbeitet werden, ob ein anderer Request gerade kompiliert und (mit der Programmdatei `/home/rauch/tools/readload.Linux_X86`) ob der jeweilige Rechner genug freie Kerne hat, um den Request zu bearbeiten. Anschließend wird die Zahl der Requests, die bereits auf diesem Rechner abgearbeitet werden, ermittelt. Wenn diese klein genug ist, der Rechner noch Kapazität frei hat, kein anderer Request kompiliert und keine Maintenance-Phase ist, wird die Prozedur `TMAW_iaat_run.sh` gestartet.
- `TMAW_iaat_run.sh` ist die eigentliche Hauptprozedur. Nach einer Änderung des Formatstrings für  $T_{\text{eff}}$  und  $\log g$  (mit Hilfe der unter `/home/rauch/bimod` abgelegten Programmen `TEFF_I7.Linux_X86` und `LOGG.Linux_X86`) und einer Normierung der Elementhäufigkeiten (`TMAW_norm.Linux_X86`) wird zuerst geprüft, ob ein früheres Modell mit den selben Sternparametern bereits eine Ergebnisdatei erzeugt hat. Ist dies der Fall, wird eine Email mit einem Übersichtsplot über vorhandene SEDs mit dieser Häufigkeitskombination im  $\log g$ - $\log T_{\text{eff}}$ -Diagramm an den Nutzer gesendet. Ansonsten startet `TMAW_iaat_run.sh` zuerst die Prozedur `TMAW_iaat_prepare.sh` für die Erzeugung der für die spätere Berechnung der Atmosphäre benötigten Input-Daten. Anschließend wird die Prozedur `TMAW_iaat_jobfile.sh` zur eigentlichen Berechnung der Modellatmosphäre und synthetischen Spektren ausgeführt.

Sind diese beiden Prozeduren beendet und die SEDs erzeugt, werden von `TMAW_iaat_run.sh` grafische Darstellungen der Temperaturstruktur und des Flussverlaufs sowie Metadaten (mit Hilfe von `split.Linux_X86`, `fcloudy.Linux_X86` und `devide.Linux_X86`) erstellt. Die Ergebnisse der Atmosphärenberechnung (SEDs, Darstellungen der Temperaturstruktur und des Flusses) werden in ein komprimiertes tar-Archiv zusammengefügt und in ein von außerhalb des Instituts frei zugängliches Verzeichnis gelegt, sodass der Anwender die Daten mit Hilfe eines `wget`-Kommandos herunterladen kann.

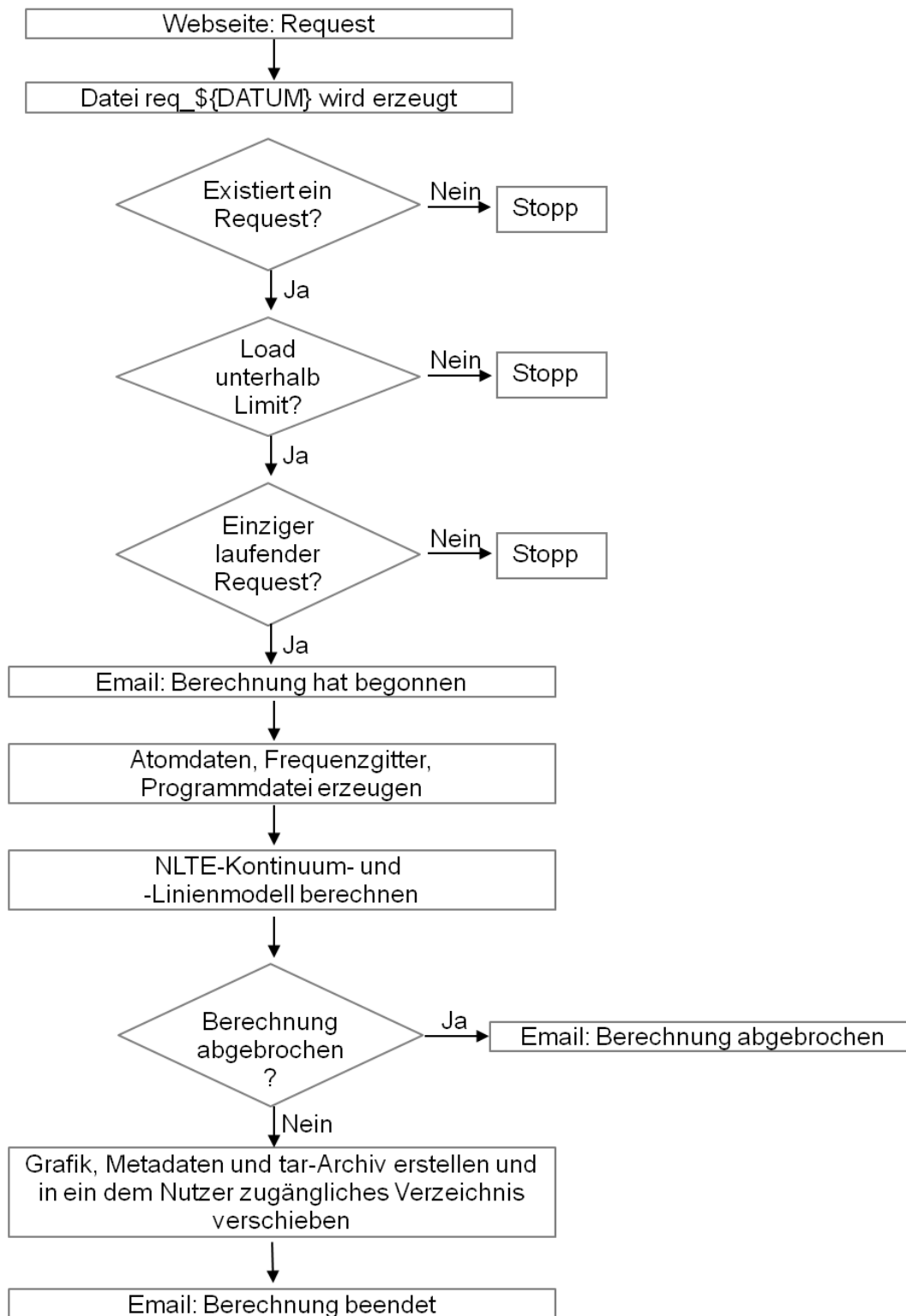


Abbildung 3.9.: Schematische Übersicht über die grundlegenden *TMAW*-Schritte.

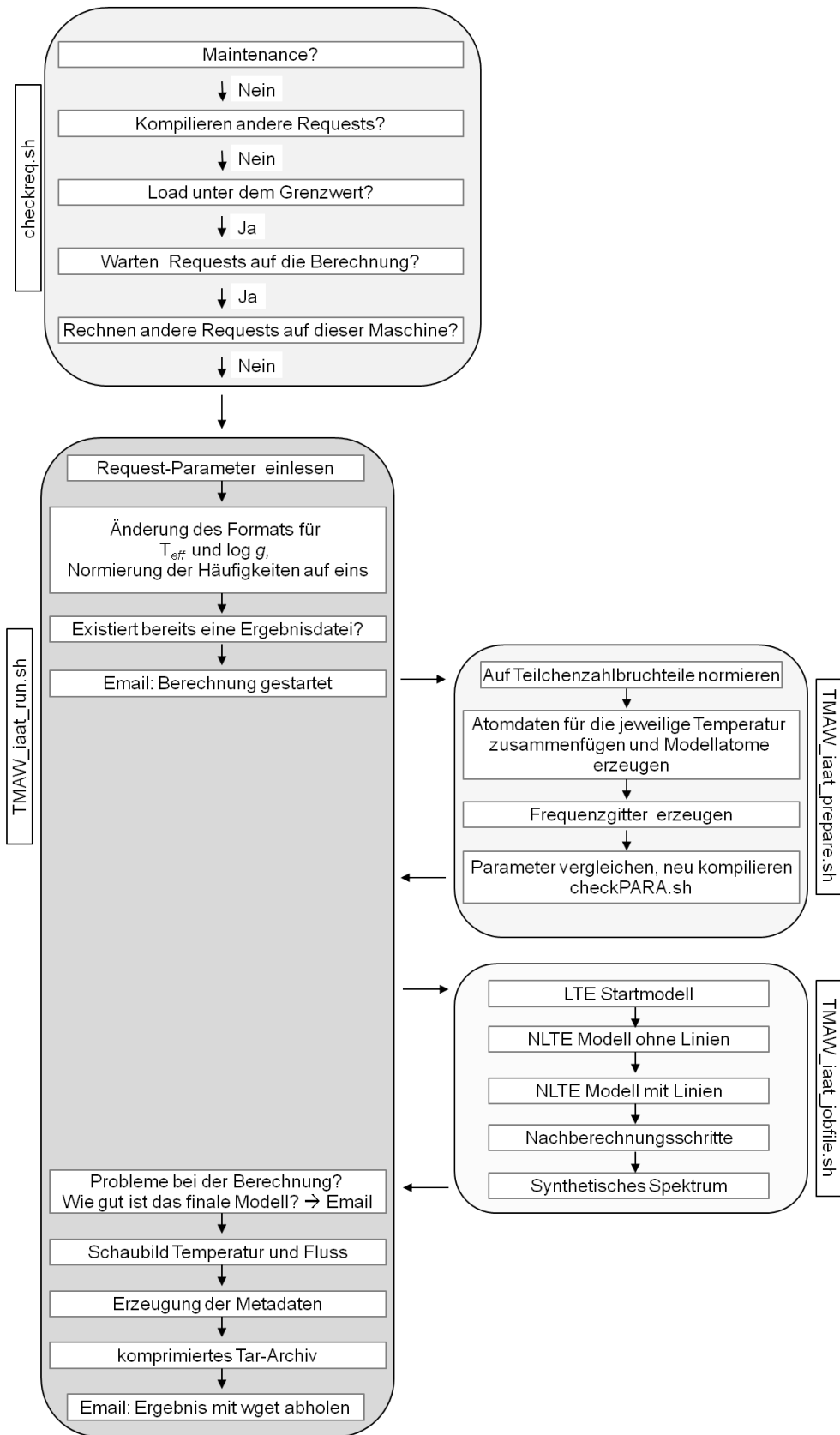


Abbildung 3.10.: Schematische Übersicht über die Aufgaben der TMAW-Hauptprozeduren.

Tabelle 3.1.: Statistik ausgewählter *TMAW*-Modellatome für unterschiedliche Temperaturbereiche. Für jedes Ion ist die Zahl der NLTE-Levels und die Zahl der Linien (RBB) angegeben.

$T_{\text{eff}}/ \text{kK}$	20 – 30		70 – 100		150 – 180	
Ion	NLTE	RBB	NLTE	RBB	NLTE	RBB
H I	10	45	10	45	10	45
H II	1	–	1	–	1	–
He I	19	35	19	35	5	3
He II	14	91	14	91	14	91
He III	1	–	1	–	1	–
C I	15	19				
C II	16	37	16	37		
C III	13	32	13	32		
C IV	14	35	14	35	14	35
C V	1	0	1	0	19	33
C VI					1	0
N I	1	0				
N II	14	14	1	0		
N III	13	24	13	24		
N IV	16	30	16	30	1	0
N V	1	0	14	35	14	35
N VI			1	0	11	14
N VII					1	0
O I	14	18				
O II	16	26	16	26		
O III	21	38	21	38		
O IV	18	39	18	39	1	0
O V	1	0	17	35	17	35
O VI			1	0	14	33
O VIII					1	0

Die Prozedur benachrichtigt sowohl den Anwender als auch den *TMAW*-Administrator immer wieder über den Status des Requests per Email. Der Anwender erhält Emails, wenn die Berechnung seines (oder des ersten Requests seines Gitters) beginnt und wenn die Berechnung abgeschlossen ist. Sind alle Requests eines Gitters abgearbeitet, so ist in einer finalen Email zusätzlich eine Übersicht ( $\log g$ - $\log T_{\text{eff}}$ -Diagramm) über alle Requests, die die Berechnung abgeschlossen haben, enthalten.

Der *TMAW*-Administrator erhält zusätzlich Emails zur Qualitätskontrolle des *TMAW*-Dienstes bzw. wird benachrichtigt wenn die Berechnung erfolglos war.

- `TMAW_iaat_prepare.sh` wird von der Hauptprozedur gestartet und ist für die Vorbereitung der Modellatmosphärenberechnung zuständig. Bevor diese Berechnung beginnen kann, müssen (nach einer Normierung der Elementhäufigkeiten auf Teilchenzahlhäufigkeiten mit Hilfe von `/home/rauch/bimod/TMAW_mass2num.Linux_X86` und `/home/rauch/bimod/TMAW_gt.Linux_X86`) zuerst Atomdaten mit den für die jeweilige Temperatur relevanten Ionisationsstufen erstellt werden. Dabei sind für die Berechnung des Start-, des NLTE-, des Lineformationsmodells und des synthetischen Spektrums unterschiedliche Atomdaten nötig. Die Atomdaten, die in *TMAW*-Rechnungen verwendet werden, sind in der *TMAD*-

Tabelle 3.2.: Wie Tabelle 3.1, für die Lineformation-Rechnung.

$T_{\text{eff}}/ \text{kK}$	20 – 30		70 – 100		150 – 180	
	NLTE	RBB	NLTE	RBB	NLTE	RBB
Ion						
H I	15	105	15	105	15	105
H II	1	–	1	–	1	–
He I	103	504	103	504	29	69
He II	20	190	20	190	20	190
He III	1	–	1	–	1	–
C I	15	19				
C II	31	116	16	37		
C III	44	190	30	101		
C IV	14	35	35	148	35	148
C V	1	0	1	0	30	64
C VI					1	0
N I	1	0				
N II	23	39	1	0		
N III	31	109	31	109		
N IV	30	88	30	88	1	0
N V	1	0	35	149	35	149
N VI			1	0	26	80
N VII					1	0
O I	14	18				
O II	26	64	20	35		
O III	30	79	25	56		
O IV	18	39	20	47	20	47
O V	1	0	20	43	30	95
O VI			1	0	35	145
O VIII					1	0

Datenbank zu finden. Zusammen mit *TMAD* werden auch die *TMAW*-Modellatome regelmäßig auf den neuesten Stand gebracht. Um eine angemessene Rechenzeit zu garantieren, sind die *TMAW*-Modellatome allerdings klein im Vergleich zu denen, die für *TMAP*-Rechnungen verwendet werden. Ein Überblick über die Anzahl der berücksichtigten Levels und Linien ist in Tabelle 3.1 und 3.2 zu finden.

Sind die Atomdaten erstellt, werden die zugehörigen Frequenzgitter erzeugt. Für jedes Atomdaten-Frequenzgitter-Paar müssen die Parameter statements im FORTRAN-Code den Arraygrößen angepasst werden. Die jeweiligen Parameter werden mit denen der letzten Berechnung verglichen und eine neue Programmdatei wird, falls nötig, erstellt. Dies geschieht mit der Prozedur `checkPara.sh`, die die Programme `/home/TMAW/bin/read_PARA.Linux_x64` und `/home/TMAW/bin/create_PARA.Linux_x64` verwendet. Tritt beim nachfolgenden Kompilieren ein Fehler auf, so wird der *TMAW*-Administrator per Email benachrichtigt.

- `TMAW_iaat_jobfile.sh`: Für die eigentliche Berechnung der Modellatmosphäre wird die Prozedur `TMAW_iaat_jobfile.sh` verwendet. Sie erzeugt als Startmodell für die NLTE-Berechnung ein Kontinuumsmodell in grauer Näherung. Ausgehend von diesem werden in bis zu neun Schritten Kontinuumsmodelle berechnet (Tabelle 3.3). Dabei werden zuerst nur die statistischen Gleichungen gelöst, später werden auch die anderen Nebenbedingungen – bis auf die Glei-

Tabelle 3.3.: Die einzelnen Schritte der *TMAW*-Berechnung. Die ersten neun Schritte werden für das Kontinuumsmodell durchgeführt, die anderen für das Modell, das Linien berücksichtigt.

Schritt	LOG CVEC	Korrektur	Iterationen
1	-2.0	STAT EQU ONLY	30
2	-1.0	STAT EQU ONLY	20
3	0.0	STAT EQU ONLY	10
4	-2.0	NO T-CORR	10
5	-1.0	NO T-CORR	10
6	0.0	NO T-CORR	10
7	-2.0	UL T-CORR	10
8	-1.0	UL T-CORR	15
9	0.0	UL T-CORR	30
10	-2.0	NO T-CORR, STEP-UP	15
11	-2.0	NO T-CORR	10
12	-1.0	NO T-CORR	10
13	0.0	NO T-CORR	10
14	-2.0	UL T-CORR	10
15	-1.0	UL T-CORR	15
16	-0.5	UL T-CORR	15
17	0.0	UL T-CORR	15
18	-2.0	T-CORR ONLY -5.0 +4.0	15
19	-1.0	T-CORR ONLY -5.0 +4.0	15
20	0.0	STAT EQU ONLY	15

UL steht für Unsöld-Lucy, LOG CVEC für REDUCE LOG CVEC, NO T-CORR für NO TEMPERATURE CORRECTION, T-CORR ONLY -5.0 +4.0 für T-CORRECTION ONLY IN LOG M INTERVAL: -5.0 +4.0, STEP-UP für STEP-UP F-VALUES und STAT EQU ONLY für SOLVE STATISTICAL EQUATIONS ONLY.

chung für das radiativen Gleichgewicht zur Bestimmung der Temperaturschichtung – mit einbezogen. Für die Berechnung der Temperaturschichtung wird die Unsöld-Lucy-Temperaturkorrektur (Lucy, 1964) verwendet. REDUCE LOG CVEC trägt zur numerischen Stabilität bei, indem der Inhomogenitäts-Vektor skaliert und die Korrekturen verkleinert werden. Schritt drei und vier werden zur Zeit übersprungen, da ihre Durchführung nicht wesentlich zur Stabilisierung der Rechnung beiträgt.

Nach der Berechnung der Kontinuumsmodelle werden Linien hinzugefügt. Dafür werden zuerst deren Oszillatorenstärken in 15 Iterationen langsam hochgefahren. In sieben weiteren Schritten wird die Atmosphärenstruktur zunächst ohne Temperaturkorrektur berechnet, dann wird die Unsöld-Lucy-Temperaturkorrektur verwendet. Auch hier wird Schritt 16 momentan aus zeitlichen Gründen übersprungen. Schritt 17 wird bis zu vier Mal mit größerer Iterationszahl (50) wiederholt, um eine höhere Konvergenzstufe zu erreichen.

Nachdem Schritt 17 beendet ist, werden drei weitere Schritte durchgeführt, in denen zuerst die *TMAP*-Temperaturkorrektur (aus der radiativen Gleichgewichtsbedingung) verwendet und dann eine Lineformationrechnung durchgeführt wird. Das resultierende Modell wird als Basismodell für die nun folgende SED-Berechnung herangezogen.

Wenn die Zahl von Requests eines Gitters 150 übersteigt, dauert die Berechnung auf IAAT-Maschinen zu lange und die eigentlichen Nutzer des Clusters werden stark eingeschränkt. In diesem Fall kann

auf Rechner, die über das *AstroGrid-D*<sup>12</sup> zugänglich sind, zugegriffen werden (Kapitel 3.6). Dafür ruft die Prozedur `checkreq.sh` nicht `TMAW_iaat_run.sh` sondern `TMAW_grid_run.sh` auf, die mit `TMAW_grid_prepare.sh` die nötigen Atomdaten und Frequenzgitter erzeugt und in ein komprimiertes tar-Archiv packt. Die Prozedur `TMAW_grid_jobfile.sh` wird erst auf *AstroGrid-D*-zugänglichen Rechnern ausgeführt.

### 3.5. Qualitätskontrolle

Über die *VO*-Registries kann jeder seine Daten oder Dienste öffentlich zugänglich machen. Diese Registries sind wichtige Einrichtungen, um jeden am *VO* zu beteiligen und um möglichst viele Daten und Werkzeuge verfügbar zu bekommen. Um diese Dienste als *VO*-Nutzer verwenden zu können, ist es allerdings wichtig zu wissen, wofür sie erstellt worden sind, welche Qualität sie haben. Je nachdem welche Ziele der Nutzer oder auch der Erzeuger der Dienste verfolgt, kann die Qualität stark abweichen. Soll z.B. von einem beobachteten Spektrum nur der Flussverlauf betrachtet werden, ist eine genaue Wellenlängenkali- bration der Daten nicht ganz so wichtig als wenn eine detaillierte Spektralanalyse durchgeführt werden soll. Dies gilt nicht nur für Beobachtungsdaten, sondern betrifft alle veröffentlichten Daten, Dienste und Analysewerkzeuge. Abhängig vom Ziel der Arbeit sollte das verwendete *VO*-Produkt unterschiedlichen Ansprüchen genügen. Die Genauigkeit der Produkte sollte für den *VO*-Nutzer möglichst einfach einsehbar bzw. nachvollziehbar sein. Um zuverlässige wissenschaftliche Arbeit mit Hilfe von *VO*-Diensten und -Werkzeugen zu gewährleisten, muss deshalb dringend eine Qualitätskontrollfunktion eingerichtet werden.

Wie genau diese Qualitätskontrollfunktion aussehen soll, wird von den *VO*-Nutzern – bisher ohne Ergebnis – diskutiert. Momentan ist für dieses Thema, trotz der dringenden Notwendigkeit, noch keine einheitliche Regelung gefunden worden. Eine zentrale Einrichtung, die alle veröffentlichten Dienste und Daten kontrolliert und anschließend dokumentiert, übersteigt die Möglichkeiten der *VO*-Einrichtungen. Damit jeder Veröffentlichender seine Produkte selbst dokumentieren kann, wären Richtlinien nötig, an die sich die Autoren halten können.

Für viele beobachtete Daten wurden bereits einige Kontrollinstanzen eingerichtet, die die Qualität der Daten überprüfen und diese eventuell neu reduzieren bzw. nochmals überarbeiten. Zu diesen Kontrollinstanzen gehören z.B. die ESO Quality Control and Data Processing Group<sup>13</sup> oder das NASA Quality Program<sup>14</sup>. Für die beobachteten Daten wurden außerdem bereits Metadaten eingeführt, die oft im Kopfteil der FITS-Dateien stehen. So werden z.B. einzelne Reduktions- oder Kalibrationsschritte dokumentiert oder Informationen über das Instrument oder die Beobachtungsnacht angegeben. Auch für theoretische Daten existieren Metadaten, die zur Qualitätskontrolle verwendet werden könnten. Zum Beispiel wird der Nutzer der *TheoSSA*-Datenbank über die Metadaten informiert, ob es sich um ein mit *TMAP* oder mit *TMAW* berechnetes Spektrum handelt.

Ein weiteres Beispiel für bereits eingeführte Qualitätskontrollen ist bei den Atomdaten zu finden. Einige Datenbanken vergleichen die Wellenlängen und Oszillatorenstärken mit im Labor bestimmten Daten – falls vorhanden – und erhalten so Atomdaten mit höherer Genauigkeit und können Rückschlüsse auf ihr Programm zur Berechnung der Daten ziehen.

Die ersten Schritte in Richtung Qualitätskontrolle im *VO* sind also bereits von einzelnen Gruppen unternommen worden. Sie können als Hinweise oder Vorschläge, wie eine globale Qualitätskontrolle im *VO* aussehen bzw. nach welchen Richtlinien eine globale Qualitätskontrolle umgesetzt werden könnte, verstanden werden. Eine einheitliche Regelung sollte dennoch bald gefunden werden.

<sup>12</sup><http://www.gac-grid.net/>. *AstroGrid-D* ist ein Gemeinschaftsprojekt mehrerer Einrichtungen der Astronomie und Informatik, das in Deutschland arbeitenden Astronomen den Zugang zu mehreren Clustern nach Vorgaben der IVOA ermöglicht und so eine Vernetzung der Arbeit fördert.

<sup>13</sup><http://www.eso.org/observing/dfo/quality/>

<sup>14</sup><http://www.hq.nasa.gov/office/codeq/quality/>

### 3.5.1. Benchmarktests

Obwohl für beobachtete Daten mehrere Kontrollinstanzen existieren, gibt es auf der theoretischen Seite kaum eine Einrichtung, die über das *VO* zugängliche Daten prüft. Eigentlich ist jeder *VO*-Nutzer selbst dafür verantwortlich, in der vorhandenen Literatur (z.B. in Handbüchern oder Dokumentationen) zu überprüfen, wofür die Dienste und Daten verwendet werden können. Trotzdem werden viele Datensätze ohne eine solche Hintergrundprüfung verwendet.

Beim *TMAW*-Dienst ist die Situation ähnlich. Der *VO*-Nutzer kann den Dienst quasi als Blackbox ohne Hintergrundwissen verwenden und es bleibt fraglich, ob er sich informiert wozu *TMAW* bzw. der zugrunde liegende Modellatmosphärencode (*TMAP*) geeignet ist und inwiefern *TMAP* und *TMAW* übereinstimmende Ergebnisse liefern. Um letzteres zu prüfen, wurde damit begonnen, die Genauigkeit von *TMAW*-SEDs zu ermitteln und somit die ersten Benchmarktest für theoretische Daten im *VO* einzuführen. Für diese Tests werden Spektren, erzeugt mit dem *TMAW*-Dienst, mit detaillierteren und präziseren *TMAP*-Spektren verglichen (Kapitel 3.5.1 und 4). Nutzt der Anwender diese Tests, so kann er leicht abschätzen, mit welcher Genauigkeit *TMAW* beispielsweise für Spektralanalysen verwendet werden kann.

#### Unterschiede zwischen *TMAW* und *TMAP*

Die ursprüngliche Motivation für *TMAW* war, synthetische, ionisierende Spektren für Photoionisationscodes für Planetarische Nebel zur Verfügung zu stellen, um eine – im Gegensatz zu Schwarzkörperflüssen – realistische Darstellung eines Zentralsterns zu liefern (Abb. 3.11). Für exakte *TMAP*-Rechnungen werden zum Teil Monate an Rechenzeit benötigt. Dies ist eine unannehmbare Wartezeit für den *VO*-Nutzer. Um eine angemessene Rechenzeit, und damit Wartezeit auf das Produkt, zu gewährleisten, kann *TMAW* nicht dieselbe Genauigkeit wie *TMAP*-Rechnungen haben. Es werden von *TMAW* deshalb nur die Elemente H, He, C, N und O berücksichtigt, die Modellatome sind kleiner gewählt, beinhalten also weniger NLTE-Levels, es wird eine stabilere Temperaturkorrektur verwendet und weniger Iterationen werden durchgeführt, wodurch das formale Konvergenzkriterium (absolute Korrekturen kleiner  $10^{-4}$ ) eventuell nicht erreicht ist. Mit diesen Vereinfachungen ist es möglich, einen Request innerhalb von ein bis zwei Tagen zu berechnen.

*TMAP* wird erfolgreich für präzise Spektralanalysen verwendet, weshalb grundsätzlich auch die *TMAW*-SEDs für Spektralanalysen herangezogen werden können. Da *TMAW* dafür aber ursprünglich nicht ausgelegt war, werden im Rahmen dieser Arbeit Tests durchgeführt, um die Genauigkeit der mit *TMAW*-Modellen erreichten Ergebnisse zu überprüfen. Sind die erzeugten Spektren trotz der eben beschriebenen Vereinfachungen noch präzise genug, um zuverlässige Spektralanalysen anhand von hochaufgelösten Beobachtungen in allen Wellenlängenbereichen zu gewährleisten? Muss die Zahl der Iterationen oder die Modellatome angepasst werden? Wird das Ziel einer Genauigkeit von 10% oder weniger im Vergleich zu *TMAP*-Flüssen und besser als 20% im Vergleich zu *TMAP*-Analysen für  $T_{\text{eff}}$ ,  $\log g$  und die Häufigkeiten erreicht?

#### Der Fall von AA Doradus – ein erster Benchmarktest

Für einen ersten Test bot sich ein Vergleich von *TMAW*-Modellen mit einem feinen Gitter von *TMAP*-Modellen an, das kürzlich für die Spektralanalyse des sdOB-Primärsterns von AA Doradus erzeugt wurde (Klepp & Rauch, 2011). Bei der Berechnung der Modelle dieses Gitters wurden im Rahmen der Analyse von Klepp & Rauch (2011) die Elemente H, He, C, N, O, Mg, Si, P, S, Ca, Sc, Ti, V, Cr, Mn, Fe, Co und Ni mit 530 NLTE-Levels und 771 Linien für die klassischen Modellatome und 636 Superlinien (Kapitel 2) für die Eisengruppenelemente berücksichtigt. Um das am besten zu den optischen UVES<sup>15</sup>-Beobachtungen (ProgID 66.D–180) passende Modell zu finden, wurde von Klepp & Rauch (2011) ein  $\chi^2$ -Fit durchgeführt. Die daraus resultierenden  $T_{\text{eff}}$ - und  $\log g$ -Werte stimmen nun mit Werten aus früheren Analysen der Lichtkurven überein, was das so genannte  $\log g$ -Problem löste.

<sup>15</sup><http://www.eso.org/sci/facilities/paranal/instruments/uves/>



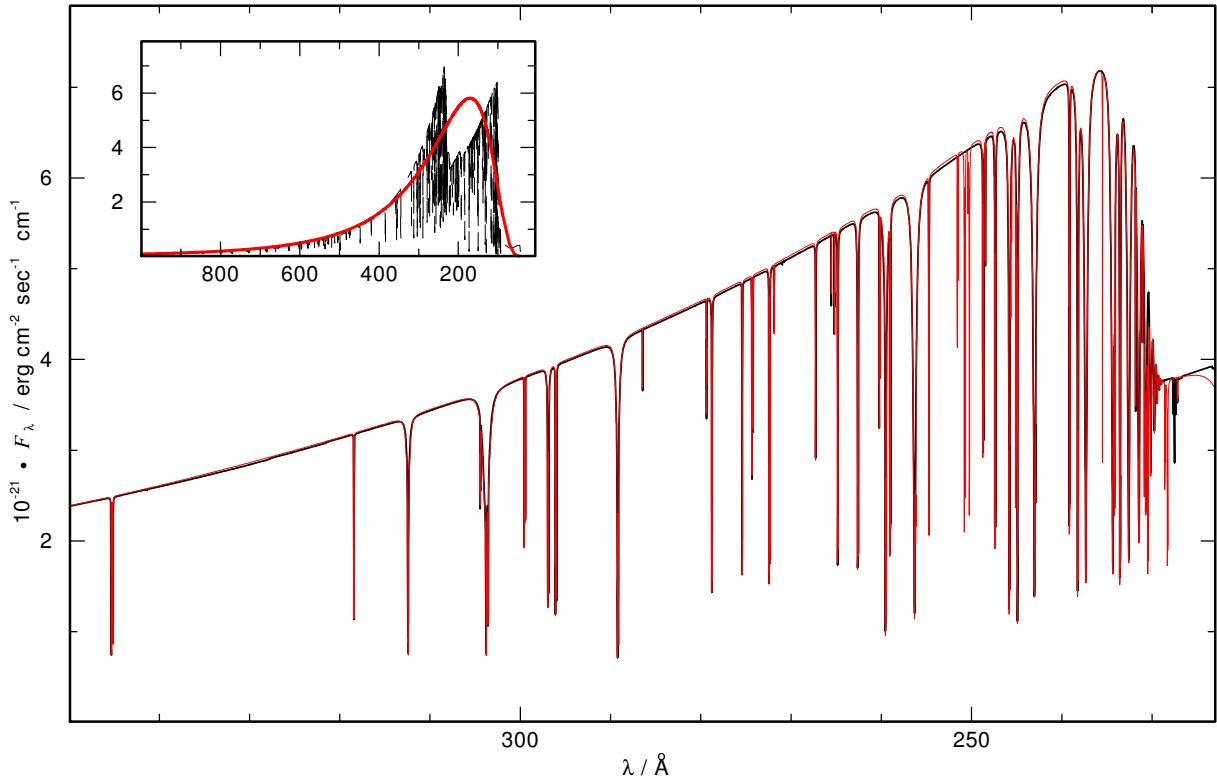


Abbildung 3.11.: Fluss eines *TMAP*-Modells (rot, dünn) verglichen mit einem *TMAW*-Fluss ( $T_{\text{eff}} = 170\,000\text{ K}$ ,  $\log g = 6.5$ ) im Bereich der stärksten Abweichungen. Das kleine Plot zeigt den Vergleich mit einem Schwarzkörperfluss (dicke, rote Linie) mit der selben Temperatur. Die Modelle berücksichtigen die Elemente H, He, C, N und O.

Das für die Analyse verwendete Gitter wurde im Rahmen dieser Arbeit mit *TMAW* nachgerechnet. Dabei wurden die Elemente H, He, C, N und O mit den von Klepp & Rauch (2011) bestimmten Häufigkeiten verwendet. Das Gitter besteht aus 527 Modellen ( $39\,500 \leq T_{\text{eff}} \leq 43\,500\text{ K}$ ,  $\Delta T_{\text{eff}} = 250\text{ K}$ ,  $5.30 \leq \log g \leq 5.60$ ,  $\Delta \log g = 0.01$ ), die auf über das *AstroGrid-D* zugänglichen Rechnern berechnet wurden. Die resultierenden synthetischen Spektren wurden an die bestehende  $\chi^2$ -Fitprozedur von Klepp & Rauch (2011) übergeben. Der von Rauch durchgeführte  $\chi^2$ -Fit hatte zuerst kein eindeutiges Ergebnis. Obwohl die Modelle denen von *TMAP* sehr ähnlich sind (Abb. 3.12), waren sie nicht exakt genug für einen verlässlichen  $\chi^2$ -Fit.

Die Abweichungen zwischen den Modellen kommen vor allem durch die unterschiedliche Anzahl berücksichtigter Elemente (in Abb. 3.12 deutlich an den zusätzlichen Linien im *TMAP*-Modell zu erkennen), Iterationen (ca. 300 für die *TMAW*-Modelle, mehr als 8000 für die Klepp & Rauch (2011)-Modelle), Größe der Modellatome (für die *TMAW*-Modelle wurden Modellatome mit 207 NLTE-Levels und 498 Linienübergängen verwendet) und einer unterschiedlichen Temperaturkorrektur zu Stande.

Die Verwendung kleiner Modellatome hat unter anderem zur Folge, dass Linien im optischen Wellenlängenbereich, die (für die Elemente C, N und O) vor allem von Übergängen zwischen Levels mit Hauptquantenzahl  $n = 6$  und  $n = 8$  entstehen, nicht im Modell enthalten sind. Außerdem ist der Unterschied in der Temperaturschichtung, hauptsächlich verursacht durch die Verwendung der Unsöld-Lucy-Temperaturkorrektur, für die *TMAW*-Modelle deutlich (Abb. 3.13).

### Erweiterungen des *TMAW*-Dienstes

Um trotz der kleinen Abweichungen der Modelle ein Ergebnis mit einem  $\chi^2$ -Fit zu erzielen, wurden weitere Schritte zur Berechnung der *TMAW*-Spektren eingeführt. Diese erfolgen zeitlich nach den ursprüng-

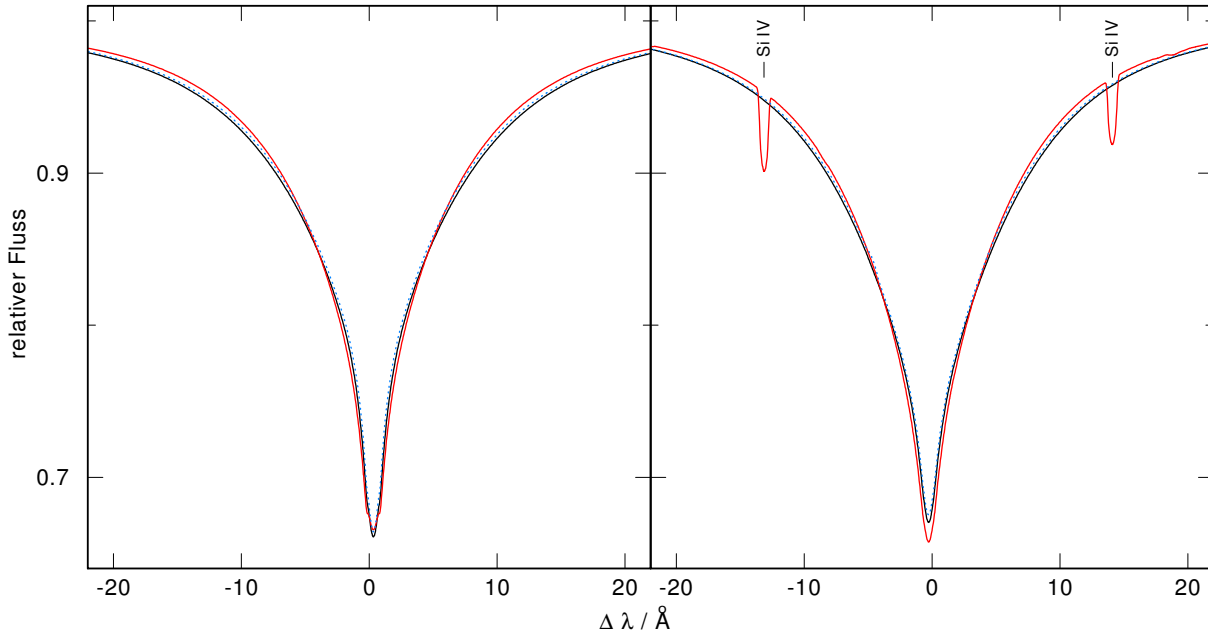


Abbildung 3.12.: Vergleich der H I-Linien bei 4861 Å ( $H\beta$ , links,  $v_{\text{rad}} = 30$  km/sec) und 4102 Å ( $H\delta$ , rechts,  $v_{\text{rad}} = 30$  km/sec) des *TMAP*-Modells von Klepp & Rauch (2011, rot,  $T_{\text{eff}} = 42\,000$  K,  $\log g = 5.46$ ) und eines *TMAW*-Modells vor (schwarz, nur Unsöld-Lucy-Temperaturkorrektur) und nach der Einführung der Nachberechnungsschritte (blau, gepunktet) mit den selben Sternparametern.

lichen *TMAW*-Rechenschritten und werden deshalb hier Nachberechnungsschritte genannt (Schritt 18, 19 und 20 in Tabelle 3.3). Die Idee für diese Nachberechnungsschritte war, die Unterschiede zwischen den *TMAW*- und *TMAP*-Berechnungen zu verkleinern, indem in der Nachberechnung zuerst die *TMAP*-Temperaturkorrektur in zwei zusätzlichen Schritten und dann größere Modellatome in einer Lineformationsrechnung im letzten Schritt verwendet werden. Um die Berechnung stabiler zu machen, wird in Schritt 18 und 19 die Temperaturkorrektur nicht in der gesamten Atmosphäre, sondern nur im Bereich von  $-5 \leq \log m \leq 4$  durchgeführt. Im Praxistest mit Requests von „außerhalb“, also von externen *TMAW*-Nutzern, hat sich später gezeigt, dass manche Modelle an diesen beiden Schritten scheitern. Deshalb wird nun zusätzlich eine Dämpfung für die Schritte 18 und 19 berücksichtigt. Für Schritt 20 wurden neue Modellatome angefertigt, die groß genug sind, um Linien im optischen Spektralbereich in die Modellierung mit einzubringen, aber gleichzeitig klein genug, um für jeden Temperaturbereich die maximal möglichen Parameterwerte (Arraygrößen) und eine kurze zusätzliche Rechenzeit einzuhalten.

Die hier beschriebenen Nachberechnungsschritte wurden zuerst für das Testgitter für AA Doradus durchgeführt. Die neuen Modelle wurden wieder an die bestehende  $\chi^2$ -Fitprozedur übergeben. Dies führte zu  $T_{\text{eff}} = 41\,150$  K und  $\log g = 5.42$ , was nur 1% in  $T_{\text{eff}}$  und 0.7% in  $\log g$  vom Klepp & Rauch (2011)-Ergebnis ( $T_{\text{eff}} = 40\,600$  K,  $\log g = 5.46$ ) abweicht. Diese Genauigkeit übertraf die Erwartungen bei weitem, und das obwohl *TMAW* deutlich weniger Elemente berücksichtigt, als zur Berechnung der *TMAP*-Modelle verwendet wurden.

### Bedeutung der erreichten Genauigkeit

Die Genauigkeit der *TMAW*-Modelle des ersten Benchmarktests für den Primärstern von AA Doradus liegt mit 1% in  $T_{\text{eff}}$  bzw. 0.7% in  $\log g$  deutlich unter dem gewünschten Wert von 20%. Wie ist diese Genauigkeit nun in den Kontext der Spektralanalyse von heißen, kompakten Objekten insgesamt einzuordnen?

Jeder Modellierung liegen mehrere Ungenauigkeiten zugrunde. Zum Beispiel werden immer noch

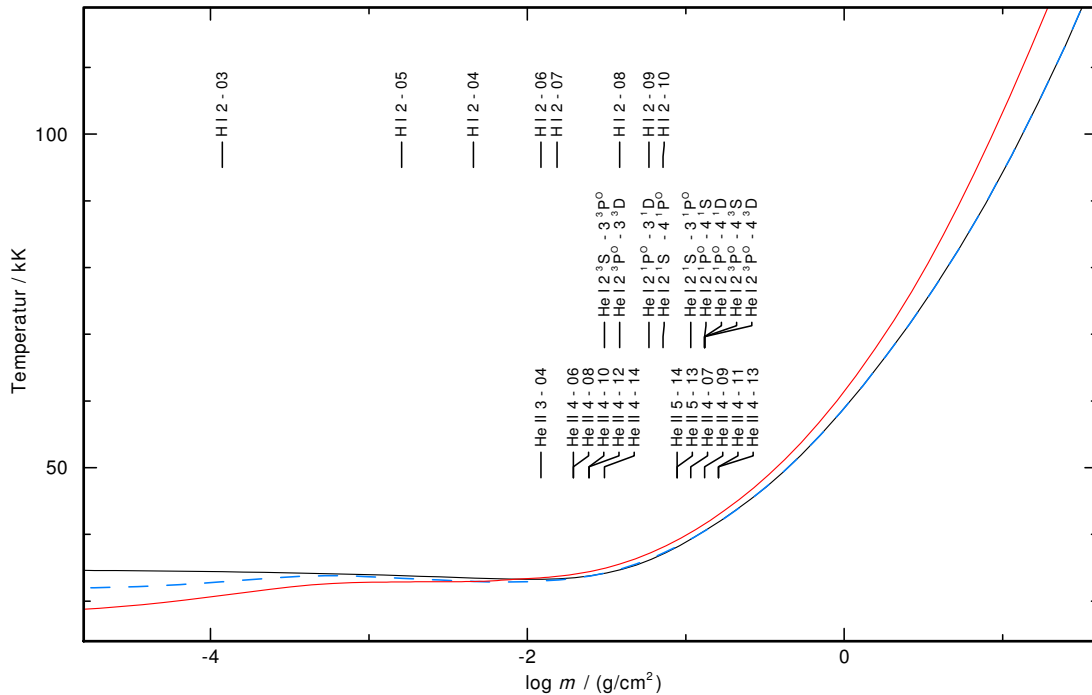


Abbildung 3.13.: Vergleich der Temperaturstrukturen des *TMAP*-Modells von Klepp & Rauch (2011) (rot,  $T_{\text{eff}} = 42\,000\text{ K}$ ,  $\log g = 5.46$ ) und des *TMAW*-Modells vor (schwarz, nur Unsöld-Lucy-Temperaturkorrektur) und nach der Einführung der Nachberechnungsschritte (blau, gestrichelt) mit den selben Parametern. Zusätzlich wurden die Entstehungstiefen der Wasserstoff- und Helium-Linienkerne eingezeichnet.

LTE-Modelle für Sternparameter, die im NLTE-Regime liegen, berechnet. Die Unterschiede zwischen Modellen mit den selben Parametern, erzeugt mit LTE- oder NLTE-Codes, sind nicht zu vernachlässigen. Dies ist z. B. in Abb. 3.14 zu sehen. Die Abbildung zeigt Linienprofile, die mit LTE- und NLTE-Bedingungen berechnet wurden. Da für die Berechnung der selbe Sternatmosphärencode verwendet wurde, stammen die Unterschiede allein von LTE - NLTE-Abweichungen. Wird die Abschätzung auf NLTE-Modelle begrenzt, so ergeben sich systematische Abweichungen zwischen den einzelnen Codes von etwa 5%. Die zugrunde liegenden Atomdaten selbst können Ungenauigkeiten von bis zu 15% haben. Die Verwendung unterschiedlicher Frequenzgitter kann zu Abweichungen von bis zu 5% führen und auch eine unterschiedliche Wahl der Tiefenskala trägt mit bis zu 10% zu den Ungenauigkeiten bei. Insgesamt sind so bei jeder Berechnung einer NLTE-Sternatmosphäre Ungenauigkeiten von etwa 10% enthalten. Wird in diesem Zusammenhang die Genauigkeit von *TMAW*-Spektren betrachtet, so ist *TMAW* durchaus für zuverlässige Spektralanalysen von hochauflösten Beobachtungen in allen Wellenlängen geeignet.

Dieser erste Test ist ein Beispiel für eine Qualitätskontrolle, wie sie an theoretischen Daten durchgeführt werden kann. Wie diese global auszusehen hat – ob eine zentrale Kontrollinstanz eingerichtet wird, jeder Veröffentlichung selbst eine Kontrolle durchführen sollte oder auf irgendeine andere Art und Weise – übersteigt den Rahmen dieser Arbeit. In diesem Fall hat der Benchmarktest eine weitere Verbesserung des *TMAW*-Dienstes bewirkt, durch die nun Spektralanalysen mit noch höherer Genauigkeit möglich sind. Die dafür eingeführten Nachberechnungsschritte sind, verglichen mit der Gesamtzeit der Berechnung, schnell aber sehr effektiv, wie z.B. in Abb. 3.15 zu erkennen ist. Diese Abbildung zeigt keine genaue Statistik der Rechenzeiten, sondern nur den Trend (Verhältnis der Gesamtzeit zur Nachberechnungszeit bzw. steigende Rechenzeit für die Berücksichtigung mehrerer Elemente), ermittelt aus einer zufällig ausgewählten Berechnung. Auch der Vergleich zwischen IAAT- und *AstroGrid-D*-Rechenzeiten fällt je nach Auslastung der jeweiligen Cluster unterschiedlich aus und ist hier nur für eine Testrechnung dargestellt.

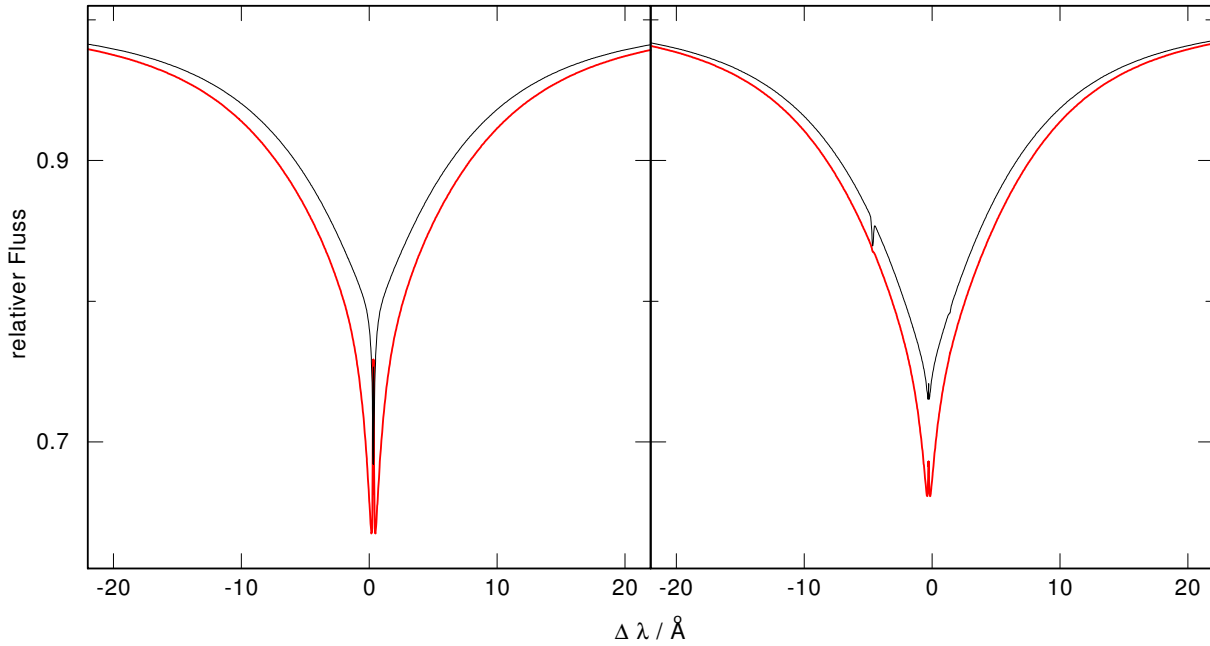


Abbildung 3.14.: Vergleich der H I-Linien bei  $4861 \text{ \AA}$  ( $H\beta$ , links) und bei  $4102 \text{ \AA}$  ( $H\delta$ , rechts) berechnet mit *TMAP* mit NLTE- (rot, dick) und mit LTE-Annahmen (schwarze, dünne Linie). Das Modell wurde für  $T_{\text{eff}} = 42\,000 \text{ K}$  und  $\log g = 5.46$  berechnet.

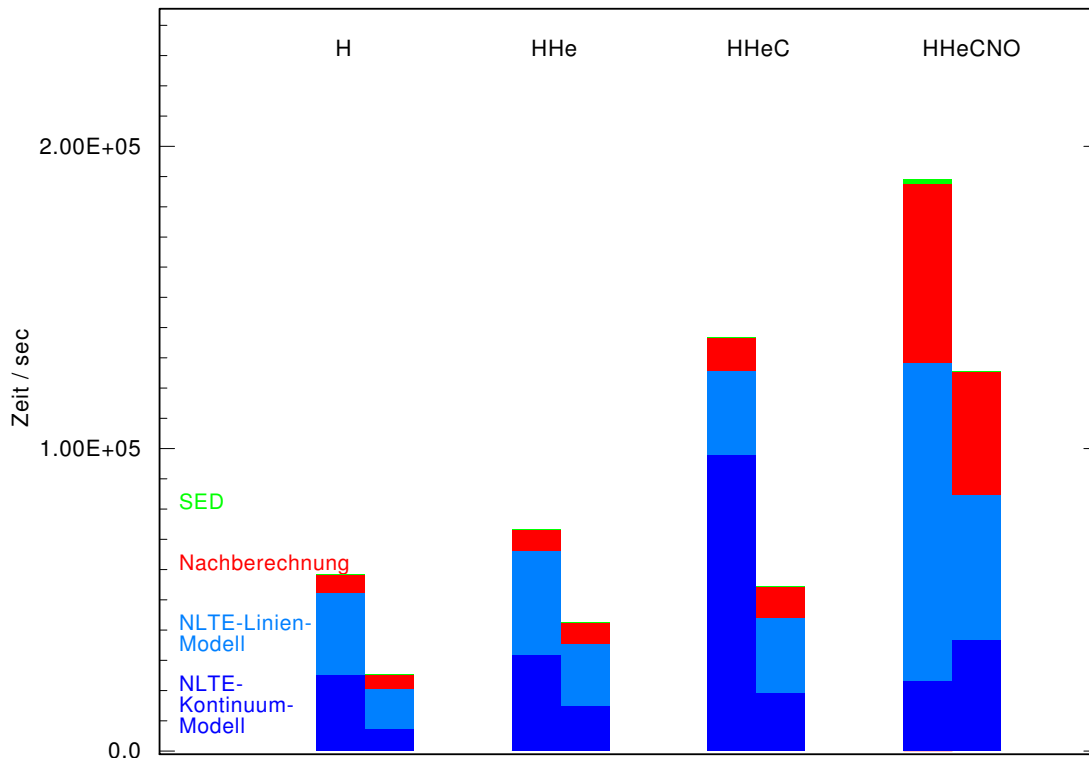


Abbildung 3.15.: Rechenzeiten verschiedener *TMAP*-Modelle mit unterschiedlichen Elementen (H, HHe, HHeC, HHeCNO) im Vergleich. Die Säulen auf der linken Seite stehen jeweils für Rechenzeiten (wall-clock times) am IAAT-Cluster auf dem *GAVO* @IAAT-Server, die auf der rechten Seite für über das *AstroGrid-D* zugängliche Rechner.

### 3.5.2. Weitere Änderungen am TMAW-Dienst im Rahmen der Einführung einer Qualitätskontrolle

Im letzten Abschnitt wurden die Änderungen an TMAW im Rahmen des ersten Benchmarktests beschrieben. Um die damit eingeführte Qualitätskontrolle auszubauen und die Arbeit des TMAW-Administrators zu erleichtern, wurden einige – zuvor meist von Hand durchgeführten – Kontrollschritte automatisiert und in das TMAW-Verfahren eingebaut. Dies vereinfacht und verbessert die Kontrolle der einzelnen Anfragen und ermöglicht eine benutzerfreundlichere Bedienung von außerhalb.

- Für eine erste Übersicht über den Parameterbereich der Modelle eines Requests werden diese in einem  $\log T_{\text{eff}}\text{-}\log g$ -Diagramm dargestellt, das den Bereich umfasst, für den TMAP zuverlässige Ergebnisse liefern kann (Abb. 3.16). Ist die Berechnung abgeschlossen, werden alle resultierenden Modelle ebenfalls in dieser Übersicht dargestellt.

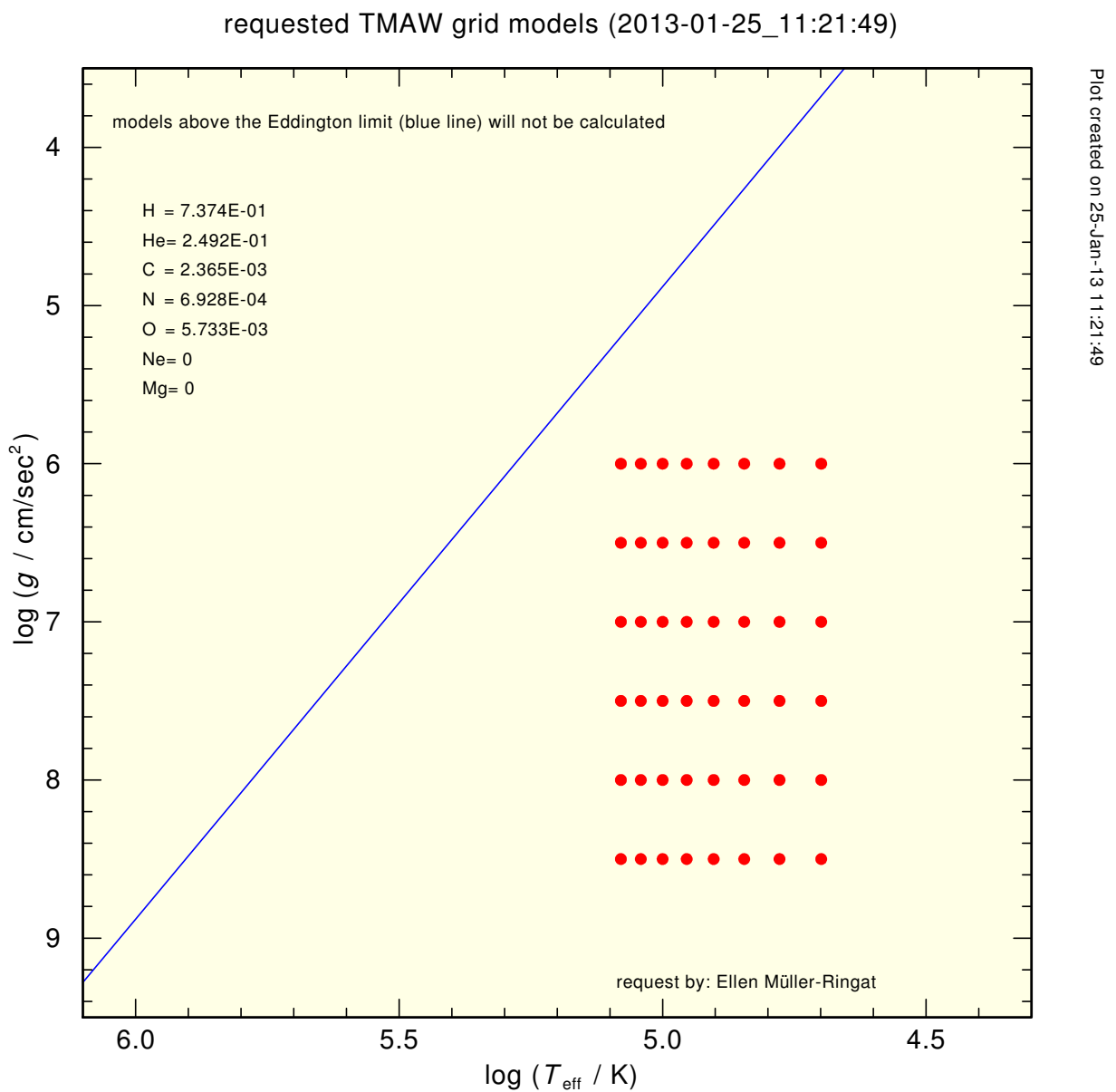


Abbildung 3.16.:  $\log T_{\text{eff}}\text{-}\log g$ -Diagramm, wie es bei der Anfrage eines Benutzers zur Übersicht über die angeforderten Modelle erzeugt wird.

- Um den Verarbeitungsprozess der Requests zu beschleunigen, werden Requests mit Parametern außerhalb des Bereichs, für den *TMAP* entwickelt wurde, nicht mehr erzeugt und berechnet. Dies betrifft z.B. Modelle mit Parametern, die oberhalb des Eddingtonlimits liegen.
- Der *TMAW*-Administrator wird durch Emails über den aktuellen Stand der jeweiligen Rechnungen informiert. Während früher die Modelle von Hand auf vollständige Berechnung oder Fehlschlag überprüft wurden, bekommt der *TMAW*-Administrator heute Informationen über die Konvergenzstufe, die das Modell erreicht hat oder ob – und wenn ja welche – Probleme bei der Berechnung aufgetreten sind. Dies passiert, wenn das Modell nicht alle beabsichtigten Rechenschritte durchläuft, wenn die Parameter der ausführbaren Programmdatei falsch eingestellt sind, oder ein Fehler bei der Berechnung des synthetischen Spektrums auftrat.
- Eine große Zeitersparnis im Vorbereitungsprozess der Modelle konnte erreicht werden, indem eine globale ausführbare Programmdatei anstelle einer individuellen für jeden Request erzeugt und verwendet wird. Mit dieser Änderung muss das Programm nicht mehr für jedes Modell kompiliert werden, sondern nur noch wenn es wirklich nötig ist.

Zusätzlich zu diesen Änderungen an den *TMAW*-Prozeduren wurden weitere Prozeduren erzeugt, um eine regelmäßige Kontrolle des *TMAW*-Dienstes zu gewährleisten. Diese werden zum Teil automatisch, zum Teil nach Bedarf von Hand gestartet und ersetzen die Überprüfung der jeweiligen Bereiche von Hand.

- Regelmäßig wird auf IAAT- und über das *AstroGrid-D* erreichbaren Rechnern automatisch (Cron-Job) ein kleines Wasserstoff-Modellgitter (neun Modelle) berechnet und die Ergebnisse werden mit früheren Rechnungen verglichen. Weichen diese voneinander ab, so hat sich eine fundamentale Einstellung des jeweiligen Clusters geändert, die überprüft werden muss. Der *TMAW*-Administrator wird darüber per Email informiert. Die zugehörige Prozedur ist im Verzeichnis `/home/TMAW/scripts/` bzw. `/home/tmaw/TMAP/` zu finden und heißt `checktestgridH.sh` bzw. `checkgridH.bat`.
- Eine weitere, automatisch ausgeführte Prozedur (`jobcontrol.sh`) überprüft einmal pro Tag, ob es Modelle gibt, die die Berechnung abgebrochen haben ohne den *TMAW*-Administrator zu informieren. Sollte dies der Fall sein, erfolgt die Benachrichtigung darüber ebenfalls als Email. Müssen Modelle neu gestartet werden, so können die Prozeduren `restart.bat`, `restartall.bat` oder `delete_lastline.sh` zur Hilfe genommen werden.
- Alle Parameterwerte werden automatisch einmal pro Woche initialisiert (`resetPARAM.sh`), um zu vermeiden, dass die Programmdateien unnötig groß werden. Außerdem wird durch ein automatisches Kompilieren (einmal pro Woche) garantiert, dass für die Berechnung jeweils die neueste *TMAP*-Version verwendet wird (`compile_MASTER.sh`).
- Um einen Einblick in den aktuellen Stand der Berechnung der Modelle zu bekommen, wurde die Prozedur `status_check.sh` entwickelt, die von Hand gestartet werden kann, um den jeweiligen Rechenschritt (siehe auch Tabelle 3.3) aller laufenden Modelle anzuzeigen.
- Sind alle Modelle eines Request abgearbeitet, kann mit `countresults.Linux_X86` nachgeprüft werden, welche Modelle wirklich Ergebnisse geliefert haben. Die nicht weiter benötigten Zwischenmodelle können mit `models_cleanup.sh` gelöscht und das konvergente Modell zusammen mit den verwendeten Atomdaten zu einem komprimierten tar-Archiv gepackt werden.
- Mit der Prozedur `maintenance.sh` kann das Bearbeiten und Absenden von Requests unterbrochen werden, um Wartungen am Dienst durchzuführen.

### 3.6. TMAW-Rechnungen über das *AstroGrid-D*

Wie bereits erwähnt, werden die Berechnungen der Modelle auch auf über das *AstroGrid-D* zugänglichen Rechnern durchgeführt, sobald die Zahl der Modelle eines Requests ein bestimmtes Limit (momentan 150) übersteigt. Wenn das der Fall ist, wird die Request-Datei, die mit den Daten von der Webseite erzeugt wird, anders benannt. Wie im Fall von IAAT-Rechnungen startet die Prozedur `checkreq.sh`, führt dann aber nicht die `TMAW_iaat_xxxx.sh`-Prozeduren aus, sondern die entsprechenden `TMAW_grid_xxxx.sh`-Prozeduren.

#### Überblick

Zu Beginn dieser Arbeit war der Sende- und Rechenprozess an bzw. über das *AstroGrid-D* gerade für ein kleines Modellgitter aus reinen Wasserstoffmodellen eingerichtet (Abb. 3.17). Im Rahmen dieser Arbeit wurde er erweitert und verbessert. Die hier beschriebenen Prozeduren waren bereits zu Beginn dieser Arbeit vorhanden und wurden zum Teil modifiziert. Der Vorbereitungsprozess läuft zunächst genau wie für die Berechnungen am IAAT ab. Die Atomdaten, Frequenzgitter und Programmdateien werden auf IAAT-Rechnern erzeugt. Da die Berechnung aber nicht am IAAT stattfindet, werden diese Input-Daten zusammen mit der Prozedur `TMAW_grid_jobfile.sh` in ein komprimiertes tar-Archiv gepackt, das in ein speziell dafür vorgesehenes Warteverzeichnis verschoben wird.

Ein Zugriff von *AstroGrid-D*-zugänglichen Clustern auf IAAT-Rechner muss ausgeschlossen werden. Deshalb ist einer der IAAT-Rechner (*AstroGrid-D* @IAAT-Server), der ausschließlich für den Datentransfer von und zu *AstroGrid-D*-zugänglichen Ressourcen zuständig ist, speziell eingerichtet. Er befindet sich außerhalb der IAAT-Firewall, damit die übrigen IAAT-Rechner weiterhin geschützt bleiben. Auf diesen Computer werden die zur Berechnung nötigen komprimierten tar-Archive transferiert. Auf diesem Rechner wird anschließend eine Perl-Prozedur (`/home/tmaw/TMAP/tmap_grid.pl`) zum Senden und Auspacken der komprimierten tar-Archive und zum Starten der Berechnung (die Prozedur `TMAW_grid_jobfile.sh` wird gestartet) auf den *AstroGrid-D*-zugänglichen Rechnern verwendet. Nach der Berechnung der Spektren sorgt die Perl-Prozedur dafür, dass alle verwendeten und erzeugten Daten wieder in einem komprimierten tar-Archiv verpackt und zurück zum *AstroGrid-D* @IAAT-Server gesendet werden.

Das Übertragen der Dateien auf den Rechner außerhalb der Firewall und das Starten der Perl-Prozedur erfolgen automatisch durch eine Unix-Shell-Prozedur (`/home/tmaw/TMAP/scp2li04.bat`), die außerdem auch die nötigen Verzeichnisse zum Speichern der Daten auf dem *AstroGrid-D* @IAAT-Server erzeugt (jeweils ein \*.in- und ein \*.out-Verzeichnis).

Mit der Perl-Prozedur `/home/tmaw/TMAP/job_status.pl`, die anzeigt, ob die Rechnungen noch aktiv sind, kann der Status der Rechnungen geprüft werden. Um auf den *AstroGrid-D*-zugänglichen Rechnern „aufzuräumen“, also zum Beispiel Reste abgebrochener Jobs zu entfernen, kann die Perl-Prozedur `/home/tmaw/TMAP/job_cleanup.pl` verwendet werden.

Über das *AstroGrid-D* hat der Nutzer Zugriff auf mehrere Cluster. Die Berechnungen werden jeweils auf dem Ersten aus der Liste `/home/tmaw/TMAP/grid_cluster_hosts.csv` gestartet. Nach Ende aller Rechnungen können die Daten vom *AstroGrid-D* @IAAT-Server mit Hilfe der Prozedur `/home/tmaw/TMAP/scp2IAAT.bat` auf die IAAT-Rechner kopiert und dort entpackt werden. Mittlerweile werden `scp2li04.bat` und `scp2IAAT.bat` automatisch alle 15 Minuten ausgeführt, um eventuell auf die Berechnung wartende Requests zu bearbeiten.

#### Erweiterungen

Für jede TMAP- und TMAW-Berechnung sind Daten für die RBF-Querschnitte aus dem Opacity Project nötig. Diese wurden zu Beginn dieser Arbeit zusammen mit den anderen Input-Daten bei jedem Request an die *AstroGrid-D*-zugänglichen Rechner übermittelt. Um die maximale Größe der übertragenen Datei-

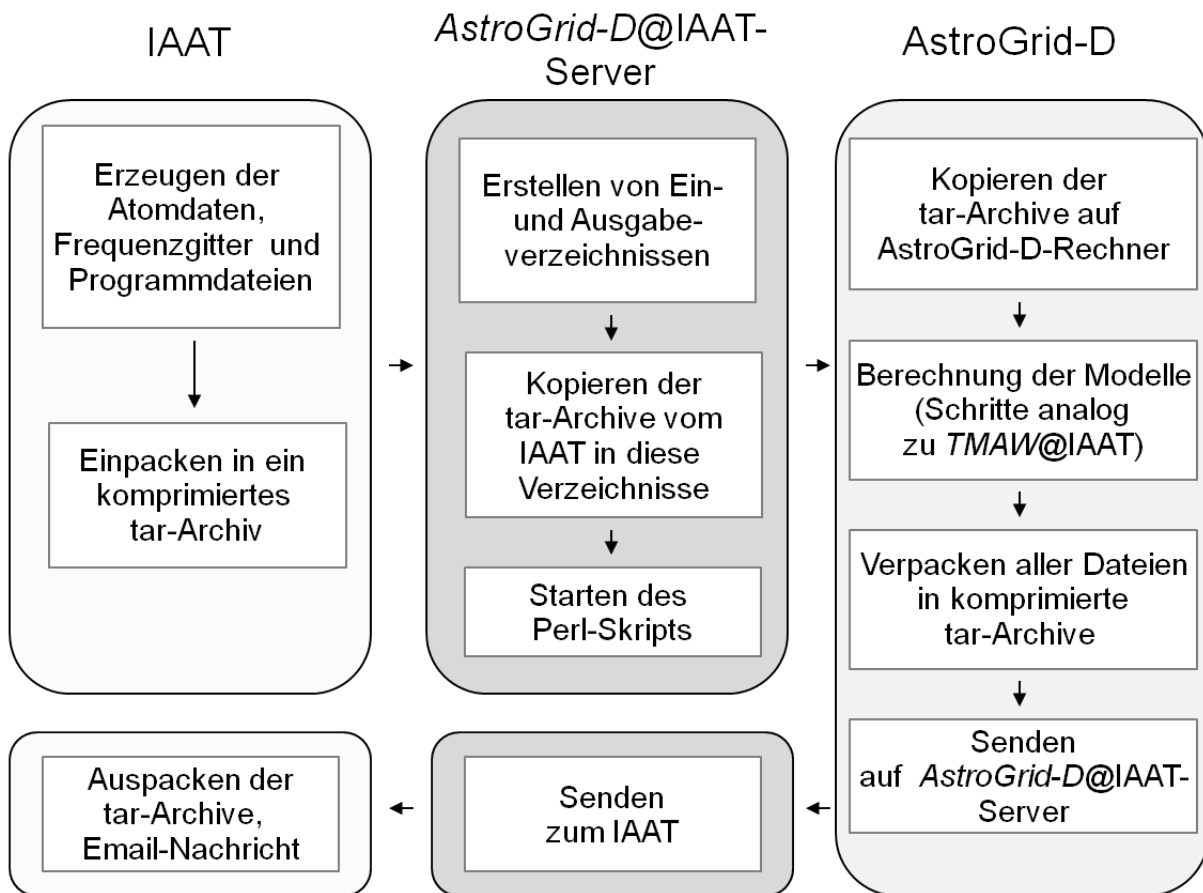


Abbildung 3.17.: Schematische Darstellung der Berechnung von *TMAW*-Requests über *AstroGrid-D*.

en nicht zu überschreiten, handelte es sich dabei um reduzierte Querschnittsdaten, was eine Nachberechnung auf IAAT-Rechnern mit den korrekten Querschnittsdaten zur Folge hatte. Da diese Nachberechnung sehr zeitintensiv war, wurden später die vollständigen, zur Berechnung nötigen Opacity Project-Daten auf den jeweiligen Frontend-Maschinen ausgewählter *AstroGrid-D*-zugänglicher Cluster gespeichert.

Ebenso wurde mit den für die Berechnung synthetischer Spektren nötigen Daten zur Verbreiterung der Linien verfahren. Damit am IAAT überhaupt keine weiteren Schritte außer dem Auspacken der tar-Archive mehr nötig sind, wurde auch das Programm *WRPLOT*<sup>16</sup>, das zur Erzeugung von grafischen Darstellungen der resultierenden Flüsse und Temperaturstrukturen verwendet wird, auf einzelnen Frontend-Maschinen installiert. Mittlerweile werden so auf den *AstroGrid-D*-zugänglichen Rechnern alle Rechenschritte analog zu denen am IAAT ausgeführt (Abb. 3.17). Die *TMAW\_grid\_run.sh*-Prozedur ist deshalb im Vergleich zur *TMAW\_iaat\_run.sh*-Prozedur um den Bereich der Modellberechnung und den anschließenden Email-Sende- und Darstellungsprozess gekürzt worden.

Für die Berechnungen über das *AstroGrid-D* wurden analog zur Ausführung am IAAT einige Kontrollinstanzen eingeführt. Einmal pro Woche wird das Wasserstofftestgitter bestehend aus neun Modellen auch auf *AstroGrid-D*-erreichbaren Rechnern berechnet (*/home/tmaw/TMAP/checkgridH.bat*) und die Ergebnisse mit früheren Rechnungen verglichen. Bei Start und Ende der Berechnungen, sowie bei Abweichungen der Ergebnisse wird der *TMAW*-Administrator informiert.

Um einen Überblick über den Erfolg der Rechnungen über das *AstroGrid-D* zu erhalten, wurde die Prozedur */home/tmaw/TMAP/checkquality.bat* erzeugt, die alle tar-Archive eines Requests auspackt und jeweils prüft ob die Modelle konvergiert oder fehlgeschlagen sind und wie stark die Effek-

<sup>16</sup>[http://astro.uni-tuebingen.de/~rauch/TMAP/WRPLOT/WRPLOT\\_Guide.html](http://astro.uni-tuebingen.de/~rauch/TMAP/WRPLOT/WRPLOT_Guide.html)



tivtemperatur des jeweiligen Modells vom geforderten Wert abweicht.

Die Berechnungen auf den über das *AstroGrid-D*-erreichbaren Clustern liefen nicht immer reibungslos ab. Durch Änderung der auf *AstroGrid-D*-Rechnern installierten Programme bzw. Programmversionen und aufgrund des unterschiedlich limitierten Speicherplatzes der *AstroGrid-D*-Nutzer, erfüllten einige Rechencluster die für die Berechnung der *TMAW*-Modelle notwendigen Anforderungen nicht mehr. Je nach Auslastung der Cluster, brachen die *TMAW*-Rechnungen zum Teil mitten in der Berechnung ab, ohne dass sie numerisch instabil waren. Die Liste der Cluster, die alle Voraussetzungen erfüllen und auf denen die *TMAW*-Rechnungen zuverlässig laufen, wurde immer kleiner (`grid_cluster_hosts.csv`). Momentan wird nur noch ein Cluster verwendet, dieser stellte sich aber als sehr zuverlässig heraus.

Als Reaktion auf diese Schwierigkeiten wurde die Zahl der Modelle, ab denen ein Request über das *AstroGrid-D* berechnet werden soll, auf 150 erhöht. Außerdem wurde am IAAT ein *GAVO@IAAT*-Server mit zwei Hyper-Threading-fähigen Intel-Prozessoren angeschafft, auf dem bis zu 24 Modelle gleichzeitig berechnet werden können. Mit diesen Änderungen können auch große Gitter am IAAT schnell berechnet werden.

Seit Mitte 2012 wird das *AstroGrid-D* nicht mehr finanziert. Der Zugang zu den *AstroGrid-D*-Rechnern wird deshalb bald nicht mehr gewährleistet sein. Als Alternative bietet sich das *bwGRiD*<sup>17</sup> an. Einzelne Anfragen wurden bereits auf dem *bwGRiD* berechnet. Für eine Umstellung von *AstroGrid-D* auf *bwGRiD*-Rechner müssen nur die Skripte zum Kopieren der tar-Archive von und zum *bwGRiD* sowie die Skripte zum Starten der einzelnen Requests angepasst werden, die Rechnungen können dann analog zu denen auf dem *AstroGrid-D* ausgeführt werden.

---

<sup>17</sup><http://www.bw-grid.de>, Teil der Deutschen D-Grid Initiative und finanziert vom Bundesministerium für Bildung und Forschung und vom Ministerium für Wissenschaft, Forschung und Kunst Baden-Württemberg



## 4. GAVO-Dienste in der Spektralanalyse

Durch die in den letzten beiden Kapiteln beschriebenen *GAVO*-Dienste kann jedermann über das *VO* Spektralanalysen auf unterschiedlichen Niveaus durchführen. Es stellt sich jedoch die Frage, wie wichtig diese Dienste für die Spektralanalyse sind und mit welcher Genauigkeit welche Ergebnisse mit diesen Diensten erzielt werden können. Sind die aus *TMAW* resultierenden Spektren überhaupt für präzise Spektralanalysen zu verwenden? Sind sie nur für eine erste, grobe Voranalyse geeignet? Oder kommen sie nur als Input für Photoionisationsmodelle in Frage? Macht die, im Vergleich zu *TMAP*-Rechnungen, stark verkürzte Rechenzeit von *TMAW* die Ergebnisse unrealistisch? Können sowohl der UV- als auch der optische Wellenlängenbereich mit ähnlicher Genauigkeit analysiert werden? Sind die resultierenden Spektren für alle Objekte mit Parametern, die im NLTE-Bereich liegen, geeignet? Wie wichtig ist die Berücksichtigung der Eisengruppenelemente für unterschiedliche Arten von Objekten in unterschiedlichen Sternparameterbereichen? Können bisher unerklärbare Beobachtungen, für deren Ursache Eisengruppenelemente vermutet werden, mit den aktualisierten Atomdaten reproduziert werden?

Um diese Fragen zu beantworten, wurden der *TMAW*-Dienst und *IrOnIc* ausgiebig getestet. Dafür wurden verschiedene Objekte ausgewählt, die aus unterschiedlichen Spektralklassen stammen und somit verschiedene chemische Zusammensetzungen und Temperaturen haben. Sterne der ausgewählten Klassen haben sich nicht – wie etwa 3/4 der Sterne mittlerer Anfangsmasse – nach den Standardsternentwicklungswegen entwickelt, sondern zeigen Atmosphären, denen andere Entwicklungsszenarien zugrunde liegen müssen. Ihre Modellierung ist deshalb zum Teil schwieriger als die wasserstoffdominierter, „normaler“ Sterne. Diese Objekte sind hervorragend geeignet, um die Stabilität der *TMAW*-Rechnungen und die Auswirkungen der kleineren Modellatome, die vor allem für He, C, N, und O Unterschiede zu *TMAP*-Modellatomen aufweisen, auf die Sternatmosphäre zu untersuchen. Für einige der ausgewählten Objekte wurde außerdem eine ungewöhnlich hohe Eisengruppenhäufigkeit als Erklärung für ihren beobachteten, extrem flachen Flussverlauf im UV vermutet. Die einzelnen Objekte werden deshalb in den folgenden Kapiteln vorgestellt und ihre Analyse mit den *GAVO*-Diensten sowie die Ergebnisse beschrieben. Begonnen wird mit unterleuchtkräftigen Sternen mit beinahe reiner Heliumatmosphäre (Kapitel 4.1), dann werden PG 1159-Sterne (Kapitel 4.2) mit kohlenstoffdominierten Atmosphären getestet. Für zusätzliche Tests der Auswirkungen der Eisengruppenelemente auf Objekte mit zum Teil extrem hohen Eisengruppenhäufigkeiten werden wasserstoffreiche unterleuchtkräftige Sterne (Kapitel 4.3) und ein DO-weißer Zwerg (Kapitel 4.4) herangezogen.

### 4.1. Die Heliumreichen Unterleuchtkräftigen Sterne LSE 153 und LSE 263

LSE 153 und LSE 263 wurden für den ersten der hier beschriebenen Tests ausgewählt. Sie gehören zur Klasse der heliumdominierten unterleuchtkräftigen Sterne (He-sdOs, Husfeld et al., 1989). Mit ihren  $T_{\text{eff}}$  von ca. 60 000 K liegen sie im mittleren Temperaturbereich aller gewählten Testobjekte. Ihr  $\log g \approx 5$  gehört zu den niedrigeren Oberflächenschwerebeschleunigungen aller getesteten Objekte. Der Fokus der Analyse dieser beiden Objekte wird auf den optischen Bereich gelegt, für die Abschätzung der Größenordnung der Eisengruppenhäufigkeiten wird der UV-Bereich hinzugezogen. Ein Vergleich der resultierenden Modelle mit *TMAW*-Spektren liefert die Genauigkeit einer *TMAW*-Analyse in diesem Parameterbereich. Da diese Klasse von Objekten auch in Kapitel 4.3 betrachtet wird, wird hier zuerst ein Überblick über die Eigenschaften und Entwicklung heißer unterleuchtkräftiger Sterne gegeben.

#### 4.1.1. Eigenschaften und Entwicklung unterleuchtkräftiger Sterne

Heiße unterleuchtkräftige Sterne (so genannte subdwarfs, sd) können in zwei Unterklassen eingeteilt werden. Die kühleren, meist von einer Wasserstoffhülle umgebenen, unterleuchtkräftigen B-Sterne (sdBs), die Heliumkernbrennen vollziehen, und die heißere, inhomogene Gruppe der unterleuchtkräftigen O-Sterne (sdOs, Verhältnis sdB:sdO = 3:1). Ein Überblick über heiße unterleuchtkräftige Sterne wird z.B. von Heber (2009) gegeben.

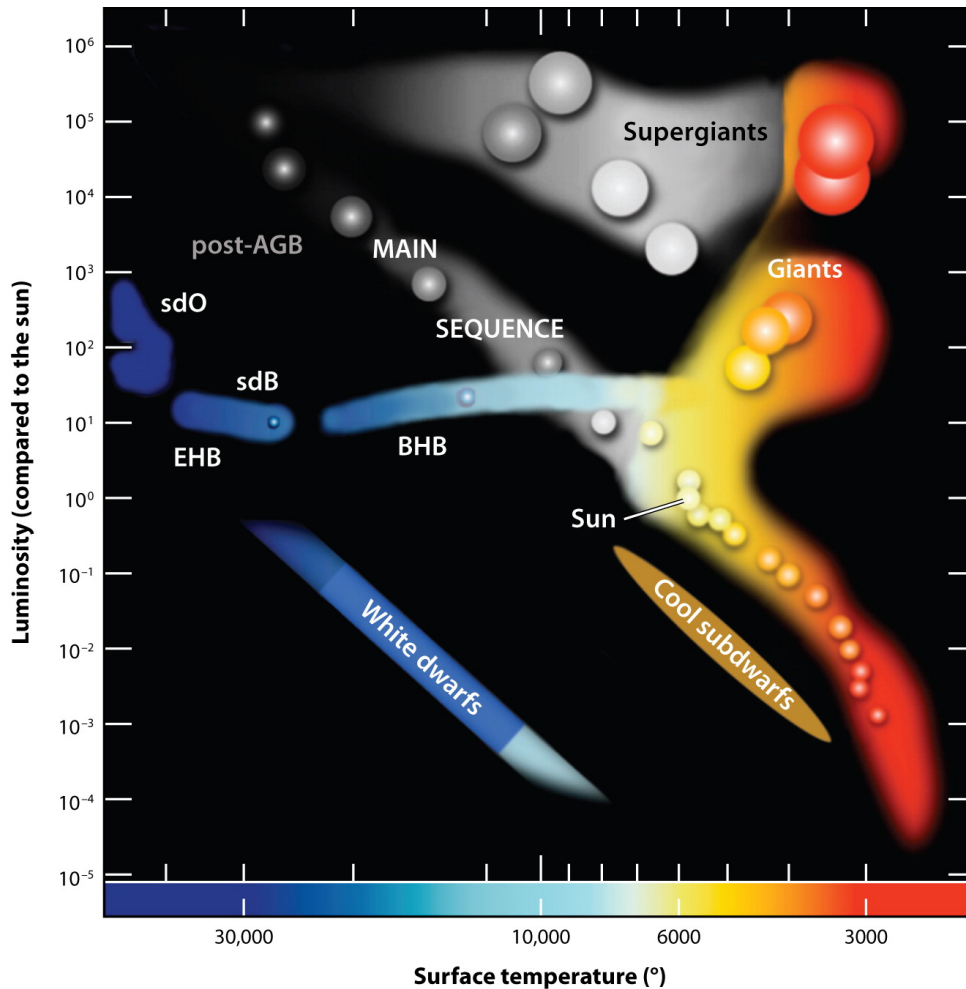
Die Spektren der sdB-Sterne zeigen vor allem breite, tiefe Balmerlinien und befinden sich, aufgrund des Heliumkernbrennens, im HRD meist auf einer Erweiterung des Horizontalast, dem extremen Horizontalast (extreme horizontal branch, EHB, Abb 4.1). Sie haben Effektivtemperaturen von 25 000 K bis zu über 50 000 K und  $\log g$ -Werte zwischen 4.0 und 6.5. Die meisten sdBs sind heliumarm, manche zeigen jedoch schwache He I-Linien. Spektren, die denen der sdBs ähnlich sehen, zusätzlich aber noch He II-Linien zeigen, gehören zu den sdOB-Sternen.

sdO-Sterne haben Spektren, in denen die He II  $\lambda 4686$ -Linie und eventuell weitere Heliumlinien zu sehen sind. Manche sdO-Spektren zeigen zusätzlich H I-Balmerlinien, die meisten sdOs sind jedoch (im Gegensatz zu den sdBs) heliumreich (so genannte He-sdOs). Diese heliumreichen sdOs zeigen zusätzlich entweder eine erhöhte Stickstoff- oder Kohlenstoffhäufigkeit. Insgesamt bilden die sdO-Sterne im Gegensatz zu den sdB-Sternen keine homogene Klasse. Die heliumreichen sdOs bevölkern ein schmales Gebiet im HRD, die heliumarmen sind weit verstreut, was darauf schließen lässt dass sie aus verschiedenen Entwicklungssequenzen stammen. Im Mittel sind sie mit  $40\,000\text{ K} \leq T_{\text{eff}} \leq 90\,000\text{ K}$  heißer als die sdB-Sterne, überdecken jedoch einen ähnlichen  $\log g$ -Bereich.

In ihrer Entwicklung verlassen die sdBs die normale Entwicklungssequenz nach dem Roten-Riesen-Stadium in Richtung der Weißen Zwerge. Dafür müssen sie in der Roten-Riesen-Phase (RGB-Phase) erhöhten Massenverlust gehabt haben, wodurch die Wasserstoffhülle zu klein ( $< 10^{-4}M_{\odot}$ ) für weiteres Wasserstoffbrennen wurde. Die Ursache für diesen Massenverlust ist unklar, allerdings könnte ein Begleiter (Stern oder Planet) dabei eine wichtige Rolle spielen, da für ca. zwei Drittel aller sdBs ein solcher gefunden wurde (Maxted et al., 2001). Für ähnlich viele heliumarme sdOs wurde ebenfalls ein Begleiter nachgewiesen und ihre Atmosphären scheinen wie die der sdBs von Diffusionsprozessen bestimmt zu sein, was auf eine Entwicklungssequenz von sdBs zu heliumarmen sdOs schließen lässt. Zudem konnte auch theoretisch gezeigt werden, dass in einem engen Doppelsternsystem genügend Massenverlust stattfinden kann, sodass nur eine dünne Wasserstoffhülle nach dem Roten-Riesen-Stadium bleibt (Mengel et al., 1976). Es existieren jedoch mehrere Kanäle der Doppelsternentwicklung, die zu einem sd-Stern führen (Han et al., 2002, 2003). Im Gegensatz zur großen Zahl beobachteter Begleiter von sdB-Sternen sind nur wenige He-sdOs radialgeschwindigkeitsvariabel. Sie haben daher eher keinen Begleitstern oder -planeten.

Ein Szenario, das sowohl die Entstehung von einzelnen sdBs als auch die der (He-)sdOs erklären könnte ist das Hot-Flasher-Szenario (z.B. D’Cruz et al., 1996, 2000; Castellani & Castellani, 1993; Lanz et al., 2004; Miller Bertolami et al., 2008). In diesem verlässt der Stern den normalen Entwicklungsweg aufgrund sehr hohen Massenverlusts nach der RGB-Phase und erfährt später (analog zum „born again“ Szenario der PG 1159-Sterne, siehe Kapitel 4.2) einen Helium-Flash, wodurch er wieder zum Roten Riesen wird. Im Gegensatz zu den PG 1159-Sternen wird jedoch nicht das Heliumschalenbrennen, sondern das Heliumkernbrennen gezündet. Je nachdem wann sich der Flash ereignet, zeigt der Stern danach die typische Wasserstoff-Helium-Mischung der sdBs („früher“ Flash), leicht erhöhte Helium- und Stickstoffhäufigkeiten („später“ Flash mit „shallow mixing“) oder stark erhöhte Helium-, Kohlenstoff- und Stickstoffhäufigkeiten („später“ Flash mit „deep mixing“, wobei der Wasserstoff verbrannt wird und ein He-sdO entstehen könnte). Die von diesem Szenario vorhergesagten  $\log g$ -Werte sind allerdings höher als die beobachteten.

Ein Szenario zur Entstehung beider Arten von sd-Einzelsternen ist das Merger-Szenario. Eine Verschmelzung zweier Weißer Zwerge oder eines Weißen Zwergs und eines (post-)sdB-Sterns (Merger) kann zu einem He-sdO führen (z.B. Saio & Jeffery, 2000; Justham et al., 2011; Zhang & Jeffery,



 Heber U. 2009.  
 Annu. Rev. Astron. Astrophys. 47:211–51

Abbildung 4.1.: Hertzsprung-Russell-Diagramm (Heber, 2009). Eingezeichnet sind auch der extreme Horizontalast (EHB) sowie die Lage der sdB- und sdO-Sterne.

2012). Die Atmosphäre zeigt nach dem Mergerprozess erhöhte Häufigkeiten von Elementen aus dem CNO-Prozess. Allerdings wird die vorhergesagte, schnelle Rotation nicht beobachtet und die beobachteten Temperaturen sind heißer als prognostiziert. Wenn stattdessen ein sdB-Vorläufer im Roten Riesen-Stadium mit einem Braunen Zwerg<sup>1</sup> oder Planeten während der Common Envelope-Phase<sup>2</sup> merged, kann ein sdB-Stern entstehen. Für sdO-Sterne sind außerdem auch Entwicklungswege über den asymptotischen Riesenast möglich.

Die Atmosphären der sdB-Sterne zeigen im Vergleich zu anderen (prä-) Weißen Zwergen ungewöhnliche Elementhäufigkeiten. Eine Erklärung dafür könnten Diffusionsprozesse sein (z.B. Michaud et al., 2008, 2011), die durch ein Gleichgewicht zwischen gravitivem Absinken und radiativem Auftrieb schwere Elemente an der Sternoberfläche vermehrt anlagern. Eisengruppenelemente und schwerere bis zu Pb wurden bereits supersolar in unterleuchtkräftigen Sternen beobachtet (bis auf Eisen selbst, was

<sup>1</sup>Ein Brauner Zwerg ist zu massearm ( $< 0.08 M_{\odot}$ ), um Wasserstoffbrennen zünden zu können.

<sup>2</sup>In der Doppelsternentwicklung kommt es oft (je nach Masseverhältnis der beiden Komponenten) zu einer Phase instablen Massentransfers vom schwereren zum leichteren Objekt. Da das leichtere die Masse nicht aufnehmen kann, umgibt sie beide Komponenten des Systems als so genannter Common Envelope.

meist solar oder subsolar auftritt), während C und O subsolar sind (z.B. O’Toole & Heber, 2006; Blanche et al., 2008). Dieser Trend ist in Übereinstimmung mit von Diffusionsmodellen vorhergesagten Häufigkeiten. Allerdings wurde beobachtet, dass die Sterne trotz gleicher oder ähnlicher  $T_{\text{eff}}$ - und  $\log g$ -Werte eine große Streuung in den Häufigkeitswerten zeigen. Dieser Punkt spricht gegen die derzeitigen Diffusionsmodelle.

Die durch Diffusionsmodelle vorhergesagte He-Häufigkeit ist deutlich kleiner als die beobachtete (z.B. Heber, 2009). Durch die Berücksichtigung eines Sternwinds mit  $10^{-13} - 10^{-14} M_{\odot}/\text{Jahr}$  kann die beobachtete He-Häufigkeit allerdings erklärt werden (Fontaine & Chayer, 1997; Unglaub & Bues, 2001). Dieser Wind kann theoretisch ebenfalls für eine höhere Metallhäufigkeit in der Atmosphäre sorgen und eventuell auch für unterschiedliche Häufigkeiten bei gleichen  $T_{\text{eff}}$  und  $\log g$  verantwortlich sein. Detaillierte Modelle, die diesen Wind und alle Elemente berücksichtigen, wurden allerdings noch nicht berechnet.

Die beobachtete Eisenhäufigkeit ist, im Gegensatz zu den Häufigkeiten der anderen Eisengruppenelemente, meist (sub)solar. Eisen könnte eine wichtige Rolle bei der Entstehung von Sternpulsationen spielen und die Analyse dieses Elements ist deshalb von großer Bedeutung. Bisher wurde theoretisch ein Unterschied der Fe-Häufigkeit für Pulsatoren und nicht-pulsierende sdBs vorhergesagt. So zeigen z.B. Diffusionsrechnungen, dass pulsierende sdBs mehr Eisen als nicht-pulsierende innehaben sollten (Charpinet et al., 1997), was bisher allerdings noch nicht beobachtet wurde (z.B. O’Toole & Heber, 2006). Die Streuung der Häufigkeitswerte einzelner Sterne ist allerdings so groß, dass momentan kein Trend der Häufigkeiten schwerer Elemente von Pulsatoren und nicht-Pulsatoren beobachtbar ist. Eine genaue Bestimmung der Eisengruppenelementhäufigkeiten in diesen Sternen könnte wichtige Hinweise auf die bisher noch nicht ganz verstandenen Prozesse in den sd(O)B-Sternen liefern.

Im Gegensatz zu den eventuell bei sdB-Sternen wichtigen Diffusionsprozessen findet bei He-sdOs Konvektion statt. Diffusion sollte deshalb keine große Rolle spielen. Wie bereits erwähnt zeigen die He-sdOs entweder erhöhte Kohlenstoff- oder Stickstoffhäufigkeiten (z.B. Hirsch, 2009). Der Einfluss von Eisen und Nickel auf die Atmosphären von sdOs wurde von Haas (1997) untersucht. Die beiden He-sdOs LSE 153 und LSE 263 zeigten dabei supersolare Nickel- und solare bis subsolare Eisenhäufigkeiten.

#### 4.1.2. LSE 153 und LSE 263

Zwei sdO-Sterne mit extrem hoher Heliumhäufigkeit sind LSE 153 und LSE 263. Sie wurden im Case-Hamburg Luminous Stars Survey von Drilling (1983) entdeckt und zuerst von Schönberner & Drilling (1984) analysiert. Husfeld et al. (1989) führten eine Analyse mit HHe-NLTE-Modellen durch, in die später Kohlenstoff, Stickstoff und Magnesium mit einer Lineformationrechnung integriert wurde. Zur Analyse wurden optische CASPEC<sup>3</sup>-Spektren (aufgenommen am 3./4. April 1984) verwendet. Zusätzlich wurde ein Konsistenzcheck der Modelle mit den verfügbaren IUE<sup>4</sup>-Spektren (aufgenommen am 26. Mai 1982 (low dispersion, ID: LWR13321 und SWP17051) und am 22. September 1982 (high dispersion, ID: SWP18055 und SWP18056)) durchgeführt. Da die optischen Heliumlinien von Husfeld et al. (1989) nicht reproduziert werden konnten, nahmen sie Rotationsgeschwindigkeitskorrekturen von 20 bzw. 40 km/sec für LSE 153 bzw. LSE 263 an und korrigierten so die theoretischen Spektren. Die Ergebnisse der Analyse sind in Tabelle 4.1 dargestellt.

#### HHe-Modelle

Im Rahmen dieser Arbeit sollen die HHe-Modelle von Husfeld et al. (1989) mit *TMAP* und mit *TMAW* erneut berechnet werden, um eine Aussage über die Genauigkeit des *TMAW*-Dienstes für diesen Sternparameterbereich machen zu können. Kurz vor diesem Test wurde das *TMAD* Helium-Modellatom auf

<sup>3</sup><http://www.eso.org/sci/publications/messenger/archive/no.17-jun79/messenger-no17-27-29.pdf>

<sup>4</sup><http://science.nasa.gov/missions/iue/>

Tabelle 4.1.: Parameter von LSE 153 und LSE 263 (Husfeld et al., 1989).  $\epsilon_x$  steht für die Häufigkeiten in Teilchenzahlbruchteilen,  $X_x$  für Massenbruchteile.

	LSE 153		LSE 263	
$T_{\text{eff}}/\text{K}$	70 000		70 000	
$\log g/(\text{cm}/\text{sec}^2)$	4.75		4.90	
	$\log \epsilon$	$\log X$	$\log \epsilon$	$\log X$
H	< -1.0	< -1.7	< -1.0	< -1.7
C	-1.75	-1.3	-4.0	-3.6
N	-2.6	-2.1	-2.4	-1.8
Mg	-3.3	-2.5	-3.5	-2.8
$M/M_{\odot}$	0.7		0.6	
$\log L/L_{\odot}$	3.9		3.6	
$R/R_{\odot}$	0.58		0.45	
$M_{\text{Bol}}$	-4.9		-4.4	
$M_V$	0.6		1.0	
$d/\text{kpc}$	1.3		1.2	

den neuesten Stand gebracht und erweitert, wodurch nun für He I alle Levels bis zur Hauptquantenzahl  $n = 10$  als NLTE-Levels mit aktuellen Daten berücksichtigt werden können.

Für LSE 153 und LSE 263 wurden die optischen CASPEC-Spektren, die Husfeld et al. (1989) verwendet hatten, von U. Heber (Bamberg) an die Tübinger Arbeitsgruppe übermittelt. Mit *TMAP* wurden HHe-Modelle mit den von Husfeld et al. (1989) bestimmten Parametern gerechnet. Beim Vergleich mit der Beobachtung zeigt sich, dass die optischen He II-Linien, ähnlich dem Balmerlinienproblem (Napiwotzki & Rauch, 1994) bei der Analyse wasserstoffdominierter Spektren, nicht simultan reproduziert werden können. In den Spektren von LSE 153 tritt dieser (im Folgenden der Einfachheit halber Balmerlinienproblem genannte) Sachverhalt, aufgrund einer höheren Metallhäufigkeit stärker auf als bei LSE 263 (Abb. 4.7 und 4.8). Mit den Modellen mit Literaturwerten kann allerdings das He I / He II-Ionisationsgleichgewicht nicht reproduziert werden. Die eigentliche Temperatur der Objekte scheint tiefer zu sein, da He I  $\lambda 4471 \text{ \AA}$  im Modell viel zu schwach hervortritt. Es wurden deshalb *TMAP*-Testgitter mit unterschiedlichen  $T_{\text{eff}}$ - und  $\log g$ -Werten ( $T_{\text{eff}} = 50\,000 - 70\,000 \text{ K}$ ,  $\Delta T_{\text{eff}} = 5\,000 \text{ K}$ ,  $\log g = 4.50 - 5.50$ ,  $\Delta \log g = 0.25$ ) für eine genauere Parameterbestimmung berechnet. Modelle mit  $T_{\text{eff}} = 60\,000$  oder  $65\,000 \text{ K}$  reproduzieren das Ionisationsgleichgewicht deutlich besser. Diese Erkenntnis ist keine Überraschung, sondern wurde bereits von Haas (1997) vermutet. Er schätzte die Temperatur anhand von Ionisationsgleichgewichten der Fe- und Ni-Linien im UV-Bereich an den vorhandenen IUE-Beobachtungen ab.

Für einen Vergleich mit *TMAW*-Modellen sollten nun die Modelle des *TMAP*-Gitters mit dem *TMAW*-Dienst nachgerechnet werden. Die Modellierung der *TMAW*-Spektren gestaltete sich jedoch schwierig, da, aufgrund der bereits besprochenen vereinfachten Berechnung durch den *TMAW*-Dienst, nicht alle Modelle die Berechnung erfolgreich beendeten (Abb. 4.2). Um den *TMAW*-Dienst zu verbessern und um zu überprüfen, welche Einstellungen die *TMAW*-Rechnungen stabiler machen könnten, wurde die Berechnung der Modelle auf unterschiedliche Arten wiederholt.

- *TMAP* verwendet das Verfahren der beschleunigten Lambda Iterationen (ALI), wobei zwischen einem diagonalen und tridiagonalen  $\Lambda$ -Operator gewählt werden kann. Im normalen *TMAW*-Berechnungsverfahren wird ein diagonaler  $\Lambda$ -Operator verwendet. Damit sind die linearisierten Gleichungen unabhängig von den anderen Tiefenpunkten der Atmosphäre. Alternativ kann ein tridiagonaler  $\Lambda$ -Operator verwendet werden, wodurch die benachbarten Tiefenpunkte bei der Lösung der Gleichungen mit einbezogen werden. Bei einer erneuten Berechnung des Gitters der *TMAW*-

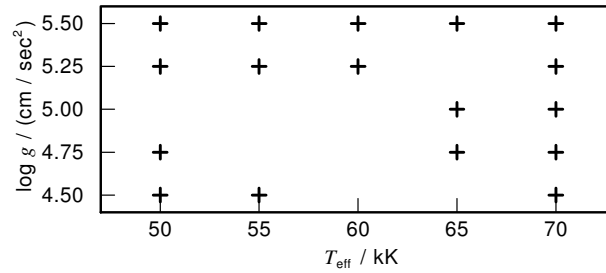


Abbildung 4.2.: Das für den *TMAP-TMAW*-Vergleich berechnete Gitter von *TMAW*-Modellen. Modellrechnungen, die mit einem + gekennzeichnet sind, beendeten die Berechnung erfolgreich.

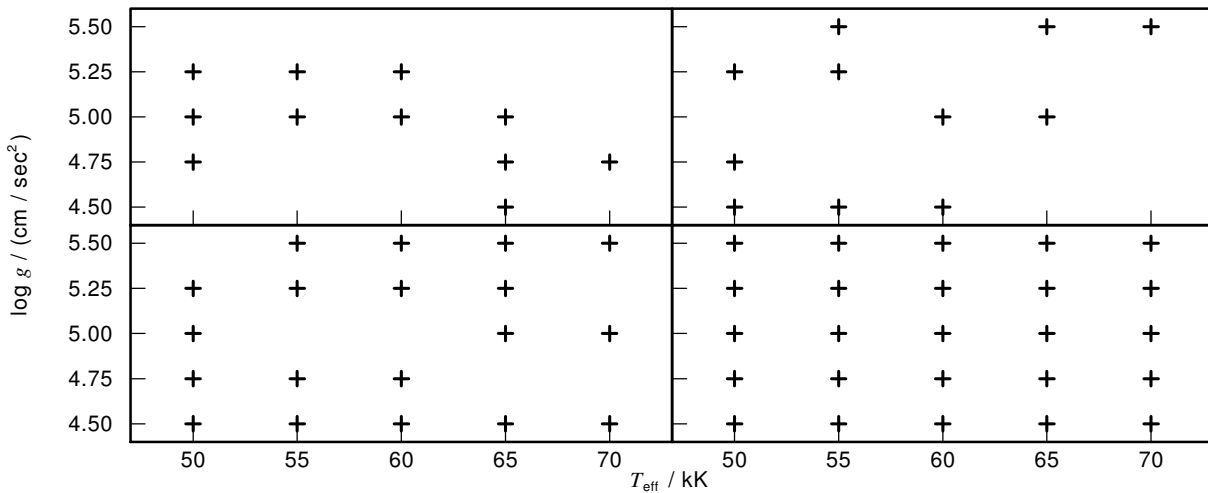


Abbildung 4.3.: Wie Abb. 4.2, bei einer abweichenden Wahl des  $\Lambda$ -Operators (oben links), der Tiefenskala (oben rechts), des Gewichts der Temperaturkorrektur (unten links) und des Newton-Raphson-Verfahren (unten rechts).

Modelle wurde nun anstelle des diagonalen, ein tridiagonaler  $\Lambda$ -Operator verwendet. Alle andere Einstellungen wurden gleich belassen, d.h. die Berechnungsschritte wurden genau wie im vorherigen Versuch wiederholt. Obwohl bei diesem Verfahren Modellrechnungen beendet wurden, die vorher abgebrochen sind, schafften nicht alle Modelle alle Berechnungsschritte (Abb. 4.3 oben links).

- In einem weiteren Test wurden die Berechnungsschritte wieder analog zum ersten Versuch durchgeführt, die Modellatmosphäre wurde aber nicht für  $\log m = -7.3 - 3.87$  sondern für  $\log m = -9.8 - 2.0$  berechnet, was für heliumdominierte Spektren realistischer ist. Dadurch beendeten zwar wieder andere Modelle als zuvor alle Berechnungsschritte, aber wie im Fall des geänderten  $\Lambda$ -Operators waren es insgesamt weniger (Abb. 4.3 oben rechts).
- Für *TMAW*-Berechnungen wird die Unsöld-Lucy-Temperaturkorrektur verwendet, da diese stabiler als die normale, von *TMAP* verwendete Temperaturkorrektur ist. Bei der Berechnung dieser Temperaturkorrektur wird der Korrekturwert mit von der optischen Tiefe abhängigen Gewichtungsfaktoren gedämpft. Einer dieser Gewichtungsfaktoren wurde in einem weiteren Test um einen Faktor zehn verkleinert. Dieser Test führte bisher zu den meisten die Berechnung beendenden Modellen (Abb. 4.3 unten links).
- Um die Rechenzeit so kurz wie möglich zu halten, können in *TMAP*-Rechnungen verschiedene Verfahren zur Lösung der Gleichungssysteme für die Nebenbedingungen verwendet werden. Um eine annehmbare Rechenzeit zu garantieren, wird in *TMAW* normalerweise die Jacobi-Matrix nur



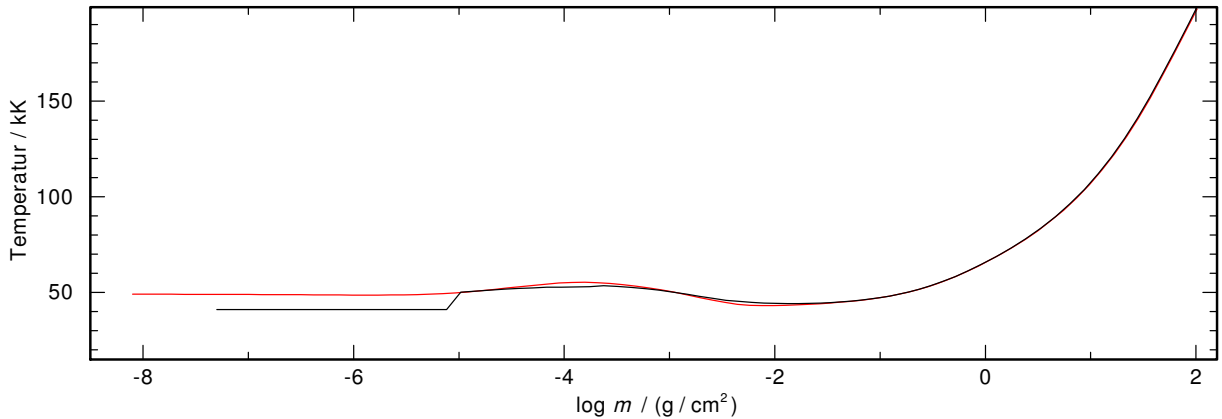


Abbildung 4.4.: Vergleich der Temperaturstrukturen der *TMAP*- (rot) und *TMAW*-HHe-Modelle (schwarz) mit  $T_{\text{eff}} = 60\,000\text{ K}$ , und  $\log g = 5.25$ .

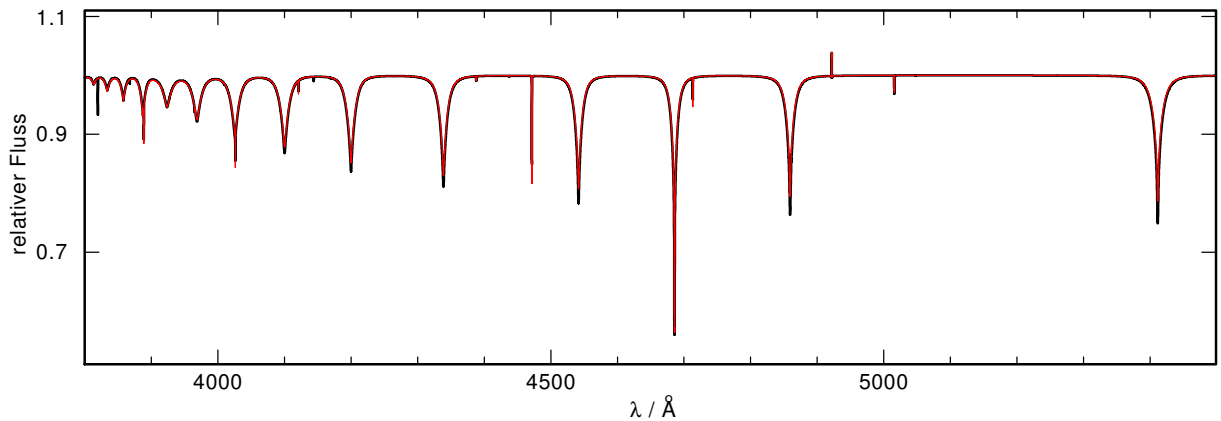


Abbildung 4.5.: Vergleich der Flüsse der *TMAP*- (rot) und *TMAW*-HHe-Modelle (schwarz) mit  $T_{\text{eff}} = 60\,000\text{ K}$  und  $\log g = 5.25$ .

am Anfang der ersten Iteration invertiert und dann konstant gehalten (Kantorovich-Verfahren). Erst nach fünf weiteren Iterationen, wird die Matrix neu invertiert. Bei manchen Parameterkombinationen führt dieses Verfahren jedoch nicht zum Erfolg. Deshalb wurde in einem weiteren Test die Berechnung der Nebenbedingungen nach dem Newton-Raphson-Verfahren durchgeführt, wobei die Inversion der Matrix jede Iteration durchgeführt wird. Das Ergebnis ist, dass alle Modellrechnungen beendet wurden (Abb. 4.3 unten rechts). Dieses Verfahren wird jedoch nicht standardmäßig in die *TMAW*-Berechnungen eingebaut, da sich die Rechenzeit ansonsten deutlich erhöht, was für den Nutzer unannehmbar ist.

Da die eben beschriebenen Testrechnungen keine deutliche bzw. nur mit erheblich höherer Rechenzeit eine Verbesserung gegenüber dem ursprünglichen Verfahren gezeigt haben, wurde die Rechenprozedur von *TMAW* nicht verändert. Nachdem im März 2012 die Modellatome überarbeitet wurden und die in Kapitel 3.5.1 beschriebenen Nachberechnungsschritte eingeführt worden sind, wurde das HHe-Gitter erneut berechnet und die resultierenden Modelle mit den *TMAP*-Modellen verglichen.

Beim Vergleich der Temperaturstrukturen wurde bereits in Kapitel 3.5.1 sichtbar, dass die Unsöld-Lucy-Korrektur eine deutlich unterschiedliche Temperaturstruktur zur Folge hat als die normale Temperaturkorrektur. Mit den eingeführten Nachberechnungsschritten gleicht sich die Temperaturstruktur der von *TMAP* weiter an (Abb. 4.4). Bevor die Lineformationsrechnung mit den größeren Modellatomen eingeführt wurde, waren die höheren Serienmitglieder der optischen von  $n = 4$  ausgehenden He II-Serie nicht im *TMAW*-Modell enthalten. Bei einem Vergleich der Spektren wird nun sichtbar, dass jetzt auch

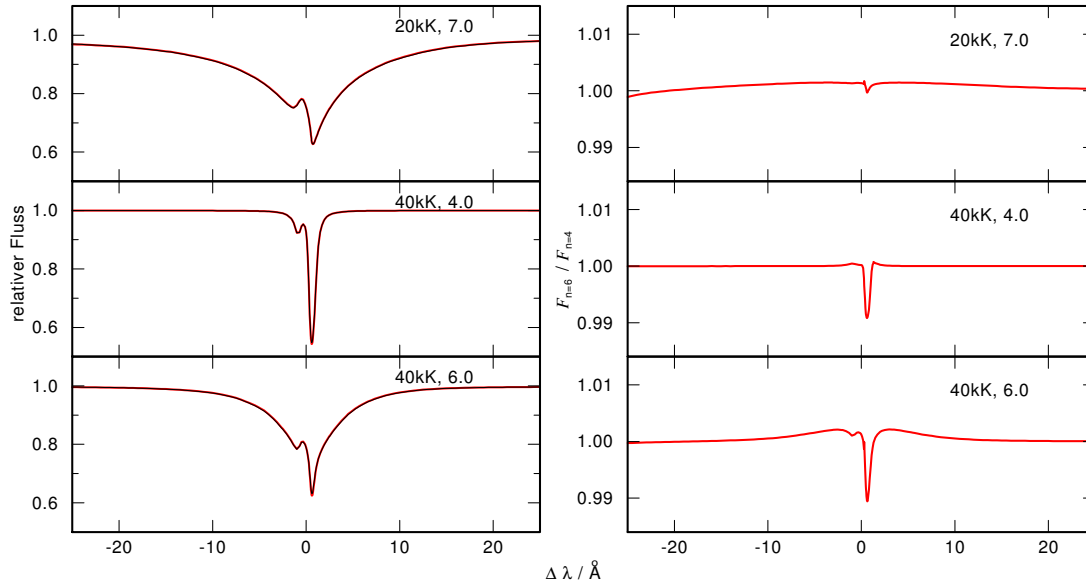


Abbildung 4.6.: Links: Vergleich von theoretischen Linienprofilen der He I  $\lambda$  4471 Å von HHe-Modellen mit Levels bis  $n = 4$  (schwarz) und  $n = 6$  (rot) für unterschiedliche Effektivtemperaturen und  $\log g$ -Werte. Rechts ist der Quotient beider Modellflüsse dargestellt.

im *TMAW*-Modell alle Linien zu sehen sind (Abb. 4.5).

Zur Temperaturbestimmung eignet sich bei diesen  $T_{\text{eff}}$  das Verhältnis der He II-Linien zur He I  $\lambda$  4471 Å. Diese He I-Linie kommt im *TMAW*- und *TMAP*-Modell allerdings unterschiedlich stark heraus. Werden die *TMAW*-Modelle für eine Spektralanalyse verwendet, kann die Temperatur anhand des Ionisationsgleichgewichts dennoch auf 20 kK genau (verglichen mit dem *TMAP*-Resultat) bestimmt werden. Um dennoch zu überprüfen, ob die He I-Linie mit *TMAW* besser modelliert werden kann, wurde ein Test mit dem neuen He I-Modellatom durchgeführt. Dabei wurde untersucht, ob eine größere Anzahl von NLTE-Levels das resultierende Linienprofile beeinflusst.

### Auswirkung einer höheren NLTE-Levelzahl auf die He I $\lambda$ 4471 Å

Um die Besetzungszahlen der Levels, zwischen denen Übergänge stattfinden, realistisch zu modellieren, müssen einige Niveaus über diesen ebenfalls als NLTE-Levels berücksichtigt werden. Für He I ist die Linie bei 4471 Å die stärkste beobachtete in dem bei dieser Analyse verwendeten optischen Spektrum. Sie entsteht zwischen den He I-Niveaus  $2p^3P$  ( $n = 2, l = 1$ ) und  $4d^3D$  ( $n = 4, l = 2$ ). In *TMAW* werden für  $T_{\text{eff}} < 100\,000$  K alle Niveaus bis einschließlich Hauptquantenzahl  $n = 4$  und Drehimpulsquantenzahl  $l = 3$  als NLTE-Levels berücksichtigt, d.h. über dem He I  $4d^3D$ -Niveau existieren nur vier weitere NLTE-Levels. Da diese Heliumlinie eine der wichtigsten Temperaturindikatoren im Temperaturbereich zwischen 50 000 und 70 000 K ist und deshalb für zuverlässige Analysen sehr wichtig ist, wurde getestet, ob mehr NLTE-Niveaus über dem He I  $4d^3D$ -Niveau berücksichtigt werden müssen.

Für diesen Test wurden HHe-Modelle mit  $T_{\text{eff}} = 40\,000$  K und  $\log g = 4.0$  und  $6.0$  berechnet, jeweils einmal mit einem Helium-Modellatom, in dem alle Niveaus bis einschließlich  $n = 4$  als NLTE-Levels berücksichtigt wurden (vier Levels über dem He I  $4d^3D$ -Niveau) und einem mit allen Niveaus bis einschließlich  $n = 6$  (16 Level über dem He I  $4d^3D$ -Niveau). Die Modelle wurde mit einem Häufigkeitsverhältnis von  $H/He = 50/50$  in Massenbruchteilen berechnet. Zusätzlich wurden Modelle mit den selben Parametern und mit solaren Häufigkeiten erzeugt. Die resultierende Linienprofile weichen maximal 1.5% voneinander ab (Abb. 4.6), was, verglichen mit den Unterschieden der *TMAP*- und *TMAW*-Modelle des vorherigen Tests, kaum etwas ausmacht.

Tabelle 4.2.: Statistik der HHeCNO-Modellatome für die *TMAP*-Rechnung.

Ion	LTE	NLTE	RBB
H I	1	15	105
H II	0	1	-
He I	74	29	69
He II	12	20	190
He III	0	1	-
C III	54	13	32
C IV	44	14	35
C V	0	1	0
N III	53	13	24
N IV	78	16	30
N V	48	14	35
N VI	0	1	0
O III	51	21	38
O IV	76	18	39
O V	109	17	35
O VI	0	1	0
Gesamt	600	195	632

Für niedrigere als die hier getesteten Temperaturen ist das He I-Ion gegenüber He II dominant. Ob sich der Einfluss der zusätzlichen NLTE-Niveaus für diesen Temperaturbereich ändert, wurde mit Hilfe eines Modells mit  $T_{\text{eff}} = 20\,000\text{ K}$  und  $\log g = 7.0$  getestet. Die Auswirkungen sind allerdings noch kleiner als im vorherigen Fall (Abb. 4.6). Für die *TMAW*-Rechnungen ist das verwendete Helium-Modellatom also ausreichend groß und muss nicht um weitere Levels erweitert werden.

### HHeCNO-Modelle

Aufgrund des deutlich sichtbaren Balmerlinienproblems (Abb. 4.7 und 4.8) wurde der Vergleich der Beobachtung mit reinen HHe-Modellen an dieser Stelle eingestellt und die in *TMAW* verfügbaren Metalle (C, N und O) in die *TMAP*-Analyse einbezogen (Tabelle 4.2 und 4.3). Ein Gitter mit insgesamt neun Modellen wurde für jedes der beiden Objekte berechnet ( $T_{\text{eff}} = 60\,000 - 70\,000\text{ K}$ ,  $\Delta T_{\text{eff}} = 5\,000\text{ K}$ ,  $\log g = 4.75 - 5.25$ ,  $\Delta \log g = 0.25$ ). Für LSE 153 stellte sich im Laufe der Analyse heraus, dass die Gitterschritte in  $T_{\text{eff}}$  zu groß sind, weshalb ein weiteres Modell mit  $T_{\text{eff}} = 62\,500\text{ K}$  berechnet wurde. Für LSE 263 wurde das Gitter um  $\log g = 5.50$  erweitert.

Die beobachteten optischen Linien zeigen bei beiden Objekten keine Anzeichen einer Rotation des Sterns. Im Gegensatz zu Husfeld et al. (1989) wurden die Modelle deshalb nicht mit einer Rotationsgeschwindigkeitskorrektur versehen. Für die Analyse wurde eine Radialgeschwindigkeitskorrektur mit

$$\lambda_{\text{gemessen}} = \lambda_{\text{Ruhe}} \left( 1 + \frac{v_{\text{rad}}}{c} \right),$$

durchgeführt, wobei für beide Objekte  $v_{\text{rad}} = -20\text{ km/sec}$  verwendet wurde.

Die Ergebnisse der *TMAP*-Analyse sind in Tabelle 4.4 dargestellt. Die bereits bei den HHe-Modellen bestimmte, im Vergleich zum Literaturwert niedrigere  $T_{\text{eff}}$ , kann mit diesen Modellen bestätigt bzw. genauer bestimmt werden.

Für LSE 153 reproduziert das Modell mit  $T_{\text{eff}} = 60\,000\text{ K}$  und  $\log g = 5.0$  die Linienprofile der optischen Heliumlinien sowie das Ionisationsgleichgewicht der optischen He II-Linien und He I  $\lambda\,4471\text{Å}$  am

Tabelle 4.3.: Statistik der HHeCNO-Modellatome für die Lineinformationrechnung.

Ion	LTE	NLTE	RBB
H I	1	15	105
H II	0	1	-
He I	74	29	69
He II	12	20	190
He III	0	1	-
C III	11	56	273
C IV	4	54	295
C V	0	1	0
N III	32	34	129
N IV	38	56	241
N V	8	54	297
N VI	0	1	0
O III	0	72	322
O IV	56	38	173
O V	50	76	472
O VI	0	1	0
Gesamt	579	556	2647

Tabelle 4.4.: Vergleich der in dieser Arbeit bestimmten Parameter mit den Literaturwerten (Husfeld et al., 1989) von LSE 153 und LSE 263. Die Häufigkeiten sind in Massenbruchteilen angegeben.

	LSE 153		LSE 263	
	Literaturwert	diese Arbeit	Literaturwert	diese Arbeit
$T_{\text{eff}}$ (kK)	70	60	70	60
$\log g$ / (cm/sec <sup>2</sup> )	4.75	5.00	4.90	5.25
H	<0.02	$4.98 \times 10^{-3}$	<0.02	$5.05 \times 10^{-3}$
He		$9.71 \times 10^{-1}$		$9.86 \times 10^{-1}$
C	0.05	$1.80 \times 10^{-2}$	$2.5 \times 10^{-4}$	< $2.37 \times 10^{-4}$
N	$7.9 \times 10^{-3}$	$5.03 \times 10^{-4}$	$1.5 \times 10^{-2}$	$1.42 \times 10^{-3}$
O		< $5.08 \times 10^{-3}$		< $7.08 \times 10^{-3}$

besten (Abb. 4.7). Das C III/C IV-Ionisationsgleichgewicht wird mit diesem Modell ebenfalls gut reproduziert.

Die optische Beobachtung von LSE 263 zeigt keine Kohlenstofflinien und hat ein niedrigeres Signal-zu-Rausch-Verhältnis als die Beobachtung von LSE 153. Als Temperatur- und  $\log g$ -Indikator wurde deshalb nur das He I/He II-Ionisationsgleichgewicht verwendet (Abb. 4.8). Die Fehlergrenzen dieser Analyse sind aufgrund dieser beiden Einschränkungen größer als für LSE 153. Das Modell mit  $T_{\text{eff}} = 60\,000$  K und  $\log g = 5.25$  reproduziert die Beobachtung am besten. He I  $\lambda 4471$  Å ist im Modell leicht zu stark, für  $T_{\text{eff}} = 65\,000$  K allerdings deutlich zu schwach. Für die Kohlenstoffhäufigkeit wurden Obergrenzen bestimmt.

Da das beobachtete optische Spektrum von LSE 153 und LSE 263 keine Sauerstofflinien zeigt, wurde

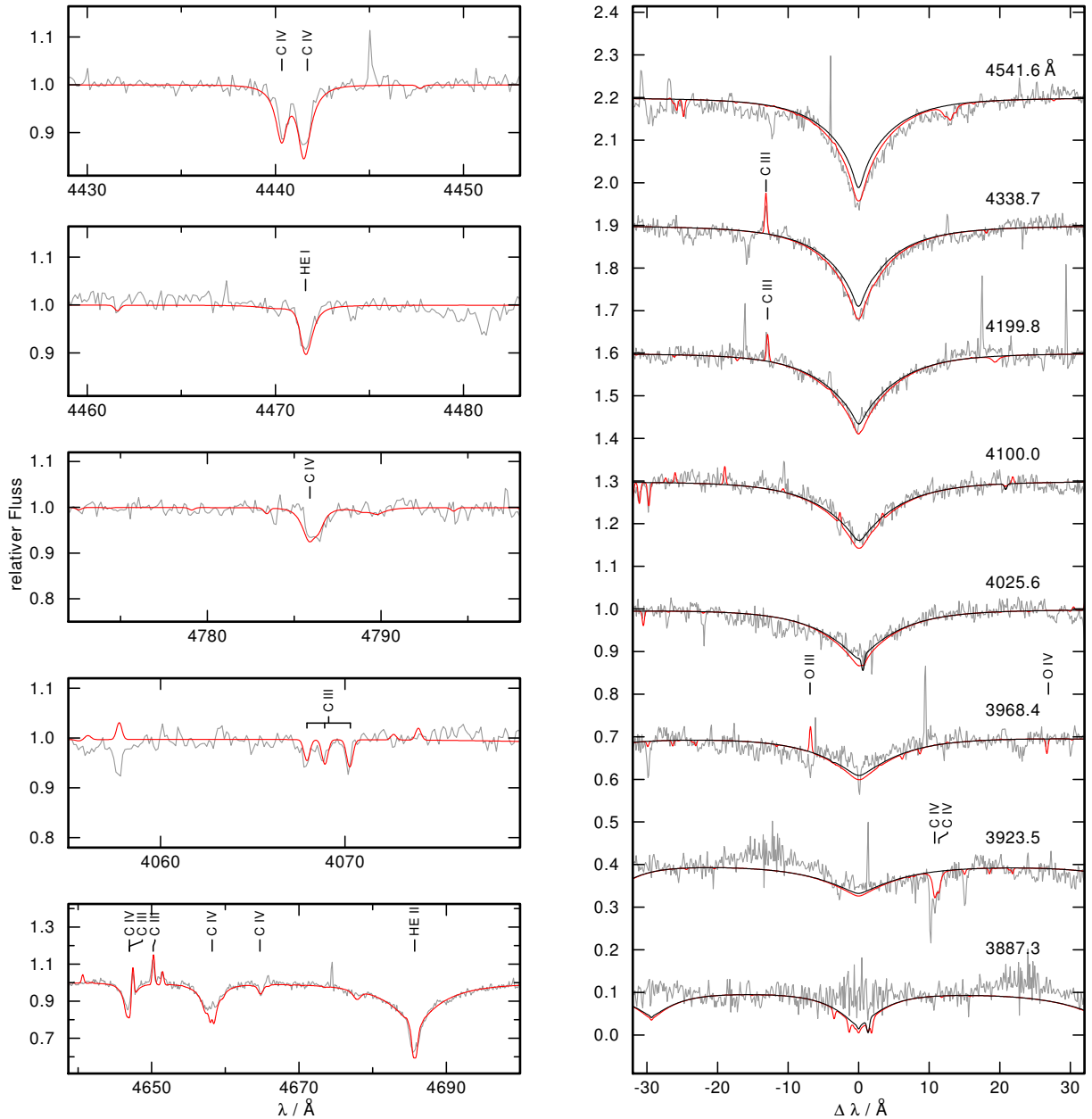


Abbildung 4.7.: TMAP-Modelle verglichen mit dem CASPEC-Spektrum von LSE 153. Die Modelle beinhalten die Elemente H, He, C, N und O (rot, Häufigkeiten wie in Tabelle 4.4) bzw. H und He (schwarz, 99.75% He, 0.25% H in Massenbruchteilen) mit  $T_{\text{eff}} = 60\,000\text{ K}$  und  $\log g = 5.0$ . Die Modelle wurden mit einem Gaussprofil der Halbwertsbreite  $0.3\text{ \AA}$  gefaltet, um die Instrumentenauflösung zu simulieren. In allen folgenden Abbildungen für Vergleiche von Modell und Beobachtung werden die Modelle mit einem Gaussprofil der Halbwertsbreite der Instrumentenauflösung gefaltet, wenn nicht anders angegeben.

für dieses Element ebenfalls nur eine Obergrenze bestimmt. Wasserstoff ist, aufgrund der Überlagerung der Wasserstoff- durch Heliumlinien, ebenfalls nicht detektierbar und wurde hier mit einer Häufigkeit von 0.005 (Massenbruchteil) angenommen.

Generell ergeben beide Analysen eine im Vergleich zum Literaturwert niedrigere  $T_{\text{eff}}$  und höheres  $\log g$ . Der Trend der niedrigeren Temperatur ist, wie bereits erwähnt, in Übereinstimmung mit den Ergebnissen aus Ionisationsgleichgewichtsbetrachtungen von Eisen- und Nickellinien im IUE-Bereich (Haas,

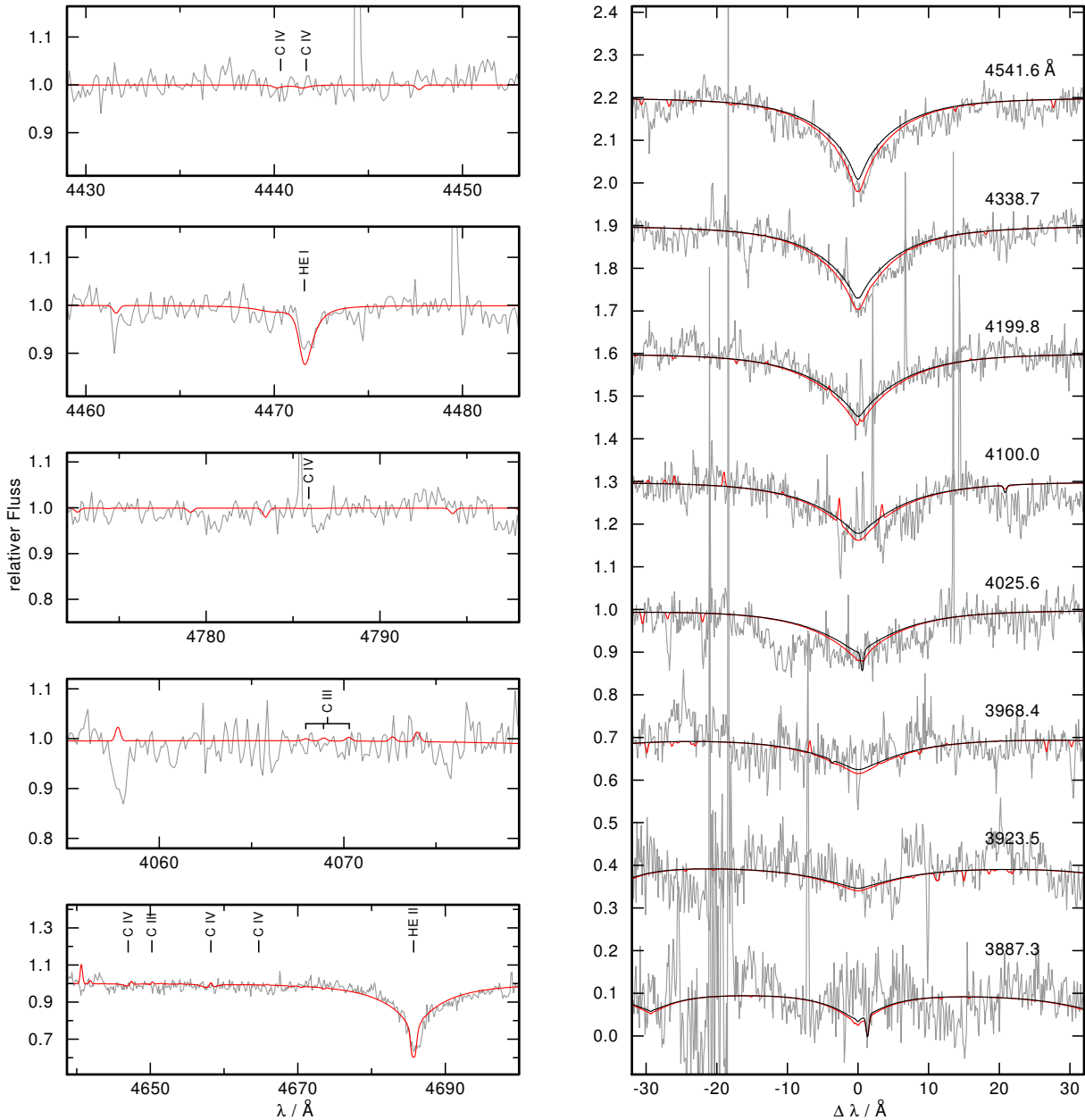


Abbildung 4.8.: Wie Abb. 4.7, für LSE 263 mit  $T_{\text{eff}} = 60\,000\text{ K}$  und  $\log g = 5.25$ . Die Häufigkeiten sind ebenfalls in Tabelle 4.4 angegeben bzw. betragen 99.75% He und 0.25% H in Massenbruchteilen.

1997). Im Vergleich zu den Literaturwerten wurden die Kohlenstoffhäufigkeiten in dieser Analyse leicht gesenkt, sind aber immer noch in Übereinstimmung mit den Ergebnissen von Husfeld et al. (1989). Die Sauerstoffhäufigkeiten der beiden Objekte wurden vor dieser Arbeit noch nicht bestimmt, die Obergrenzen sind etwas höher als die von Haas (1997) angenommenen 0.1-fach solaren Werte.

Für Stickstoff werden die beiden optischen N V-Linien bei  $4604\text{ Å}$  und  $4620\text{ Å}$  in beiden Objekten mit den neuen Häufigkeiten nicht ganz ausgefüllt. Diese sind etwas niedriger als die Literaturhäufigkeiten. Eine weitere Erhöhung der Häufigkeiten führt jedoch zu Linien im Modell, die mit der Beobachtung nicht mehr verträglich wären.

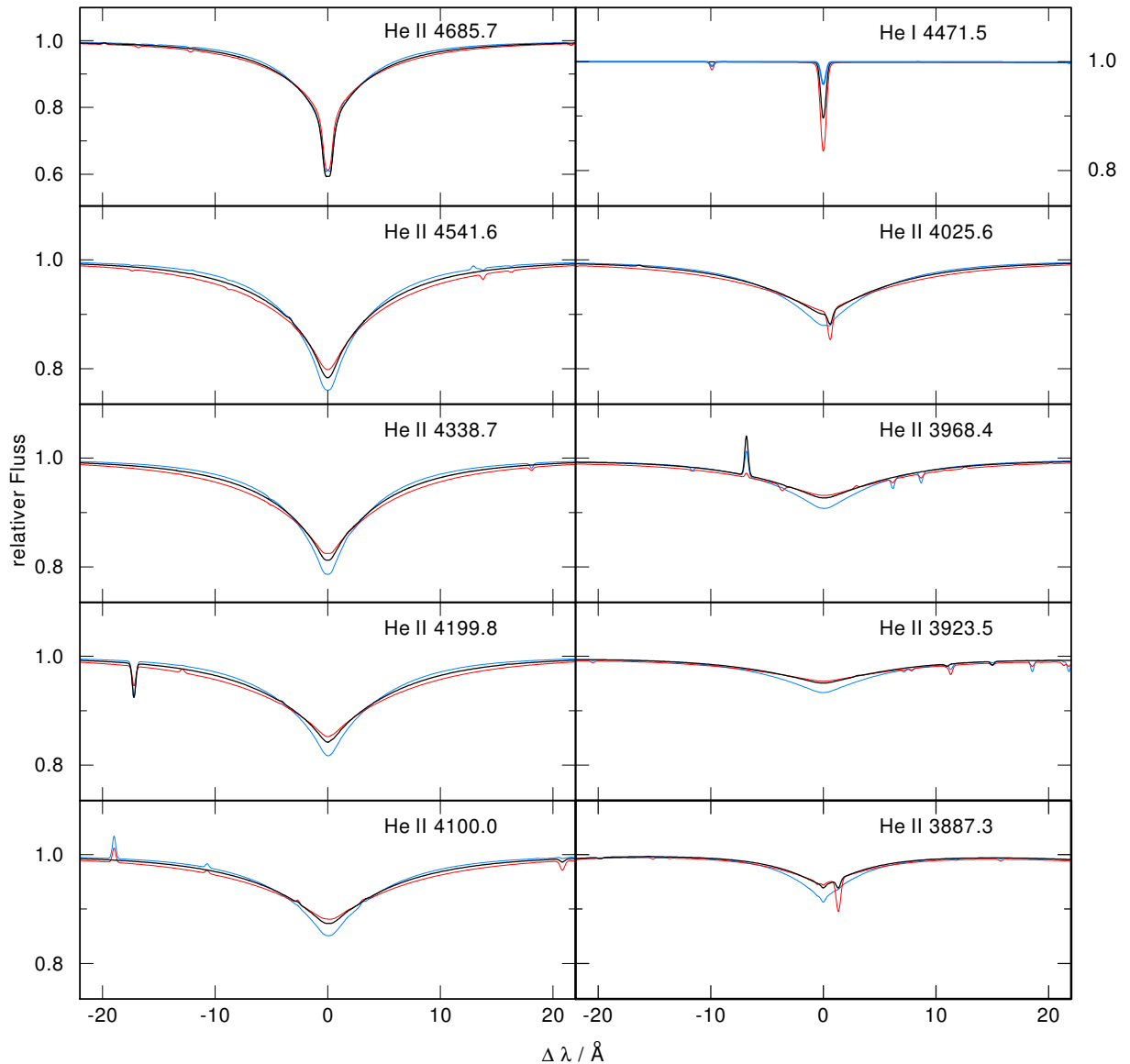


Abbildung 4.9.: HHeCNO-*TMAP*-Modelle ( $T_{\text{eff}} = 70\,000\text{ K}$ ,  $\log g = 5.0$ , blau und  $T_{\text{eff}} = 60\,000\text{ K}$ ,  $\log g = 5.5$ , rot) verglichen mit dem *TMAW*-Modell ( $T_{\text{eff}} = 65\,000\text{ K}$ ,  $\log g = 5.25$ , schwarz) mit den gleichen Elementhäufigkeiten (siehe Text) für LSE 263. Um eine Abschätzung im Bezug auf diese Analyse machen zu können, wurden die Modelle ebenfalls mit einem Gaussprofil der Halbwertsbreite  $0.3\text{ \AA}$  gefaltet.

### Vergleich der HHeCNO-Modelle mit den *TMAW*-Spektren

Im nächsten Schritt wird nun die Genauigkeit von *TMAW*-Modellen in diesem Parameterbereich untersucht. Dazu wurde jeweils das zu diesem Zeitpunkt als final erachtete Modell mit *TMAW* berechnet und anschließend mit den *TMAP*-Modellen verglichen, die die Fehlergrenzen der Analyse repräsentieren. Gegen Ende dieser Arbeit wurden die finalen Parameter beider Analysen nochmals leicht verändert, da zuvor die Standardverbreiterung anstelle der üblichen Linienverbreiterungstabellen für die Modellierung der He I-Linien verwendet wurden. Der Vergleich mit den *TMAW*-Modellen und mit den Modellen, die Eisengruppenelemente berücksichtigen (siehe unten) wurde vor dieser Änderung durchgeführt. Die verwendeten Modelle haben deshalb alle eine etwas höhere  $T_{\text{eff}}$  (zentrales Modell bei  $62\,500\text{ K}$  für LSE 153 bzw. bei  $65\,000\text{ K}$  für LSE 263) und niedrigere Stickstoffhäufigkeit ( $2.96 \times 10^{-5}$  für LSE 153

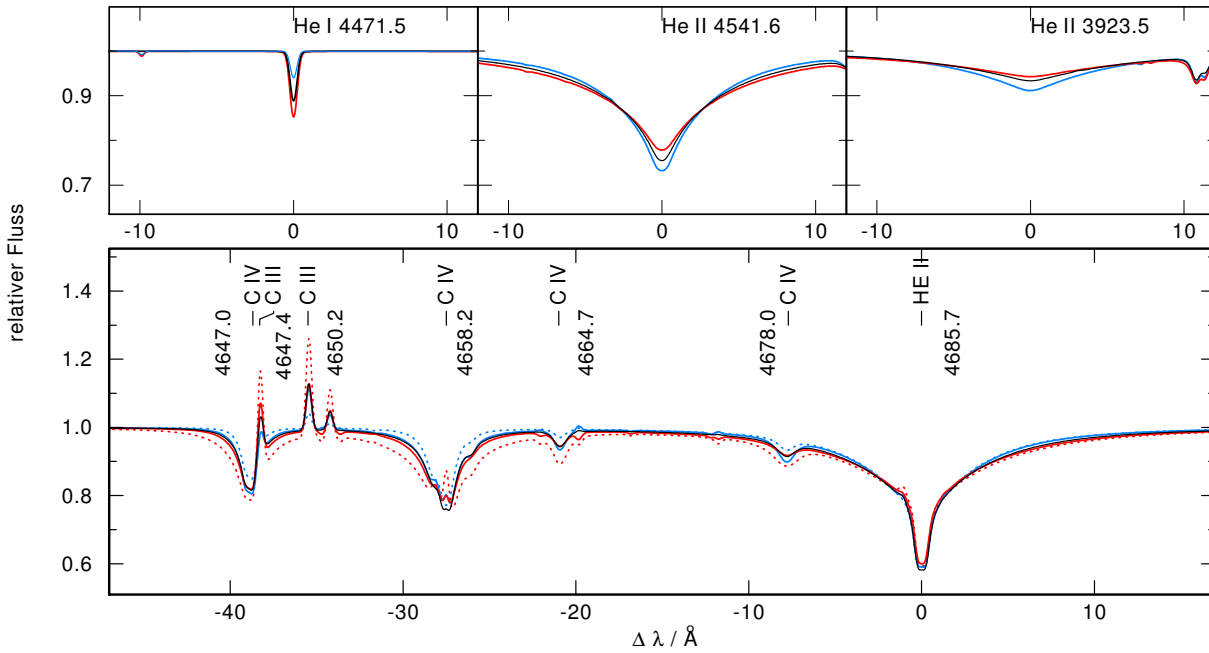


Abbildung 4.10.: HHeCNO-*TMAP*-Modelle ( $T_{\text{eff}} = 65\,000\text{ K}$ ,  $\log g = 4.75$  blau und  $T_{\text{eff}} = 60\,000\text{ K}$ ,  $\log g = 5.25$  rot) verglichen mit dem *TMAW*-Modell ( $T_{\text{eff}} = 62\,500\text{ K}$ ,  $\log g = 5.00$  schwarz) mit den gleichen Elementhäufigkeiten (siehe Text) für LSE 153, jeweils mit einem Gaussprofil mit Halbwertsbreite  $0.3\text{ \AA}$  versehen. Gepunktet dargestellt sind die Fehlergrenzen der Kohlenstoffhäufigkeit.

und  $9.50 \times 10^{-4}$  für LSE 263) als die finalen Parameter aus Tabelle 4.4. Die leicht veränderten Parameter sollten allerdings keine Auswirkungen auf das Ergebnis der *TMAW*- und Eisengruppenelementvergleiche haben.

Während der Erzeugung der *TMAW*-Modelle stellte sich heraus, dass die Berechnung der Modelle numerisch deutlich stabiler ist, wenn die O *v*-Resonanzlinie beim Modellieren der Atmosphärenstruktur um einen Faktor 20 reduziert wird. Im Lineformationschritt wird wieder der korrekte Wert verwendet. Da diese Linie im EUV-Bereich liegt und die Unsöld-Lucy-Temperaturkorrektur verwendet wird, sind die erwarteten Auswirkungen auf die Atmosphäre vernachlässigbar. Dieses Verfahren wird nun standardmäßig von *TMAW* für diesen Temperaturbereich ( $40\,000 - 70\,000\text{ K}$ ) verwendet, wodurch nun Modelle erzeugt werden können, die zuvor die Berechnung abgebrochen hatten.

Die Fehlergrenzen der Analyse von LSE 263 sind größer als die von LSE 153. Ein Vergleich des *TMAW*-Modells mit den *TMAP*-Modellen von LSE 263, die die Fehlergrenzen repräsentieren ( $T_{\text{eff}} = 70\,000\text{ K}$ ,  $\log g = 5.0$  und  $T_{\text{eff}} = 60\,000\text{ K}$ ,  $\log g = 5.5$ ) zeigt, dass das *TMAW*-Modell innerhalb der durch diese Modelle gegebenen Grenzen liegt (Abb. 4.9). Insbesondere bei der zur Temperaturbestimmung wichtigen He I  $\lambda 4471\text{ \AA}$  liegt das *TMAW*-Modell beinahe mittig zwischen den beiden Grenzen. Für steigende Serienmitglieder nähert sich das *TMAW*-Modell immer mehr dem *TMAP*-Modell mit tieferer Temperatur und höherem  $\log g$  an. Eine Analyse mit dem *TMAW*-Modell führt also zu einem Ergebnis, das um weniger als diese Fehlergrenzen (ca. 8% in  $T_{\text{eff}}$ , ca. 5% in  $\log g$ ) abweicht.

Für LSE 153 führte die Analyse zu kleineren Fehlergrenzen. Wird das *TMAW*-Modell mit den *TMAP*-Modellen für die Fehlergrenzen ( $T_{\text{eff}} = 65\,000\text{ K}$ ,  $\log g = 4.75$  und  $T_{\text{eff}} = 60\,000\text{ K}$ ,  $\log g = 5.25$ ) verglichen, ist zu erkennen, dass das *TMAW*-Modell für die Heliumlinien dazwischen liegt. Das Modell nähert sich, wie bei LSE 263, für steigende Serienmitglieder einem der Grenzmodelle an. Bei einer Analyse der für die Temperaturbestimmung relevanten Linien der Ionisationsgleichgewichte führt das *TMAW*-Modell zu einer Temperatur innerhalb dieser Fehlergrenzen (4%), bei einer  $\log g$ -Bestimmung



anhand der optischen Heliumlinien ebenfalls (ca. 5%). Werden die Fehlergrenzen der Kohlenstoffhäufigkeit (0.5 dex) berücksichtigt, liegt das *TMAW*-Modell auch für diese Linien zwischen den Grenzmodellen.

Insgesamt konnte mit dieser Analyse gezeigt werden, dass *TMAW* für diesen Parameterbereich ( $T_{\text{eff}} \approx 60\,000\text{ K}$ ,  $\log g \approx 5$  und nahezu reine Heliumatmosphären) Modelle liefert, die eine Analyse optischer Spektren mit einer Genauigkeit zwischen 4% und 8% in  $T_{\text{eff}}$  und mit ca. 5% in  $\log g$  ermöglichen. Durch die Einführung der in Kapitel 3.4 beschriebenen Nachberechnungsschritte sind die Modellatome groß genug, um auch optische Linien einzuschließen und die Atmosphärenstruktur realistisch zu modellieren. *TMAW* ist also für zuverlässige Spektralanalysen in diesem Parameterbereich im optischen Wellenlängenbereich geeignet.

## Der UV-Bereich und die Eisengruppenelemente

Bei der Spektralanalyse eines Sterns ist es wichtig, den gesamten zugänglichen Wellenlängenbereich zu analysieren. Ziel dieses Tests mit den beiden He-sdO-Sternen war allerdings keine vollständige Analyse, sondern eine Abschätzung der *TMAW*-Genauigkeit im optischen Spektralbereich im Vergleich zu *TMAP*-Modellen. Um den UV-Bereich dennoch nicht auszuschließen, wurden die finalen Modelle mit den Beobachtungen im UV-Bereich verglichen.

Für LSE 153 und LSE 263 sind sowohl je ein FUSE<sup>5</sup>-Spektrum als auch ein hochaufgelöstes IUE-Spektrum vorhanden (ID: Z9015501000 und SWP18055 für die FUSE und IUE Spektren von LSE 153, ID: D0660401000 und SWP18056 für LSE 263). Eine Analyse dieser Spektren ist nicht Teil dieser Arbeit, ein kurzer Konsistenzcheck der aus der optischen Analyse resultierenden Parameter wurde allerdings durchgeführt. In den UV-Beobachtungen von beiden Objekten sind Linien von C, N und O zu sehen, was eine weitere Abschätzung der im optischen Bereich bestimmten Obergrenzen möglich macht. Damit sind auch weitere Ionisationsgleichgewichte vorhanden, die eine genauere Beurteilung von  $T_{\text{eff}}$  zulassen.

Für beide Objekte reproduzieren die Modelle größtenteils die UV-Beobachtungen (Abb. 4.11 und 4.12). Die Ionisationsgleichgewichte von C III/C IV, N IV/N V und O III/O IV/O V werden überwiegend getroffen. Für LSE 263 sind die modellierten Kohlenstoff- und Sauerstofflinien zu stark. Die Häufigkeiten dieser Elemente sind also etwas zu hoch. Da hierfür im optischen Bereich nur Obergrenzen bestimmt wurden, ist dieses Ergebnis einer niedrigeren Häufigkeit auch für den optischen Bereich konsistent. Die beobachteten Linien sind insgesamt von Linien überlagert, die nicht im Modell zu sehen sind, also entweder von Linien anderer Elemente in der Atmosphäre oder von interstellaren Linien. Deshalb wurden die interstellaren Linien für beide Objekte mit dem Programm *OWENS* modelliert. Mit diesem Programm können verschiedene interstellare Komponenten mit unterschiedlichen Temperaturen, Säulendichten, Radial- und Turbulenzgeschwindigkeiten modelliert werden. Bei beiden Objekten können so viele Linien als interstellare Linien identifiziert werden.

Bei LSE 153 ist die Überlagerung durch interstellare oder nicht identifizierte Linien nicht so stark ausgeprägt wie bei LSE 263. Die modellierten Kohlenstofflinien füllen die beobachteten (Abb. 4.12) durch diese Überlagerung nicht immer perfekt aus. Die N IV-Linien sind mit der Beobachtung konsistent, die N V  $\lambda 1243\text{ \AA}$  ist im Modell zu stark. Die O IV-Linien werden gut reproduziert, die O V  $\lambda 1371\text{ \AA}$  und die O III  $\lambda 1154\text{ \AA}$  sind ebenfalls leicht zu stark. Insgesamt sind die Resultate der optischen Analyse mit den UV-Beobachtungen konsistent, die Abweichungen von Modell und Beobachtung können durch eine Überlagerung mit Linien, die nicht modelliert wurden oder durch eine nicht perfekt normierte Beobachtung (z.B. Linienflügel der N V-Linie in Abb. 4.12) hervorgerufen werden.

Ein Vergleich der *TMAW*-Modelle mit den *TMAP*-Modellen für die Fehlergrenzen zeigt auch in diesem Wellenlängenbereich, dass die *TMAW*-Modelle generell zwischen den *TMAP*-Modellen liegen (Abb. 4.13). Für Stickstoff und Sauerstoff weichen die Modelle etwas stärker ab. Für eine Bestimmung

---

<sup>5</sup><http://fuse.pha.jhu.edu/>

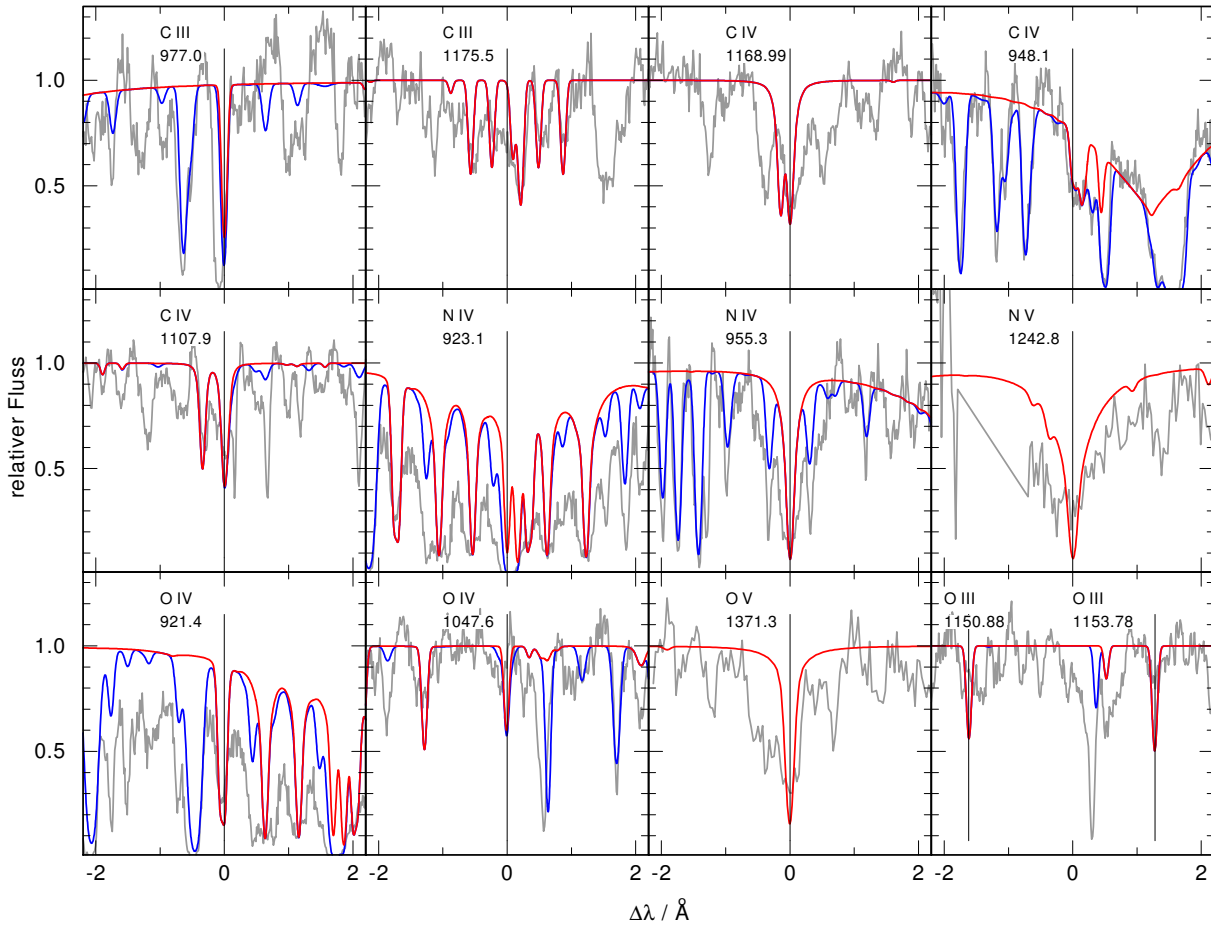


Abbildung 4.11.: Das *TMAP*-Modell von LSE 263 ( $T_{\text{eff}} = 60\,000\text{ K}$ ,  $\log g = 5.25$ , nur HHeCNO) verglichen mit Ausschnitten aus dem FUSE- und IUE-Bereich. Zusätzlich sind interstellare Linien modelliert (blau).

der Temperatur mit *TMAW* im UV-Bereich anhand des Ionisationsgleichgewichts der Kohlenstofflinien sind die Fehlergrenzen von 4% und 8% in  $T_{\text{eff}}$  erreicht. Für eine Bestimmung von  $\log g$  sind – unabhängig von *TMAW* – bei jeder Spektralanalyse heißer, kompakter Objekte die optischen Heliumlinien zuverlässiger als UV-Linien. Wird der Fehler der Häufigkeiten von N und O berücksichtigt (0.5 dex), liegen die Linien der *TMAW*-Modelle auch für diese Elemente innerhalb der Fehlergrenzen.

Als Ursache für die bereits angesprochene beobachtete Überlagerung der UV-Linien werden oft Linien der Eisengruppenelemente vermutet. Im Rahmen dieser Arbeit sollen deshalb Eisengruppenelemente in das finale *TMAP*-Modell eingefügt werden, um deren Auswirkungen auf die Atmosphäre in diesem Parameterbereich zu untersuchen. Die Ergebnisse dieses Test geben wichtige Hinweise auf die Genauigkeit einer Spektralanalyse, bei der diese Elemente nicht korrekert mitgerechnet werden. In dem Temperaturbereich der untersuchten Objekte sind im optischen Wellenlängenbereich (aufgrund der hohen Ionisationen) keine Eisengruppenlinien zu finden. Ihr Einfluss auf die Atmosphärenstruktur kann aber auch eine Änderung der optischen Linien zu Folge haben.

Für diesen Test wurden die vor den Änderungen am Modellatmosphärencode als final erachteten Modelle um die Eisengruppenelemente erweitert. Das zugehörige Modellatom enthält Eisen und Nickel einzeln, die anderen Eisengruppenelemente werden als generisches Modellatom berücksichtigt. Die Häufigkeiten dieser neuen Elemente wurden solar angenommen. Um zu überprüfen, ob eine solare Häufigkeit realistisch ist, wurden jeweils die drei stärksten Eisen- und Nickellinien der dominanten Ionisationsstufen v und vi herausgesucht. Der Vergleich mit dem Modell zeigt (Abb. 4.15), dass für von LSE 153 eine

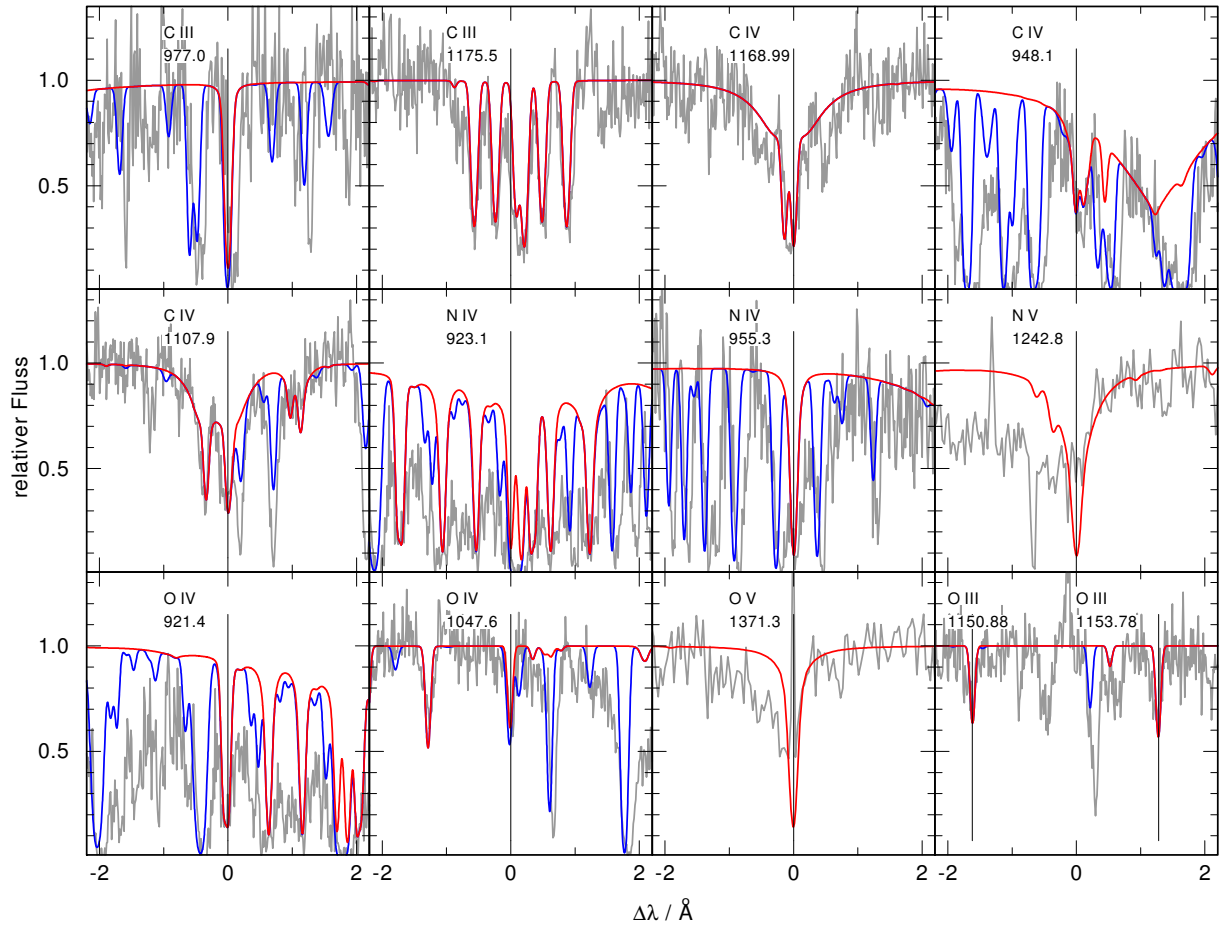


Abbildung 4.12.: Wie Abb. 4.11, für LSE 153 ( $T_{\text{eff}} = 60\,000\text{ K}$ ,  $\log g = 5.0$ ).

solare Eisen- und Nickelhäufigkeit ausreicht bzw. als Obergrenze angesehen werden kann. Die Linien liegen innerhalb des Rauschens oder füllen die beobachtete Linie (zum Teil) aus. Jeweils eine Eisen- und Nickellinie ist im Modell stärker als die Beobachtung, weshalb ein höherer Häufigkeitswert nicht in Frage kommt.

Für LSE 263 sind die Ergebnisse für die Eisenhäufigkeit ähnlich. Bei einer solaren Häufigkeit ist die Fe VI  $\lambda 1167.695\text{ \AA}$  im Modell stark genug, die anderen zu schwach (Abb. 4.16). Die Nickelhäufigkeit ist allerdings etwas höher als im Modell, da die Linien die Beobachtung nicht ausfüllen.

Testweise wurden für beide Objekte die Eisen- und Nickelhäufigkeiten im synthetischen Spektrum um einen Faktor zehn (Teilchenzahlbruchteile) erhöht. Für LSE 153 sind die resultierenden theoretischen Linien deutlich stärker als die beobachteten. Solare Häufigkeiten sind für dieses Objekt also eine Obergrenze. Bei LSE 263 sind die solaren Häufigkeiten noch nicht stark genug, um die beobachteten Linien komplett auszufüllen. Zehnfach-solare Häufigkeiten sind allerdings zu stark, weshalb die Häufigkeiten zwischen der solaren und zehnfach-solaren liegen muss. Der Trend dieser Ergebnisse stimmt mit dem von Haas (1997) überein. Die Ionisationsgleichgewichte von Eisen und Nickel bestätigen die zuvor bestimmte Temperatur und die Häufigkeiten sind für LSE 263 höher als für LSE 153. Die von Haas (1997) bestimmte niedrigere Eisenhäufigkeit von LSE 153 und höhere Nickelhäufigkeit von LSE 263 können durch die Abweichungen der Häufigkeiten der Elemente C, N und O verursacht worden sein (Haas, 1997, hat Werte von Husfeld et al., 1989, übernommen) und durch die Wahl des Modellatoms (Haas, 1997, generisches Modellatom). Ein weiterer möglicher Grund für diese Abweichungen ist, dass seither die Atomdaten der Eisengruppenelemente von Kurucz erweitert wurden.

Die resultierenden Line-Blanketing-Effekte zeigen sich in der Temperaturstruktur und im Flussver-

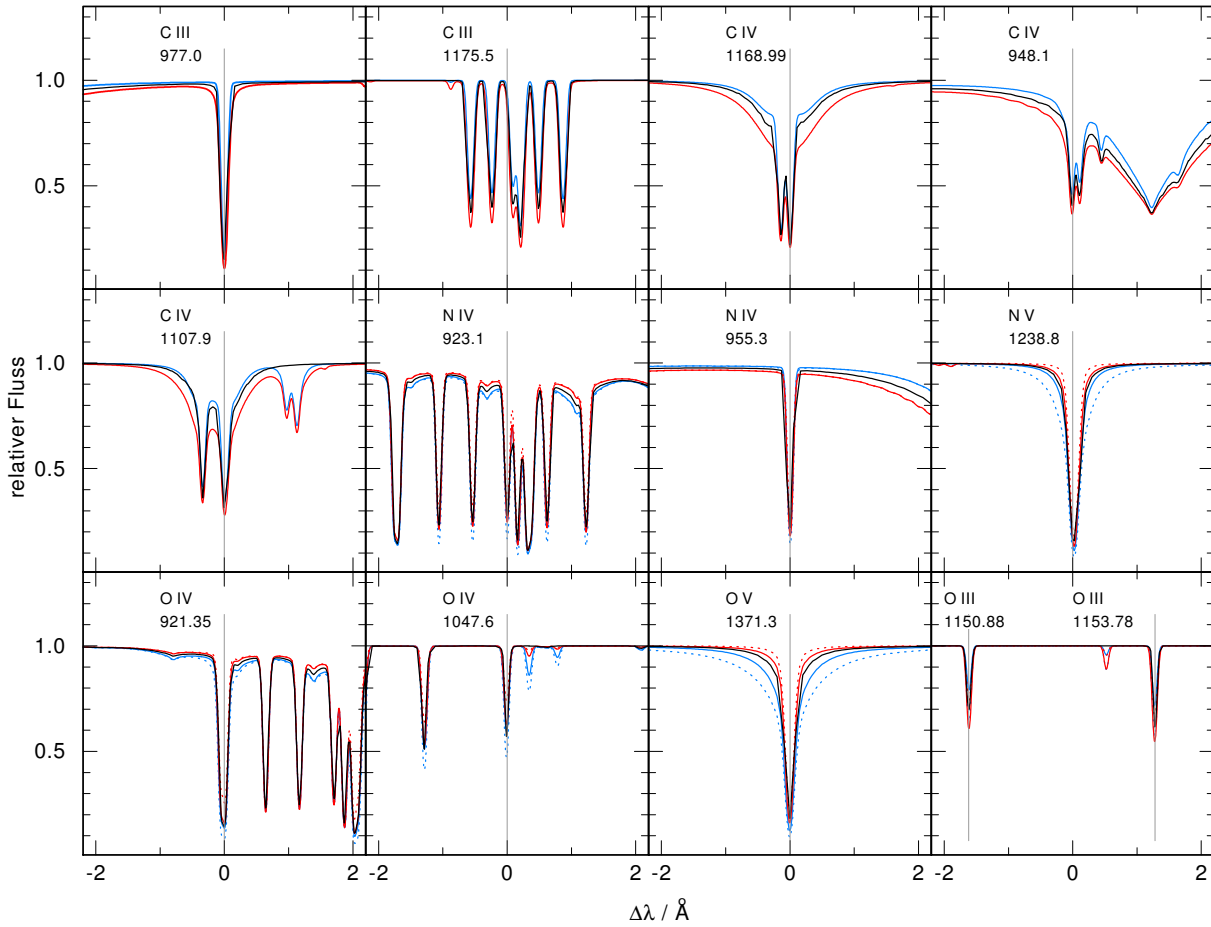


Abbildung 4.13.: Die *TMAP*- ( $T_{\text{eff}} = 65\,000\text{ K}$ ,  $\log g = 4.75$ , blau und  $T_{\text{eff}} = 60\,000\text{ K}$ ,  $\log g = 5.25$ , rot) und *TMAW*-Modelle ( $T_{\text{eff}} = 62\,500\text{ K}$ ,  $\log g = 5.00$ , schwarz) mit gleichen Elementhäufigkeiten (siehe Text) von LSE 153 im UV-Bereich. Für diesen Vergleich wurden Gaussprofile mit Halbwertsbreiten  $0.06\text{ \AA}$  bzw.  $0.1\text{ \AA}$  angebracht. Gepunktet sind die Fehlergrenzen der N- und O-Häufigkeit dargestellt.

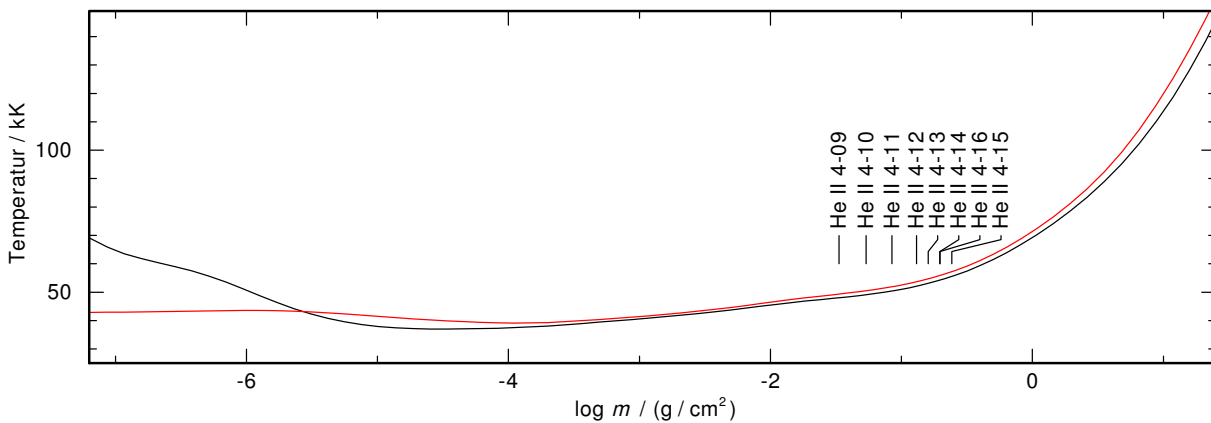


Abbildung 4.14.: Die Temperaturstruktur des HHeCNO-Modells (schwarz) von LSE 263 ( $T_{\text{eff}} = 65\,000\text{ K}$ ,  $\log g = 5.25$ ) und des Modells, das Eisengruppenelemente berücksichtigt (rot, gleiche Elementhäufigkeiten). Eingezeichnet sind außerdem die Entstehungstiefen der Kerne von Heliumlinien.

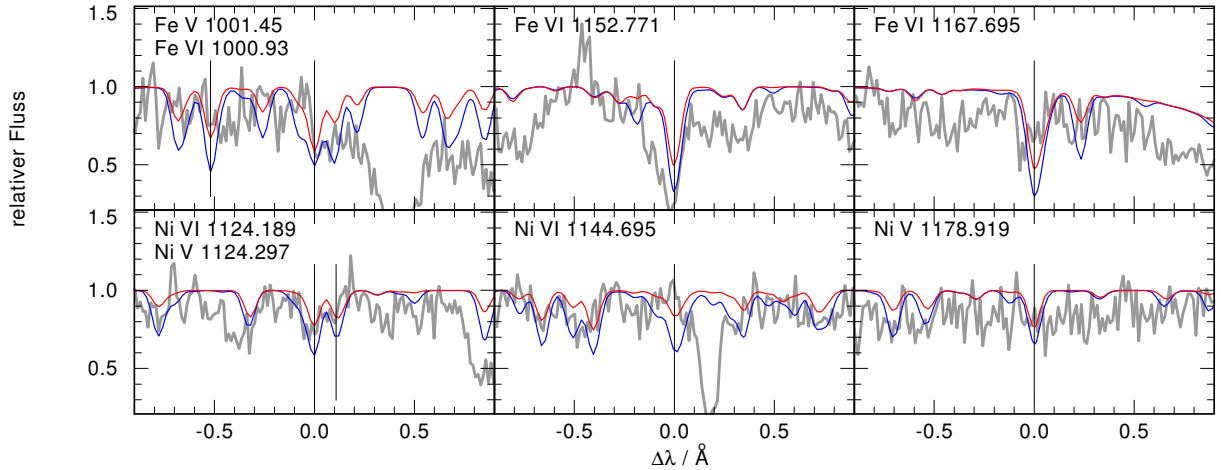


Abbildung 4.15.: Die stärksten Eisen- und Nickellinien von LSE 153 im Modell mit  $T_{\text{eff}} = 62\,500\text{ K}$ ,  $\log g = 5.00$  (Elementhäufigkeiten siehe Text). Dargestellt sind die POS-Linien mit solarer (rot) und zehnfach solarer Häufigkeit (blau) des betrachteten Elements.

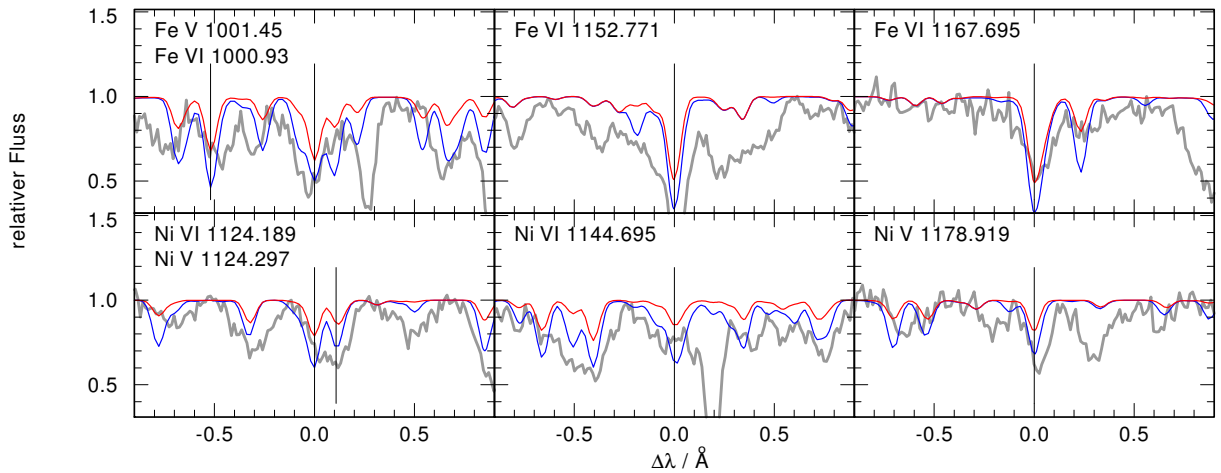


Abbildung 4.16.: Wie Abb. 4.15, für LSE 263 ( $T_{\text{eff}} = 65\,000\text{ K}$ ,  $\log g = 5.25$ ).

lauf. Im Linienentstehungsgebiet ist die Temperatur der Atmosphäre mit Eisengruppenelementen wenige Tausend Kelvin heißer (Abb. 4.14). Aufgrund der zusätzlichen Linien der Eisengruppenelemente und der Erhaltung des Gesamtflusses ist der Kontinuumsfluss der Eisengruppenmodelle kurzweilig der He II-Grundzustandskante tiefer, langweilig davon höher als bei den HHeCNO-Modellen (Abb. 4.17).

Diese Veränderungen der Atmosphäre haben eine leichte Verschiebung der Ionisationsbruchteile zur Folge, die dominante Ionisationsstufe ändert sich allerdings nicht (Abb. 4.18). Die veränderte Temperaturstruktur im Linienentstehungsgebiet resultiert in optischen Helium-Linienprofilen, die stärker als die TMAW-Modelle abweichen (Abb. 4.19 und 4.20). Für beide Objekte liegen die Flügel der Modelle mit Eisengruppenelementen nicht mehr für alle optischen Heliumlinien genau innerhalb der Fehlergrenzen, eine Analyse führt dennoch zu Werten innerhalb der Fehlergrenzen (8% in  $T_{\text{eff}}$  und 5% in  $\log g$ ). Dieser Test macht deutlich, dass die Eisengruppenelemente bei der Spektralanalyse von He-sdOs auch dann nicht vernachlässigt werden dürfen, wenn das Spektrum keine Anzeichen der Eisengruppenelemente zeigt, wie in diesem Fall das optische Spektrum. In diesem Kontext führt die Analyse mit TMAW-Modellen zu Ergebnissen, die weniger abweichen als mit Modellen, bei denen die Eisengruppenelemente berücksichtigt werden.

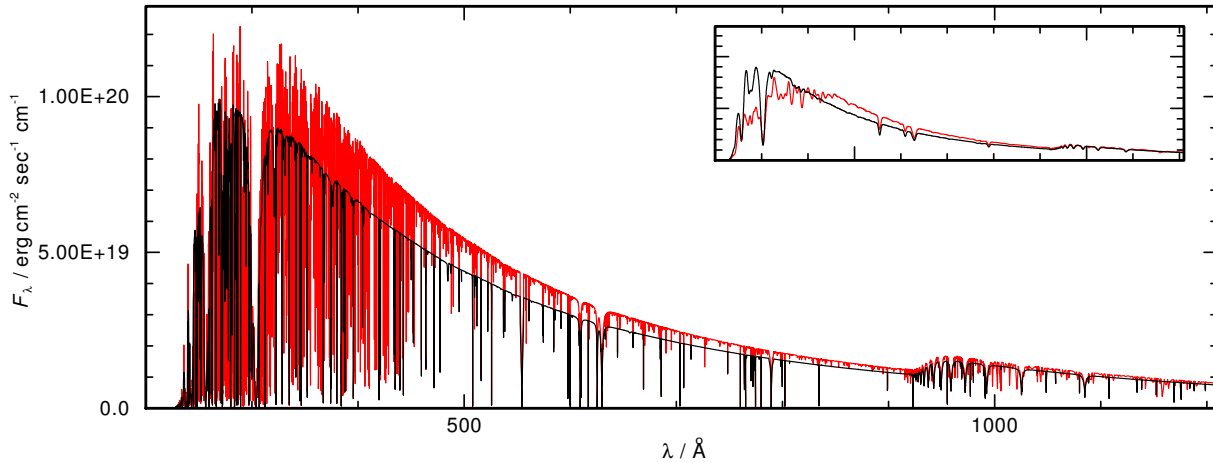


Abbildung 4.17.: Flussverlauf des Modells von LSE 263 ( $T_{\text{eff}} = 65\,000\text{ K}$ ,  $\log g = 5.25$ ) für eine HHeCNO- (schwarz) und eine HHeCNO+Eisengruppen-Atmosphäre (rot, gleiche Häufigkeiten). Die kleine Abbildung zeigt die selben Flüsse gefaltet mit einem Gaussprofil mit Halbwertsbreite  $5\text{ \AA}$ .

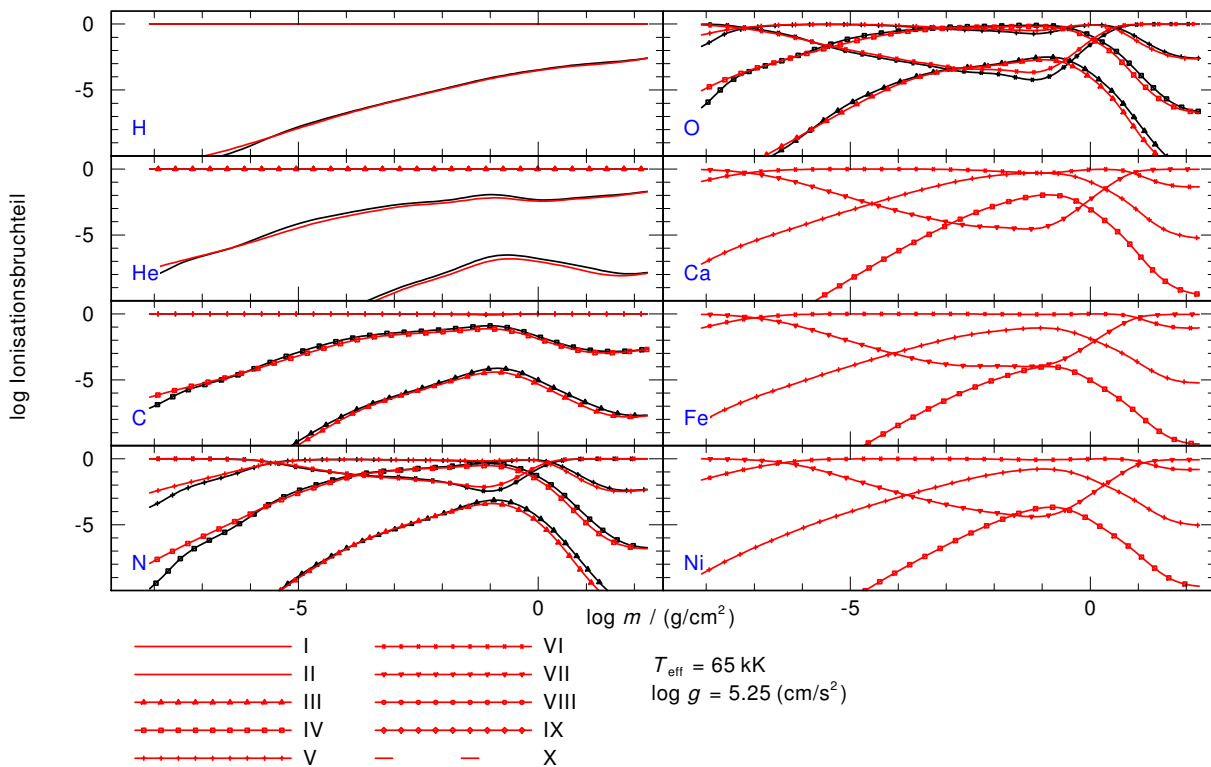


Abbildung 4.18.: Die Ionisationsbruchteile im Modell von LSE 263 ( $T_{\text{eff}} = 65\,000\text{ K}$ ,  $\log g = 5.25$ ) für ein HHeCNO- (schwarz) und ein HHeCNO+Eisengruppen-Modell (rot, gleiche Häufigkeiten).

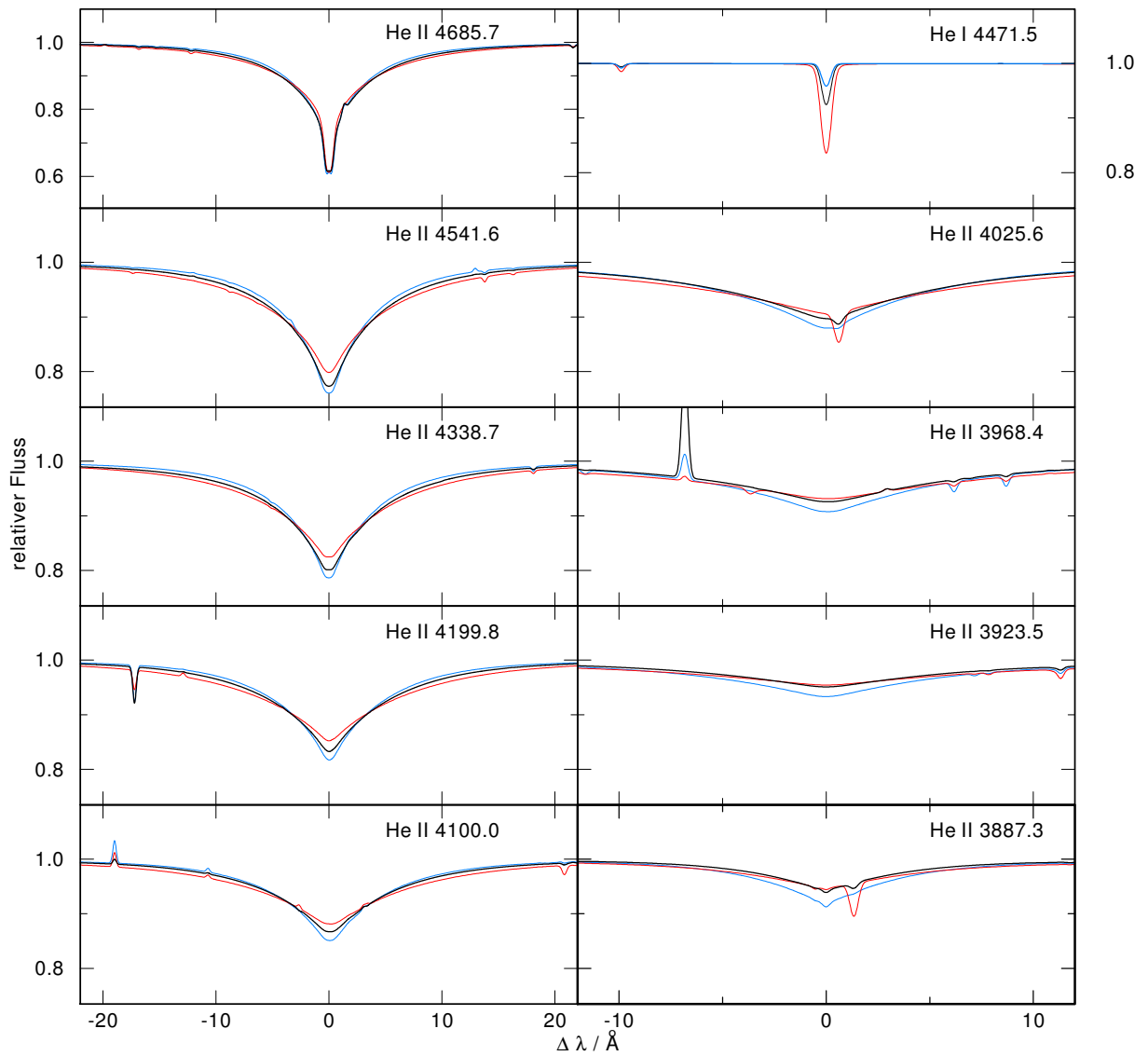


Abbildung 4.19.: Linienprofile von Helium für das Modell von LSE 263 mit Eisengruppenelementen ( $T_{\text{eff}} = 65\,000\text{ K}$ ,  $\log g = 5.25$ , schwarz, LIN-Linien, gleiche Häufigkeiten) verglichen mit den Fehlergrenzen aus der optischen Analyse (HHeCNO-Modell mit  $T_{\text{eff}} = 60\,000\text{ K}$ ,  $\log g = 5.5$  rot und  $T_{\text{eff}} = 70\,000\text{ K}$ ,  $\log g = 5.0$  in blau).

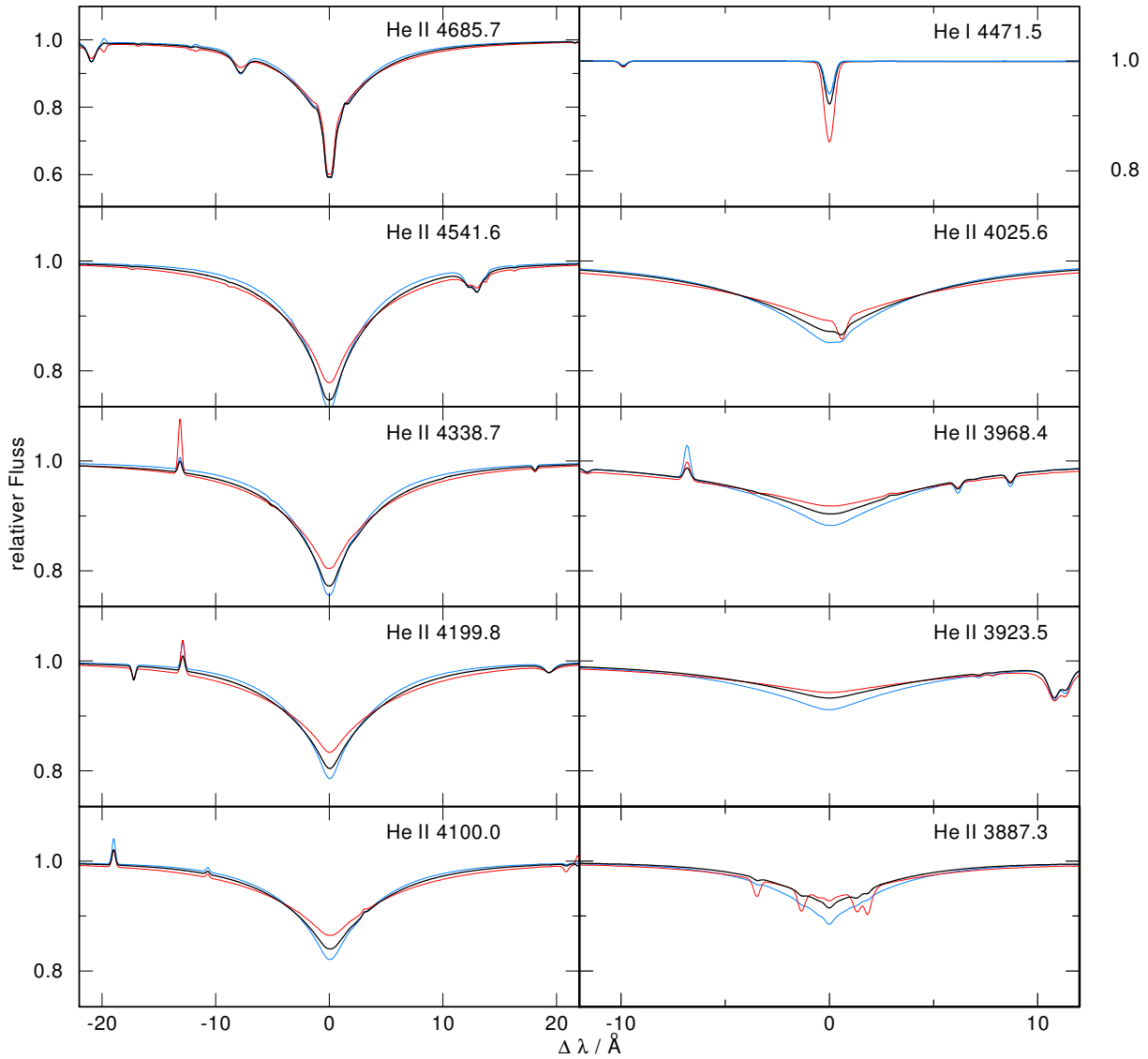


Abbildung 4.20.: Wie Abb. 4.19, für LSE 153 mit dem Modell mit Eisengruppenelementen ( $T_{\text{eff}} = 62\,500\text{ K}$ ,  $\log g = 5.00$ , schwarz, LIN-Linien) und den HHeCNO-Modellen mit  $T_{\text{eff}} = 60\,000\text{ K}$ ,  $\log g = 5.25$  (rot) und  $T_{\text{eff}} = 65\,000\text{ K}$ ,  $\log g = 4.75$  (blau).



## 4.2. Die PG 1159-Sterne PG 1424+535 und PG 1707+427

In diesem Kapitel werden die Spektren zweier PG 1159-Sterne untersucht. Die Besonderheit dieser Sterne ist, dass sie während ihrer Entwicklung beinahe ihre gesamte, ursprüngliche Hülle durch Massenverlust und Durchmischen der verbleibenden Hülle mit der heliumreichen Zwischenschale verloren haben und so Einsicht in tiefere, sonst unzugängliche Schichten bieten. Die Modellierung ihrer Spektren war bis zur Arbeit von Werner et al. (1991) nicht möglich, da NLTE-Modelle nötig sind, um die beobachteten Emissionskerne zu reproduzieren. Für zwei Vertreter dieser Klasse wurden HHeCNO-Modelle mit den Literaturwerten mit HHeCNO-Modellen von *TMAW* verglichen. Für die *TMAP*-Modelle wurde hier zusätzlich Neon in die Berechnung mit aufgenommen, weil Neon einen wichtigen Bestandteil von PG 1159-Atmosphären ausmacht. Anhand der *TMAP*-Modelle wurde ebenfalls der Einfluss der Metalle mittlerer Masse sowie der Eisengruppenelemente untersucht.

### 4.2.1. Entstehung und Entwicklung von PG 1159-Sternen

Bei etwa 75% der post-AGB-Sterne werden wasserstoffreiche Spektren beobachtet. Es existieren jedoch ebenfalls post-AGB-Sterne, die wasserstoffarm sind (z.B. Werner & Herwig, 2006). Diese Gruppe macht etwa 25% der post-AGB-Sterne aus und lässt sich weiter in Sterne mit helium- (z.B. die DO-Weißen Zwerge, Kapitel 4.4) oder kohlenstoffdominierter Atmosphäre unterteilen. Die Entwicklung von Sternen mittlerer Masse zu wasserstoffarmen (prä-) Weißen Zwergen weicht von der in Kapitel 1 beschriebenen Entwicklung zu wasserstoffreichen (prä-) Weißen Zwergen deutlich ab. Ein Szenario, mit dem vor allem Sterne mit kohlenstoffdominierten Atmosphären erklärt werden können, geht von einem zusätzlichen finalen thermischen Puls nach der AGB-Phase aus. Dieser bringt den Stern in die AGB-Phase zurück („born-again“ Szenario, Iben et al., 1983), wo der Wasserstoff der Hülle in tiefere Schichten gemischt oder verbrannt wird. Ein solcher finaler He-Schalenflash kann sich in drei Entwicklungsstadien ereignen:

- Findet der finale thermische Puls am Ende der AGB-Phase statt, wird er AGB Final Thermal Pulse (**AFTP**) genannt. Direkt nach diesem thermischen Puls verlässt der Stern den AGB. Aus diesem AFTP resultiert eine Konvektionszone, die sich über Zwischenschale und Hülle (Hüllenmasse etwa  $10^{-3} - 10^{-2} M_{\odot}$ ) des Sterns (Abb. 4.21) erstreckt. Sie verursacht, dass heliumreiches Zwischenschalenmaterial (einige  $10^{-3} M_{\odot}$ ) in die Wasserstoffhülle und wasserstoffreiches Material nach unten gemischt wird. Dadurch zeigt der resultierende Stern (ein so genannter hybrid PG 1159-Stern) noch etwa 20% Wasserstoff in der Atmosphäre. Vier solcher hybriden PG 1159-Sterne sind bekannt.
- Ein finaler thermischer Puls, der sich kurz vor Erreichen der Weißen Zwerg-Abkühlsequenz ereignet, wird Late Thermal Pulse (**LTP**) genannt. Durch die verursachte Konvektion wird – wie beim AFTP – Wasserstoff aus der Hülle (etwa  $10^{-4} M_{\odot}$ ) ins Zwischenschalengebiet gemischt. In der Photosphäre des resultierenden Sterns ist bei diesem Szenario aufgrund der kleineren Hüllenmasse nur noch ca. 1% Wasserstoff vorhanden, was nicht mehr beobachtbar ist.
- Sind die Brennprozesse bereits erloschen wenn der finale thermische Puls stattfindet, handelt es sich um einen Very Late Thermal Pulse (**VLTP**). Die Konvektionszone mischt den Wasserstoff noch tiefer in das Zwischenschalengebiet und dieser wird verbrannt. Eventuell verbleibende Wasserstoffreste verlassen den Stern durch den starken Wind in der AGB- und darauf folgenden Phase. Das resultierende Objekt besitzt keinen Wasserstoff mehr.

Es gilt als erwiesen, dass die kohlenstoff- und heliumdominierten PG 1159-Sterne durch so einen finalen thermischen Puls entstehen (z.B. Althaus et al., 2005, Abb. 4.22). Mittlerweile sind drei Objekte bekannt, von denen angenommen wird, dass sie gerade die born-again-Phase durchlaufen und zwei, die als Übergangsobjekte zwischen PG 1159- und dem Vorläuferstadium der [WC]-Sterne<sup>6</sup> klassifiziert sind. Nachfolger der PG 1159-Sterne sind nach dieser Theorie die heliumdominierten DO-Weißen Zwerge

<sup>6</sup>Diese Objekte zeigen Spektren, die von Windlinien dominiert sind und auf hohe Massenverlustraten schließen lassen.

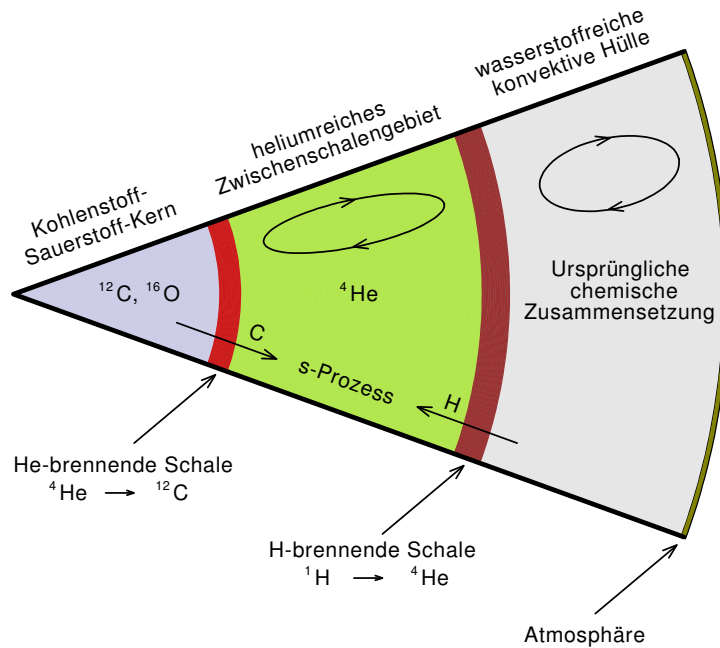


Abbildung 4.21.: Die innere Struktur eines Post-AGB-Sterns (Rauch, priv. Mitt.).

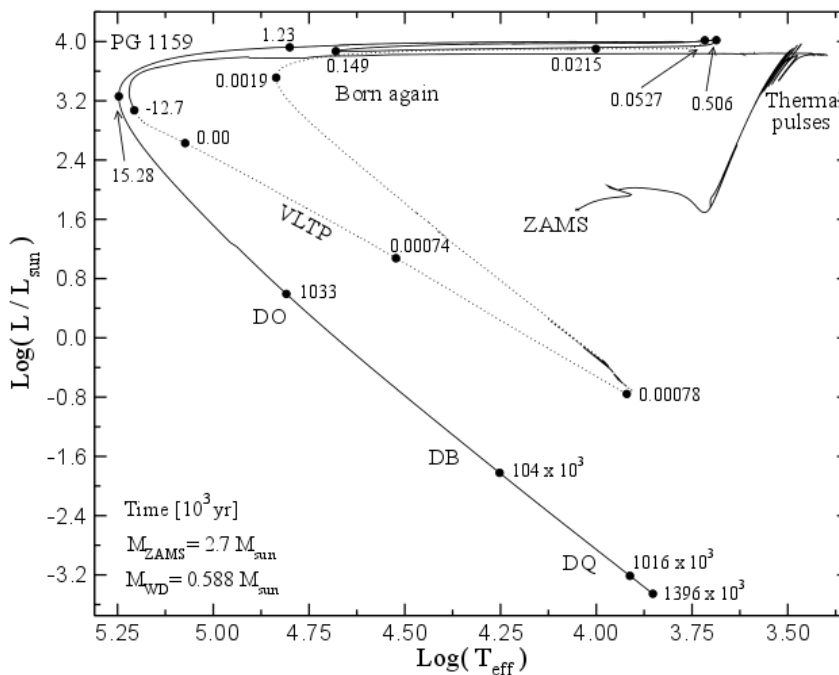


Abbildung 4.22.: Entwicklungsweg eines Sterns mit  $2.7 M_{\odot}$  Anfangsmasse (bei Beginn des Wasserstoffbrennens), der einen VLTP durchläuft (Althaus et al., 2005).

(siehe Kapitel 4.4), bzw. – falls sich noch genügend Wasserstoff im Stern befindet – wasserstoffdominierte DA-Weiße Zwerge.

Heute sind etwa 40 PG 1159 Sterne bekannt. Eine Übersicht über diese Objekte wird z.B. in Werner & Herwig (2006) gegeben. Sie sind nach ihrem Prototyp PG 1159–035 benannt und haben Temperaturen zwischen 75 000 K und 200 000 K und  $\log g$ -Werte zwischen 5.5 und 8.0. Ihre Spektren zeigen starke Helium- und Kohlenstoff- sowie Sauerstofflinien. Ein C IV- und He II-Absorptionstrog im opti-

schen Spektralbereich bei einer Wellenlänge von 4645 – 4686 Å ist charakteristisch für diese Objekte. In einigen PG 1159-Sternen sind außerdem deutliche P Cygni-Profile in den UV-Spektren zu finden, die auf Massenverlusten von ca.  $10^{-6.9} - 10^{-8.3} M_{\odot}$  pro Jahr schließen lassen (z.B. Koesterke et al., 1998; Koesterke & Werner, 1998; Herald et al., 2005).

Die Häufigkeitsbereiche der am stärksten vertretenen Elemente der Atmosphäre betragen He/C/O=0.30–0.85/0.15–0.60/0.02–0.20 (Massenbruchteile, Werner & Herwig, 2006). Im Bezug auf die N-Häufigkeit scheint es zwei Klassen von PG 1159-Sternen zu geben, eine mit hoher Häufigkeit (0.01 in Massenbruchteilen), eine mit deutlich niedrigerer (keine N-Linien zu sehen). Dies könnte ein Hinweis auf unterschiedliche Entwicklungsszenarien sein. Für VLTP-Sterne ist die theoretische N-Häufigkeit deutlich höher als für LTP-Objekte, da neuer Stickstoff durch das Verbrennen von Wasserstoff während des VLTP entsteht. Alle pulsierenden PG 1159-Sterne gehören zur Klasse der N-häufigen PG 1159-Objekte (Dreizler & Heber, 1998), was ein wichtiger Hinweis auf den bisher unverstandenen Antriebsmechanismus der Pulsationen sein könnte.

Zusätzlich zu den leichten Metallen konnten F-, Ne-, Si-, P-, S- und Ar-Linien in einigen PG 1159-Sternen identifiziert werden (Kapitel 4.2.2). Eisen konnte lange Zeit nicht in den Spektren der PG 1159-Sterne beobachtet werden. Obergrenzen, die zum Teil stark subsolar sind, konnten mit den Entwicklungsmodellen nicht erklärt werden. Grund für die subsolaren Obergrenzen war zum Teil, dass in den ersten Analysen hauptsächlich Fe VII-Linien modelliert wurden. Später konnten Werner et al. (2010) in den heißeren PG 1159-Sternen ( $T_{\text{eff}} \geq 150\,000$  K) durch die Berücksichtigung von Fe bis zur Ionisationsstufe X zum ersten Mal Eisen identifizieren und solare Häufigkeiten bestimmen. Für die PG 1159-Sterne mittlerer Temperatur (110 000 K – 150 000 K) konnte mit den neuen Atomdaten von Kurucz Fe VII und Fe VIII detektiert und eine solare Eisenhäufigkeit bestimmt werden (Werner et al., 2011). Für die kühleren PG 1159-Sterne hat sich die Situation allerdings nicht geändert.

#### 4.2.2. PG 1424+535 und PG 1707+427

PG 1424+535 und PG 1707+427 gehören zur Klasse der PG 1159-Sterne und wurden beide im Palomar-Green Survey (Green et al., 1986) entdeckt. PG 1707+427 ist ein pulsierender Stern (er gehört zur Klasse der so genannten GW Vir-Veränderlichen, Bond et al., 1984; Grauer et al., 1989), PG 1424+535 nicht. In den ersten Analysen (Wesemael et al. (1985) mit LTE-Modellen, Werner et al. (1991) mit NLTE-Modellen) wurden ihre Parameter zu  $T_{\text{eff}} = 100\,000$  K und  $\log g = 7$  bestimmt. Eine spätere Analyse der beiden Objekte von Dreizler & Heber (1998) (zusätzlich anhand von GHRS-Spektren<sup>7</sup> führte zu  $T_{\text{eff}} = 110\,000$  K,  $\log g = 7.0$  für PG 1424+535 und  $T_{\text{eff}} = 85\,000$  K,  $\log g = 7.5$  für PG 1707+427. Die aus dieser Analyse resultierenden unterschiedlichen Stickstoffhäufigkeit der beiden Objekte ( $N/He = 0.01$  in Teilchenzahlbruchteilen für PG 1707+427 und  $N/He < 10^{-5}$  für PG 1424+535) könnte darauf hindeuten, dass PG 1707+427 einen VLTP, PG 1424+535 einen LTP durchlaufen hat. In den folgenden Jahren konnten in beiden Objekten F, Ne, Si, P, S, Ar und Fe detektiert bzw. Obergrenzen bestimmt werden (Tabelle 4.5, Werner et al., 2004, 2005; Reiff et al., 2005, 2006a,b, 2007; Werner et al., 2007; Reiff et al., 2008; Werner et al., 2008, 2010, 2011). Die bestimmten Häufigkeiten der Elemente C, N und O liegen innerhalb der für PG 1159-Sterne typischen Werte. Die Häufigkeiten bzw. Obergrenzen von F, Si, Ar und Fe stimmen mit Entwicklungsrechnungen überein, während die von P und S bisher nicht erklärt werden können.

#### Das RISE-Projekt

Der Deutsche Akademische Austauschdienst (DAAD) bietet deutschen Doktoranden und englisch muttersprachigen Studenten des selben naturwissenschaftlichen Fachgebiets die Möglichkeit, für acht bis zwölf Wochen in Deutschland zusammen an einem Projekt des Doktoranden zu arbeiten. Dieses DAAD-

<sup>7</sup><http://www.stsci.edu/hst/ghrs/>

Tabelle 4.5.: Die Elementhäufigkeiten von in PG 1424+535 und PG 1707+427 in Massenbruchteilen. Die Literaturwerte wurden Dreizler & Heber (1998); Werner et al. (2004, 2005); Reiff et al. (2005, 2006b, 2007); Werner et al. (2007); Reiff et al. (2008); Werner et al. (2011) entnommen.

	PG 1424+535		PG 1707+427	
	Literaturwert	diese Arbeit	Literaturwert	diese Arbeit
C	$4.4 \times 10^{-1}$	$3.3 \times 10^{-1}$	$3.9 \times 10^{-1}$	$3.8 \times 10^{-1}$
N	$< 1.0 \times 10^{-5}$	$< 2.5 \times 10^{-6}$	$1.5 \times 10^{-2}$	$1.4 \times 10^{-3}$
O	$6.0 \times 10^{-2}$	$3.0 \times 10^{-2}$	$1.7 \times 10^{-1}$	$1.8 \times 10^{-2}$
F	$1.0 \times 10^{-4}$	$4.1 \times 10^{-5}$	$1.0 \times 10^{-4}$	$2.0 \times 10^{-4}$
Ne	$< 2.0 \times 10^{-2}$	$< 1.6 \times 10^{-2}$	$< 2.0 \times 10^{-2}$	$< 2.0 \times 10^{-2}$
Si	$3.6 \times 10^{-4}$	$9.4 \times 10^{-4}$	$7.3 \times 10^{-4}$	$4.5 \times 10^{-3}$
P	$9.5 \times 10^{-6}$	$1.0 \times 10^{-5}$	$2.0 \times 10^{-6}$	$6.0 \times 10^{-6}$
S	$5.0 \times 10^{-5}$	$1.6 \times 10^{-4}$	$2.5 \times 10^{-5}$	$2.8 \times 10^{-4}$
Ar	$3.2 \times 10^{-5}$	$2.9 \times 10^{-5}$		$< 4.2 \times 10^{-6}$
Fe	$1.3 \times 10^{-3}$	$1.3 \times 10^{-3}$	$< 1.3 \times 10^{-3}$	$< 1.3 \times 10^{-3}$
Ni		$< 7.1 \times 10^{-5}$	$< 7.1 \times 10^{-5}$	$< 7.1 \times 10^{-5}$

Projekt trägt den Namen Research Internships in Science and Engineering (RISE<sup>8</sup>). Im Rahmen dieses Projekts kam Sarah Bartram im Sommer 2011 für acht Wochen als Forschungsassistentin nach Tübingen, um mit mir an der Analyse dieser beiden PG 1159-Objekte zu arbeiten. Ziel des Projekts war es, eine begonnene Analyse (Reiff et al., 2007, 2008) mit den mittlerweile aktualisierten Atomdaten zu überprüfen und abzuschließen. Insbesondere hatte ein aktualisiertes O-Modellatom bereits bei anderen Analysen zu einer deutlich höheren Effektivtemperatur geführt (z.B. für den Zentralstern des PN K 1-16 wurde eine 30 000 K höhere  $T_{\text{eff}}$  bestimmt, Ringat, 2010). Zusammen mit Sarah Bartram wurden deshalb HeCNONE-Modelle mit aktualisierten Atomdaten neu berechnet. Dabei stellte sich heraus, dass mit den neuen Modellatomen das in den bisher verwendeten Modellatmosphären berücksichtigte  $\tau$ -Intervall nicht ausreichte. Bei *TMAP* wird die Atmosphäre so gewählt, dass mindestens die zwei äußersten Tiefenpunkte optisch dünn sind. Einzelne Linien entstehen durch die neuen Modellatome allerdings weiter außen als bisher. Die  $\log m$ - und  $\log \tau$ -Skala der Modellatmosphären wurde aus diesem Grund mehrmals erweitert.

Zur Fehlerabschätzung und Bestimmung von  $T_{\text{eff}}$  und  $\log g$  wurden Modellgitter mit  $\Delta T_{\text{eff}} = 5000$  K und  $\Delta \log g = 0.3$  (für PG 1424+535) bzw.  $\Delta T_{\text{eff}} = 10000$  K und  $\Delta \log g = 0.5$  (für PG 1707+427) angelegt. Die unterschiedlichen Schrittweiten der Gitter wurden gewählt, weil die verfügbaren optischen Spektren von PG 1707+427 ein deutlich schlechteres Signal-zu-Rausch-Verhältnis als die von PG 1424+535 zeigen, wodurch auch die Fehlergrenzen größer sind. Die verwendeten optischen Spektren wurden zwischen 1987 und 1990 am Calar Alto 3.5 m-Teleskop mit dem Boller&Chivens-Spektrographen aufgenommen und aufaddiert. Zusätzlich werden zur Analyse von He II  $\lambda 6560$  Å von PG 1707+427 2001 mit dem TWIN-Spektrographen aufgenommene Spektren verwendet, da die Boller&Chivens-Spektren von PG 1707+427 diesen Bereich nicht abdecken. Für die UV-Analyse steht jeweils ein FUSE- (ID: P1320301000 bzw. P1320401000) und ein GHRS-Spektrum (ID: Z2T20204T bzw. Z2T20304T) zur Verfügung.

Zu Anfang der Analyse zeigte sich, dass die Modelle mit den Literaturwerten (Tabelle 4.5) die Beobachtungen größtenteils reproduzieren konnten. Deshalb wurden die Parameter der Modellatmosphären zunächst nicht verändert und es wurden, nachdem die Struktur der Modelle sich nicht mehr signifikant

<sup>8</sup><http://www.daad.de/rise/de>

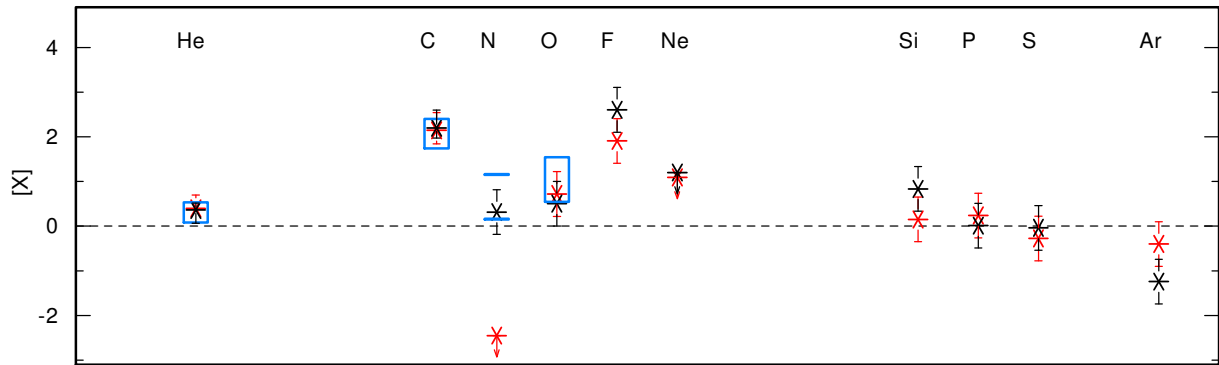


Abbildung 4.23.: Die Elementhäufigkeiten von PG 1424+535 (rot) und PG 1707+427 (schwarz).  $[X]$  steht für die Häufigkeit des Elements X relativ zum solaren Wert. Blau sind die typischen Häufigkeitswerte für PG 1159-Sterne dargestellt. Für Stickstoff wurde jeweils die erwartete Häufigkeit aus einem VLTP (oben) und einem LTP eingezeichnet (blaue Striche). Die gestrichelte Linie stellt den solaren Wert dar.

änderte, die Häufigkeiten der Spurenelemente F, Si, P, S und Ar (analog zu E. Reiff) durch einen Lineformationschritt in dem finalen Modell bestimmt (Tabelle 4.5).

Die Fortsetzung der Berechnung der Modelle ergab nach Ende unseres RISE-Projekts  $T_{\text{eff}} = 115\,000\text{ K}$  und  $\log g = 7.3$  für PG 1424+535. Deshalb wurde das Gitter um  $T_{\text{eff}} = 120\,000\text{ K}$  und  $\log g = 7.6$  erweitert.

In das finale Modell wurden ebenfalls nach Ende des RISE-Projekts für jedes Objekt die Spurenelemente mit den in der Lineformationsrechnung bestimmten Häufigkeiten sowie die Eisengruppenelemente (Fe und Ni individuell, die anderen generisch) mit solaren Häufigkeit eingefügt, um ihren Einfluss auf die Atmosphärenstruktur zu untersuchen. Die solaren Häufigkeiten von Ca, Sc, Ti, V, Cr, Mn und Co stehen nicht im Widerspruch zur Beobachtung, eine genaue Analyse dieser Elemente übersteigt jedoch den Rahmen dieser Arbeit.

Obwohl bisher weder Fe in PG 1707+427 noch Ni in den beiden Objekten identifiziert werden konnte, wurde hier aufgrund der erweiterten Kurucz-Daten ein neuer Versuch unternommen ihre Häufigkeiten zu bestimmen bzw. Linien zu identifizieren. Beide Elemente wurden solar in der Berechnung berücksichtigt.

In diesen beiden finalen Modellen wurden die Häufigkeiten aller Elemente angepasst (Tabelle 4.5, Abb. 4.23). Vor allem für C, N und O sind die Auswirkungen auf die Atmosphäre am stärksten, da sie Hauptbestandteil der Atmosphäre sind (Abb 4.31). Die neu bestimmten Werte dieser Elemente (Abb. 4.24 und 4.25) sind niedriger als die Literaturwerte, die C- und O-Häufigkeiten liegen aber immer noch in den für PG 1159-Sterne üblichen Häufigkeitsbereichen (Abb. 4.23). Für PG 1424+535 sind in der optischen Beobachtung zwei O V-Multipletts bei  $4450\text{ \AA}$  und  $5850\text{ \AA}$  und zu sehen, die deutlich stärker als die Beobachtung sind. Da diesen Linienübergängen Opacity Project-Daten zugrunde liegen, die eine Unsicherheit von ca. 10% in der Oszillatorenstärke haben, wurden diese Linien ignoriert. Für beide Objekte wurde der Stickstoffwert nach unten korrigiert. Damit liegt PG 1707+427 näher an den für einen LTP typischen Häufigkeiten als an denen für einen VLTP.

Die Häufigkeiten der Spurenelemente sind im Vergleich zu den (mit einer Lineformationsrechnung bestimmten) Literaturwerten für F, Ne und Ar in etwa in Übereinstimmung. Die hier bestimmte Häufigkeit von Si, P und S ist höher als der Literaturwert und liegt für Si bei PG 1707+427 außerhalb der Fehlergrenzen (0.5 dex). Für Fe konnte die Obergrenze bzw. Häufigkeit bestätigt werden. Im folgenden werden die einzelnen Elemente diskutiert und die gefundenen Werte mit Vorhersagen aus Sternentwicklungsrechnungen (Werner & Herwig, 2006 und Referenzen darin) verglichen.

**Neon** ist nur in der Beobachtung von PG 1424+535 zu sehen. Die stärkste Neon-Linie liegt bei  $973.3\text{ \AA}$  (Ne VII) und ist bei beiden Objekten eventuell von einer interstellaren Linie überlagert. Aufgrund der

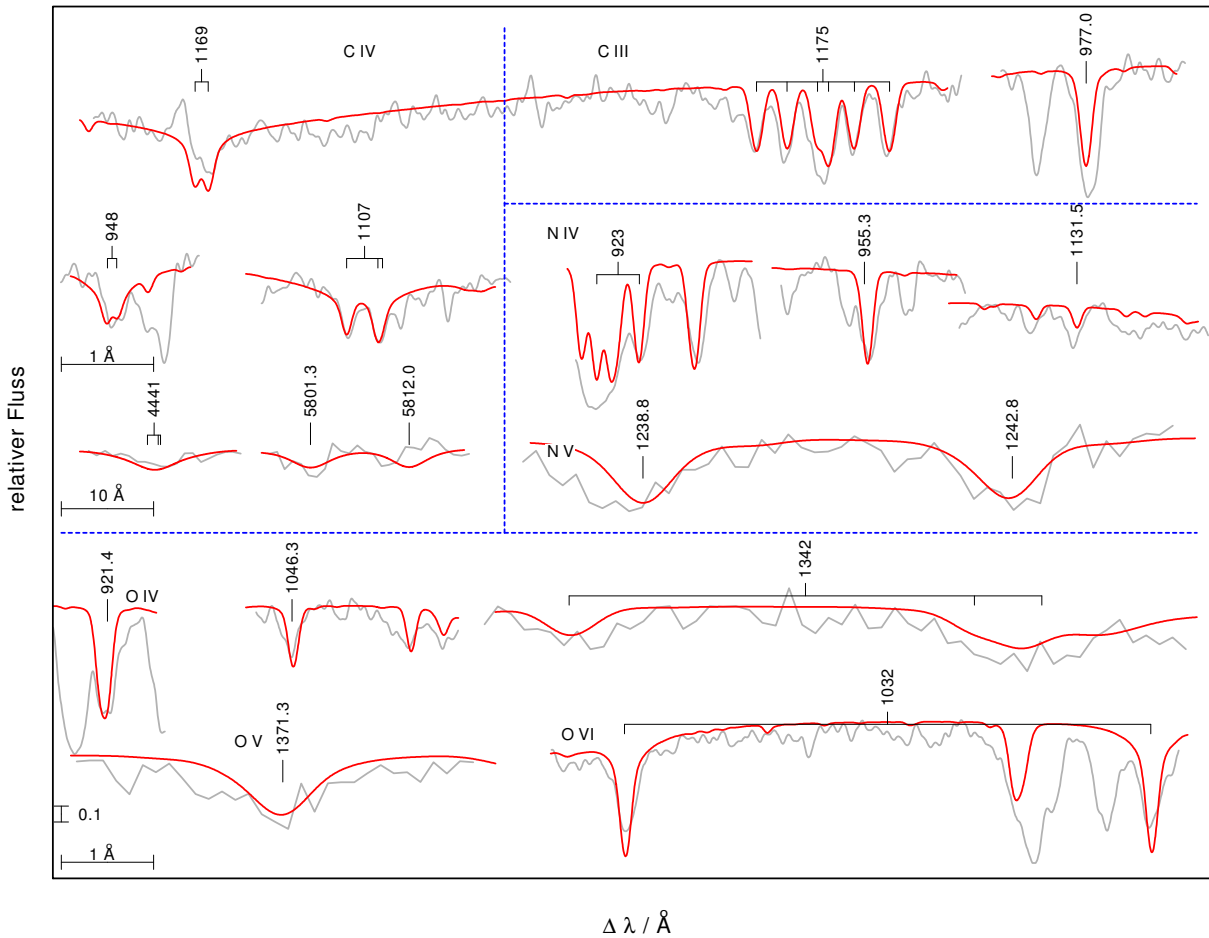


Abbildung 4.24.: Die stärksten C-, N- und O-Linien von PG 1707+427 im finalen HHeCNeFSiPSArCaFeNi-Modell (rot). Das Modell wurde mit einem Gaussprofil der Halbwertsbreite  $0.085 \text{ \AA}$ , die Beobachtung (zur Glättung) mit  $0.06 \text{ \AA}$  gefaltet. Die jeweiligen Achsenabschnitte sind durch die Markierungen dargestellt.

niedrigeren  $T_{\text{eff}}$  von PG 1707+427 wurde Ne VII in diesem Modell nicht berücksichtigt. Die Literaturhäufigkeit führt jedoch nicht zu Widersprüchen der Ne VI-Linien des Modells mit der Beobachtung.

**Fluor** zeigt die stärksten Linien bei  $1139.5 \text{ \AA}$  (F VI) und bei  $1086 \text{ \AA}$  (F V). Die Literaturwerte für F konnten innerhalb der Fehlergrenzen bestätigt werden. Auch hier ist PG 1707+427 zu kühl, um die stärkste F VI Linie zu zeigen, die F V-Linien bestätigen aber die früheren Ergebnisse (Abb. 4.26). Ein Vergleich der Häufigkeiten der beiden Objekte zeigt, dass PG 1707+427 mehr F in der Atmosphäre zeigt als PG 1424+535. Dies könnte einen Hinweis auf die Entstehung von F während der AGB-Phase liefern. Da bisher angenommen wird, dass Stickstoff in Fluor umgewandelt wird, sollte PG 1707+427 weniger Stickstoff als PG 1424+535 zeigen. Die F-Produktion ist jedoch auch von der Masse des Sterns abhängig, die hier nicht betrachtet wird.

**Silizium** ist in beiden Sternen zu finden. Die neu bestimmten Si-Häufigkeiten sind deutlich höher als die Literaturwerte, diese liegen aber für PG 1424+535 immer noch in den Fehlergrenzen. Theoretisch wird während der AGB-Phase kein Si produziert, der erwartete Wert ist also solar, die hier bestimmten Häufigkeiten leicht supersolar.

**Phosphor** kann ebenfalls in beiden Objekten beobachtet werden. Die neuen Häufigkeiten sind etwas höher als die von Reiff et al. (2007) bestimmten. Die neuen Werte liegen damit näher an den erwarteten vier- bis 25-fach solaren Werte, die Diskrepanz zwischen theoretisch vorhergesagter und beobachteter

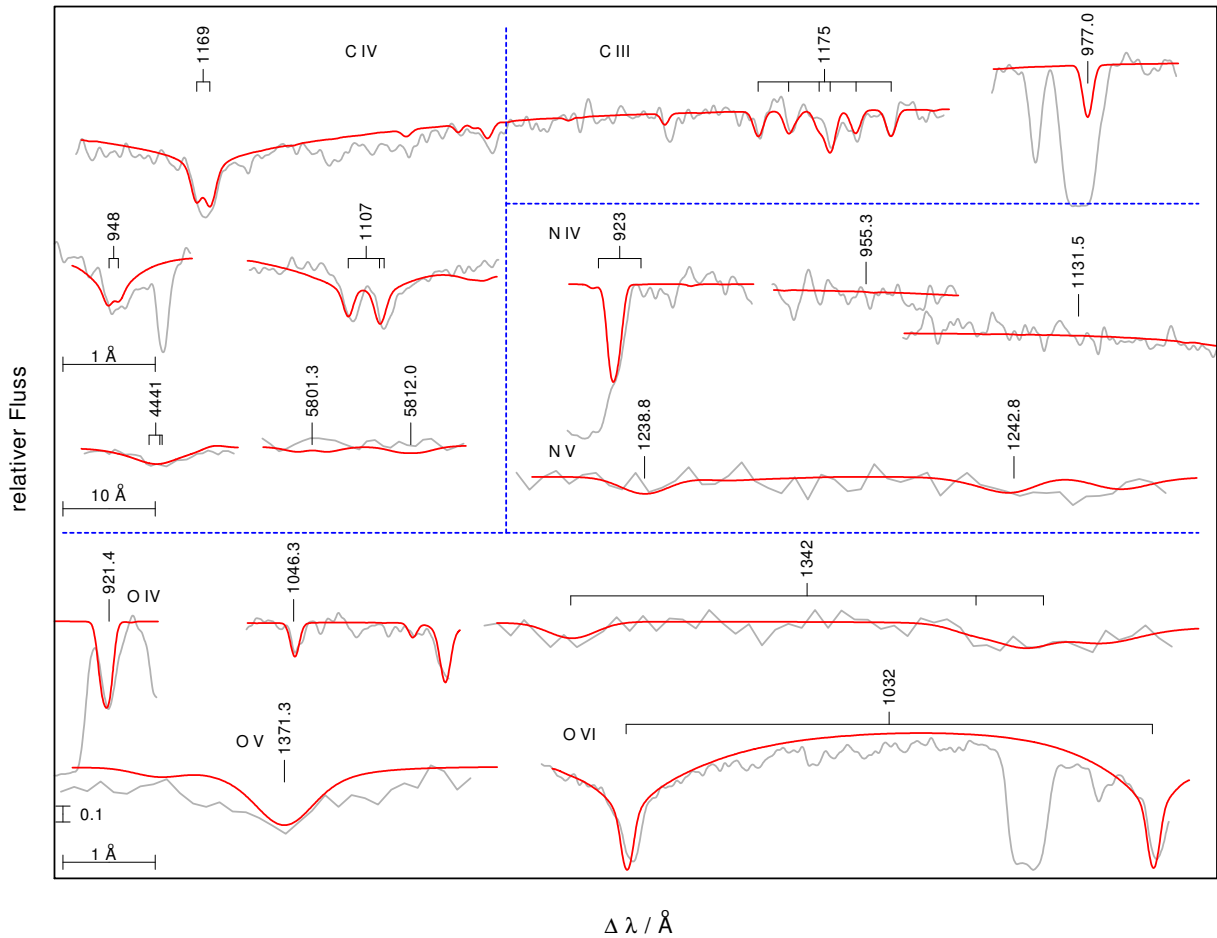


Abbildung 4.25.: Wie Abb. 4.24, für PG 1424+535.

Häufigkeit ist allerdings immer noch nicht gelöst.

**Schwefel** kann mit subsolar Häufigkeit detektiert werden. Insgesamt zeigen die PG 1159-Sterne eine bisher nicht erklärbare große Streuung beobachteter S-Häufigkeiten. Der erwartete 0.6- bis 0.9-fach solare Wert wird für beide Objekte in etwa getroffen. Wie für Si und P sind die S-Häufigkeiten (Abb. 4.27) deutlich höher als die Literaturwerte.

**Argon** konnte bisher in keinem PG 1159-Stern außer PG 1424+535 nachgewiesen werden. Die stärkste Linie liegt bei 1063.5 Å (Ar VII, Abb. 4.26). Die in dieser Arbeit bestimmte Häufigkeit stimmt mit dem Literaturwert überein und ist mit Vorhersagen aus Entwicklungsrechnungen konsistent (etwa solare Häufigkeit, Werner et al., 2007 und Referenzen darin). Bei PG 1707+427 ist diese Ar-Linie von interstellaren Linien überlagert, weshalb hier nur eine Obergrenze bestimmt wurde.

**Eisen und Nickel** sind bisher (in PG 1707+427) mit solaren Obergrenzen belegt worden. In PG 1424+535 wurde bereits Eisen mit solarer Häufigkeit detektiert, was durch die Modelle dieser Arbeit bestätigt wird (Abb. 4.28). Bei PG 1707+427 liefert das Modell mit solaren Häufigkeiten keinen Widerspruch zur Beobachtung (Abb. 4.29). Die solaren Eisenhäufigkeiten sind mit Entwicklungsrechnungen verträglich.

Für Ni gestaltet sich die Analyse schwieriger. Die Ursache dafür ist, dass aufgrund fehlender Atomdaten für Ni VII ausschließlich Linien von Ni VI analysiert werden können, was (für den heißeren der beiden Sterne) nicht der dominanten Ionisationsstufe entspricht. Obwohl sich PG 1707+427 für diesen Zweck besser eignet, konnten auch hier keine Ni-Linien in der Beobachtung gefunden werden (Abb. 4.30), eine solare Häufigkeit liefert jedoch keinen Widerspruch zur Beobachtung.

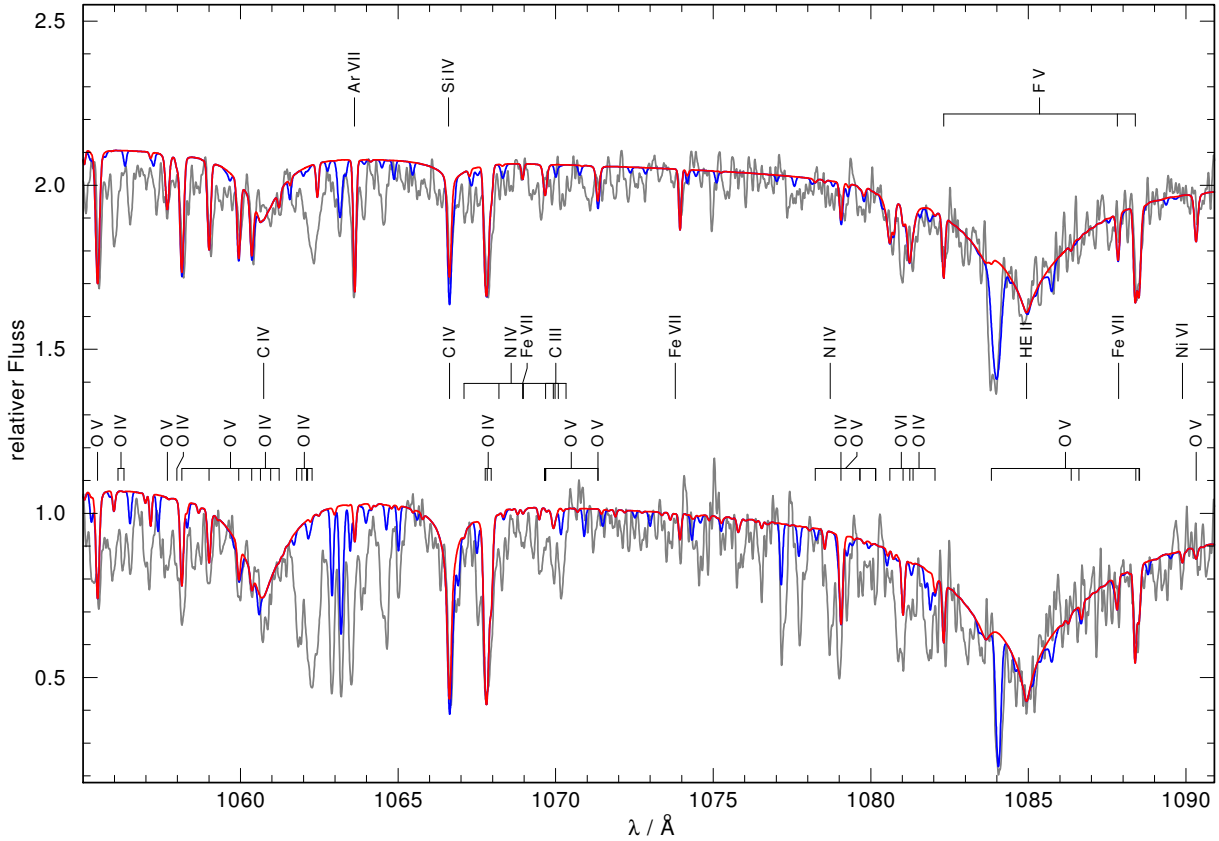


Abbildung 4.26.: Ausschnitte aus den FUSE-Beobachtungen von PG 1424+535 (oben) und PG 1707+427 (unten) verglichen mit den finalen HHeCNOFeSiPSArCaFeNi-Modellen (rot). Blau dargestellt sind die interstellaren Linien, modelliert mit *OWENS*. Die Modelle wurden mit einem Gaussprofil der Halbwertsbreite 0.085 Å, die Beobachtungen mit 0.06 Å gefaltet.

Ein Vergleich der Temperaturstrukturen der HHeCNO- und HHeCNOFeSiPSArCaFeNi-Modelle von PG 1707+427 und PG 1424+535 zeigt, dass die Berücksichtigung der Spurenelemente in der Modellatmosphärenrechnung in diesem Parameterbereich für  $\log m > -4$  nur geringe Auswirkung auf die Atmosphäre hat (Abb. 4.31). In diesem  $\log m$ -Bereich entstehen die meisten Linien der Spurenelemente. Die Bestimmung der Häufigkeiten in der Lineformationrechnung oder bei Berücksichtigung in der Modellatmosphäre führt für die meisten Spurenelemente zu einem innerhalb der Fehlergrenzen übereinstimmenden Wert.

Für die Entstehungstiefen der optischen Heliumlinien weichen die Temperaturstrukturen des HHeCNO- und des HHeCNOFeSiPSArCaFeNi-Modells mit gleichen Häufigkeiten am stärksten für He II  $\lambda 4860.7$  Å ab. Insgesamt liegen die optischen Helium-Linienprofile der HHeCNOFeSiPSArCaFeNi-Modelle innerhalb der durch die HHeCNO-Grenzmodelle der Analysen gegebenen Bereiche (Abb. 4.34, mit Ausnahme von PG 1424+535 dessen HHeCNOFeSiPSArCaFeNi-Modell für He II  $\lambda 6560.1$  Å genau auf dem Grenzmodell liegt). Eine Analyse mit HHeCNO- anstelle von HHeCNOFeSiPSArCaFeNi-Modellen liefert also  $T_{\text{eff}}$ - und  $\log g$ -Werte innerhalb dieser Fehlergrenzen. Für die UV-Linien, die wichtig zur Temperaturbestimmung sind, liegen die Kohlenstofflinien innerhalb der Fehlergrenzen (Abb. 4.32). Werden die Sauerstoffhäufigkeiten im synthetischen Spektrum variiert, so liegen die Linien des HHeCNOFeSiPSArCaFeNi-Modells ebenfalls innerhalb der durch die Fehlergrenzen gegebenen Linien.

Insgesamt kann geschlossen werden, dass eine Analyse mit HHeCNO- anstelle von



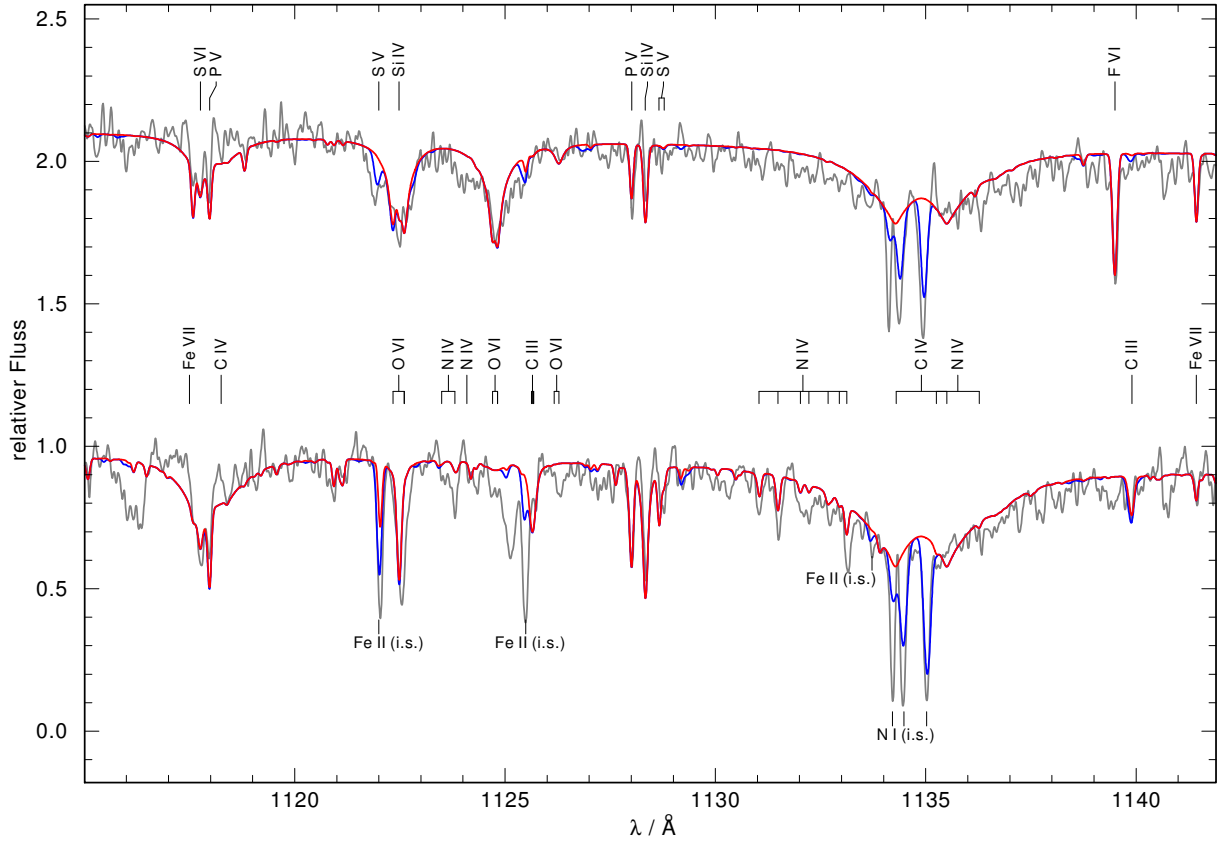


Abbildung 4.27.: Wie Abb. 4.26, für einen anderen Wellenlängenbereich.

HHeCNONeSiPSArCaFeNi-Modellen zu Abweichungen für  $T_{\text{eff}}$  zwischen maximal 4.3% und 11.8% und für  $\log g$  zwischen maximal 4.1% und 6.7% für PG 1424+535 bzw. PG 1707+427 führt.

Für eine Abschätzung der Genauigkeit von *TMAW*-Spektren in diesem Parameterbereich wurde jeweils das finale HHeCNONE-Modell mit *TMAW* berechnet. Bei diesem Vergleich sind nun im Gegensatz zum vorherigen Vergleich der *TMAW*- und *TMAP*-Modelle von LSE 153 und LSE 263 nicht mehr genau die selben Elemente in den *TMAW*- und *TMAP*-Modellen berücksichtigt. Bei der Betrachtung der Temperaturstrukturen der *TMAW*- und *TMAP*-Modelle zeigt sich, dass sie vor allem in der äußeren Atmosphäre ( $\log m < -2.5$ ) voneinander abweichen. Da He II  $\lambda 6560.1 \text{ \AA}$  am weitesten außen entsteht, sind die Abweichungen für diese Linie am stärksten (Abb. 4.37). Wie bei dem Vergleich mit den HHeCNONEFSiPSArCaFeNi-Modellen liegen die *TMAW*-Modelle von PG 1707+427 und PG 1424+535 für die Linienprofile der dargestellten Heliumlinien innerhalb der Grenzmodelle (mit Ausnahme des innersten Linienkerns der He II  $\lambda 6560.1 \text{ \AA}$  von PG 1707+427, der nur wenig schwächer ist). Die dargestellten optischen He II-Linien sind vor allem zur Bestimmung von  $\log g$  wichtig. Die zur Temperaturbestimmung wichtigen UV-Linien aus Abb. 4.35 und Abb. 4.36 zeigen das gleiche Ergebnis wie beim Vergleich mit dem HHeCNONEFSiPSArCaFeNi-Modell. Mit einer Variation der Sauerstoffhäufigkeit um 0.5 dex liegen die Modelle innerhalb der Fehlergrenzen.

Daraus kann eventuell geschlossen werden, dass die Berücksichtigung von Neon starke Auswirkungen auf die Analyse haben kann. Im Gegensatz zu den Ergebnissen des Vergleichs von LSE 153 und LSE 263 sind die Abweichungen des *TMAW*-Modells bei diesen Objekten größer als die des Modells, das alle Elemente berücksichtigt. Das Ergebnis der Analyse liegt innerhalb von 4.3% und 11.8% für  $T_{\text{eff}}$  und 4.1% und 6.7% für  $\log g$  für PG 1424+535 bzw. PG 1707+427. Der *TMAW*-Dienst kann also auch für zuverlässige Spektralanalysen von PG 1159-Sternatmosphären verwendet werden.

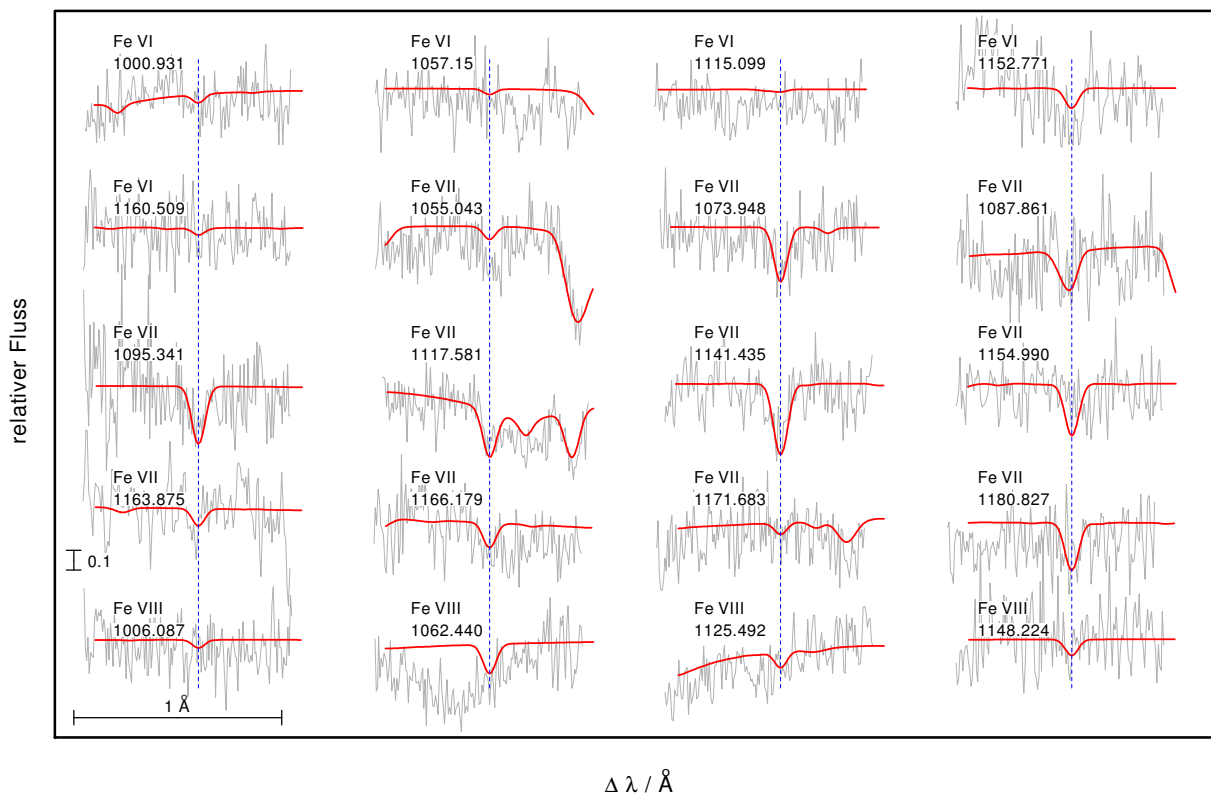


Abbildung 4.28.: Die stärksten Eisenlinien von PG 1424+535 im finalen HHeCNONEFSiPSArCaFeNi-Modell (rot, solare Eisenhäufigkeit). Die jeweiligen Achsenabschnitte sind durch die Markierungen dargestellt.

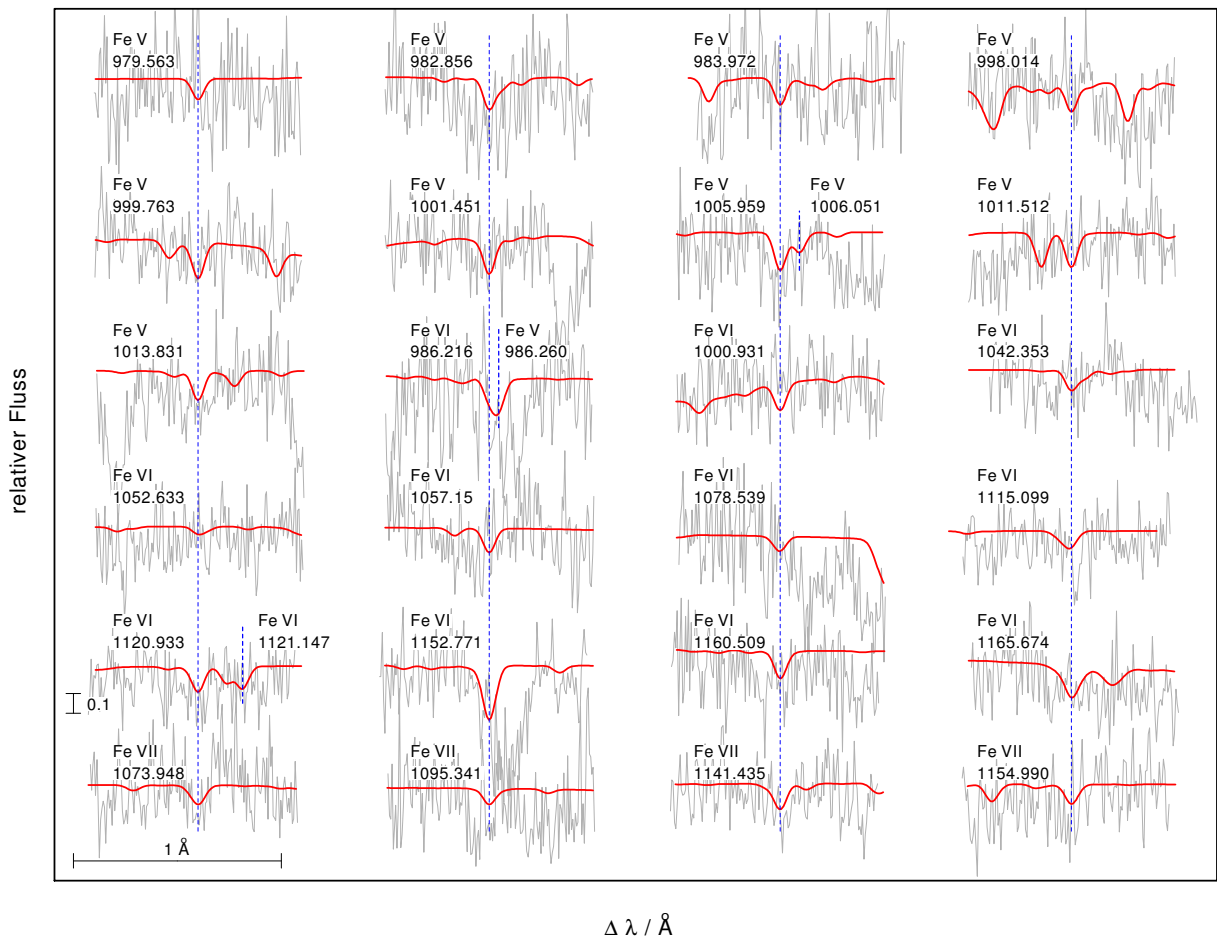


Abbildung 4.29.: Wie Abb. 4.28, für PG 1707+427.

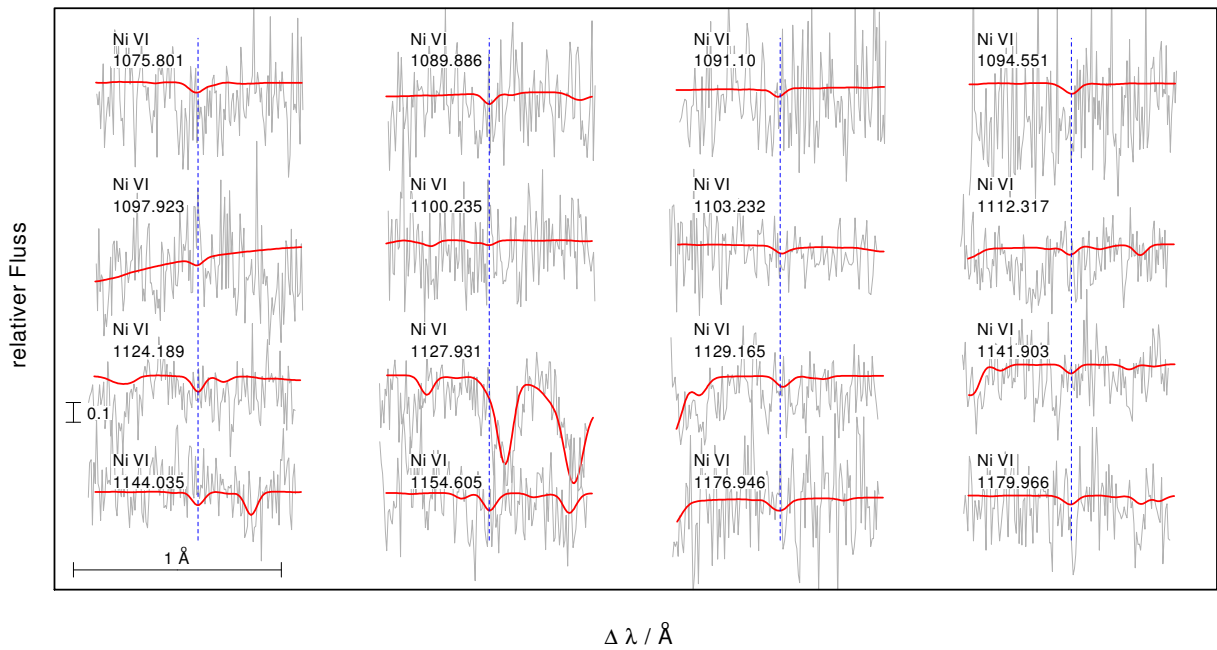


Abbildung 4.30.: Wie Abb. 4.28, für die Nickellinien (solare Häufigkeit) von PG 1707+427.

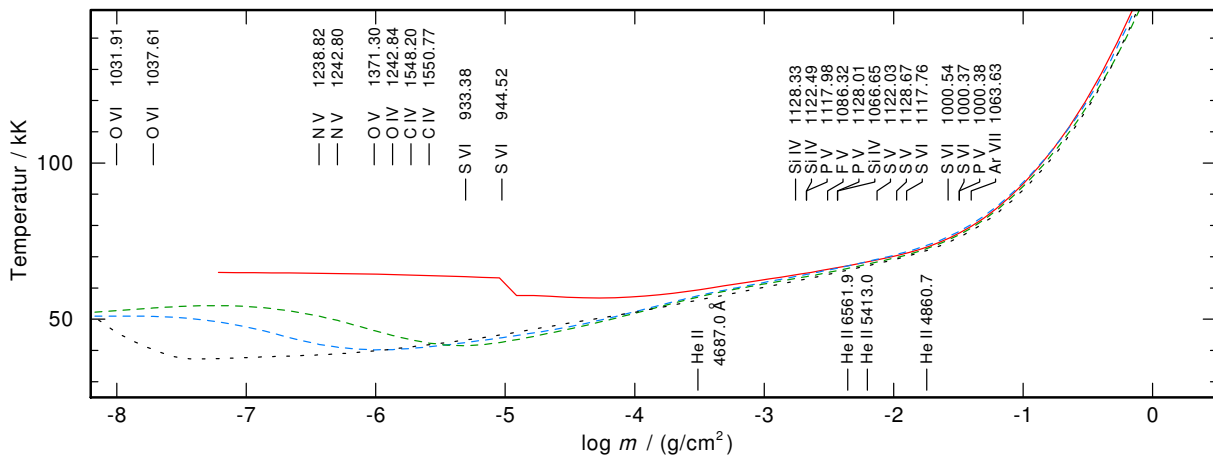


Abbildung 4.31.: Die Temperaturstrukturen der TMAW- (rot), HHeCNe- (schwarz) und HHeCNeFSiPSArCaFeNi-Modelle (blau) von PG 1707+427. Zusätzlich ist das HHeCNeFSiPSArCaFeNi-Modell mit variierten Häufigkeiten abgebildet (grün).

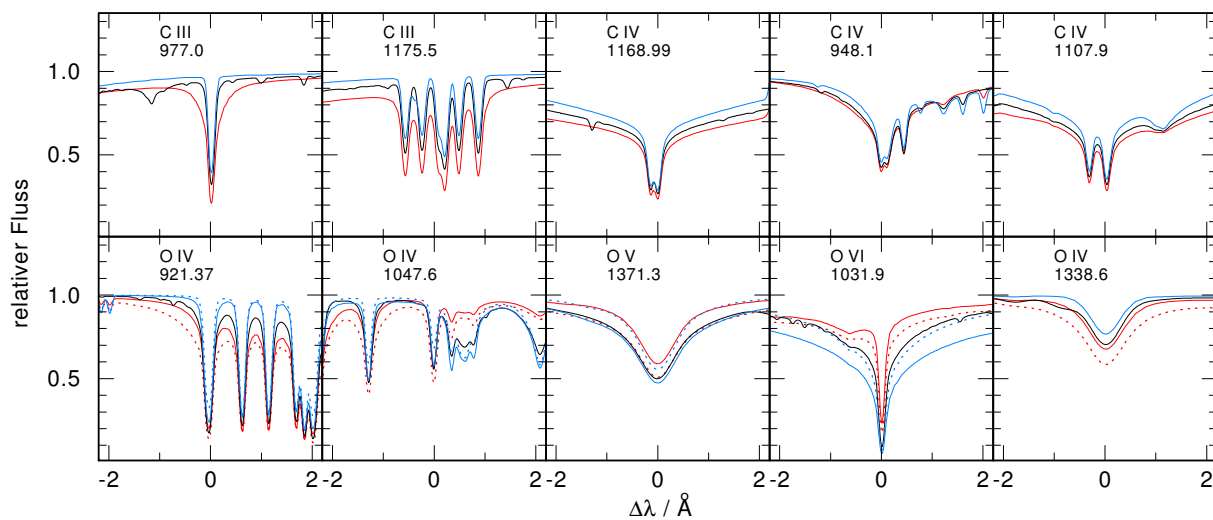


Abbildung 4.32.: Zur Bestimmung der Effektivtemperatur wichtige UV-Linien verglichen mit den HHeCNe-Modellen, die die Fehlergrenzen repräsentieren ( $T_{\text{eff}} = 75\,000\text{ K}$  und  $\log g = 8.0$  rot und  $T_{\text{eff}} = 95\,000\text{ K}$  und  $\log g = 7.0$  blau), und dem HHeCNeFSiPSArCaFeNi-Modell ohne Häufigkeitsvariation (schwarz) von PG 1707+427. Die Fehlergrenzen der Sauerstoffhäufigkeiten sind gestrichelt dargestellt.

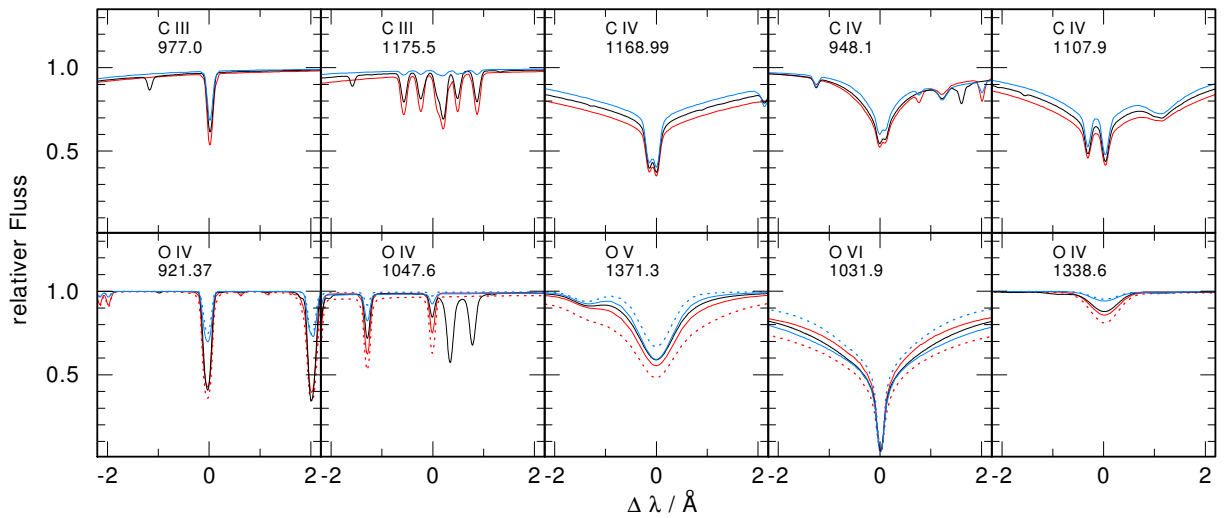


Abbildung 4.33.: Wie Abb. 4.32, für PG 1424+535 ( $T_{\text{eff}} = 110\,000\text{ K}$  und  $\log g = 7.6$ , rot und  $T_{\text{eff}} = 120\,000\text{ K}$  und  $\log g = 7.0$ , blau).

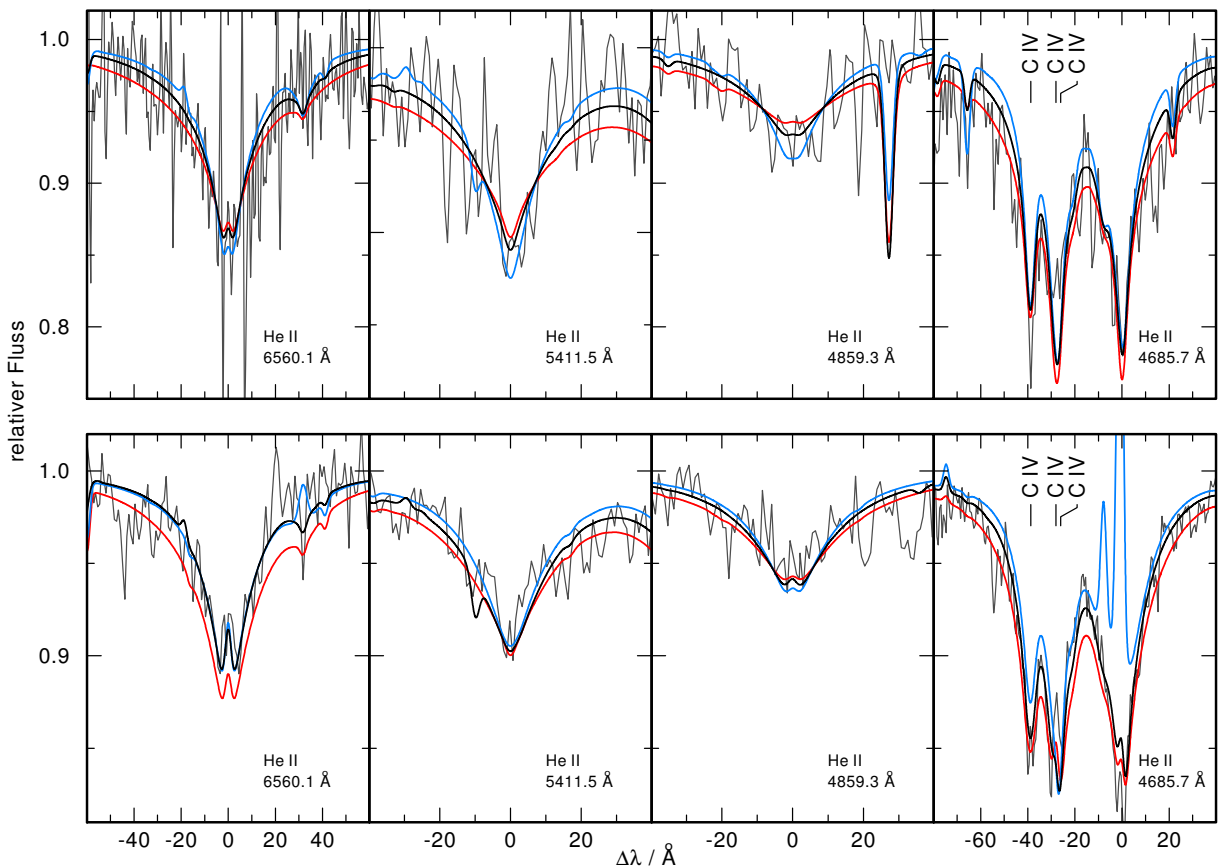


Abbildung 4.34.: Die Linienprofile der optischen Heliumlinien von PG 1424+535 (unten) und PG 1707+427 (oben) verglichen mit den HHeCNONE-Grenzmodellen. Schwarz ist das HHeCNONEFNeSiPSArCaFeNi-Modell vor der Häufigkeitsvariation mit den zentralen  $T_{\text{eff}}$ - und  $\log g$ -Werten dargestellt (115 000 K, 7.3 bzw. 85 000 K, 7.5). Die HHeCNONE-Grenzmodelle mit 120 000 K, 7.0 bzw. 95 000 K, 7.0 sind blau eingezeichnet, die mit 110 000 K, 7.6 bzw. 75 000 K, 8.0 rot.

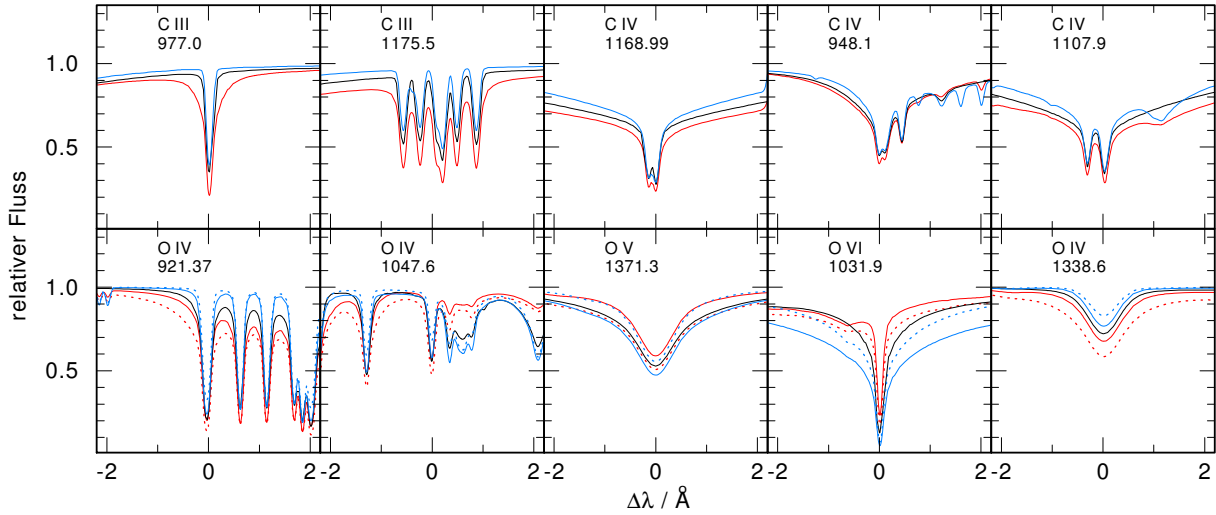


Abbildung 4.35.: Wie Abb. 4.32, für das *TMAW*- anstelle des *HHeCNOFNeSiPSArCaFeNi*-Modells.

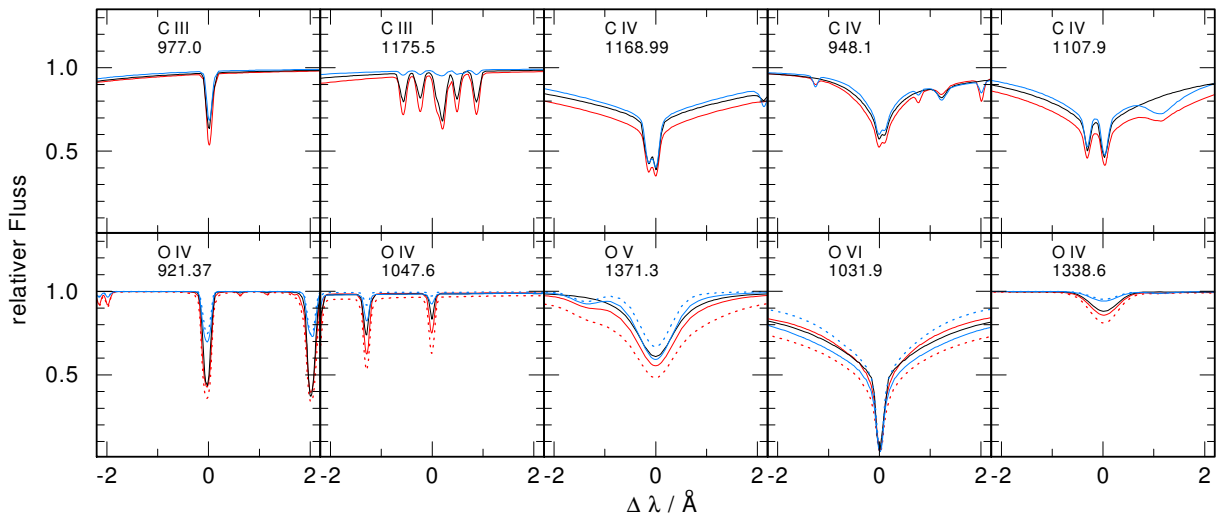


Abbildung 4.36.: Wie Abb. 4.33, für das *TMAW*- anstelle des *HHeCNOFNeSiPSArCaFeNi*-Modells.

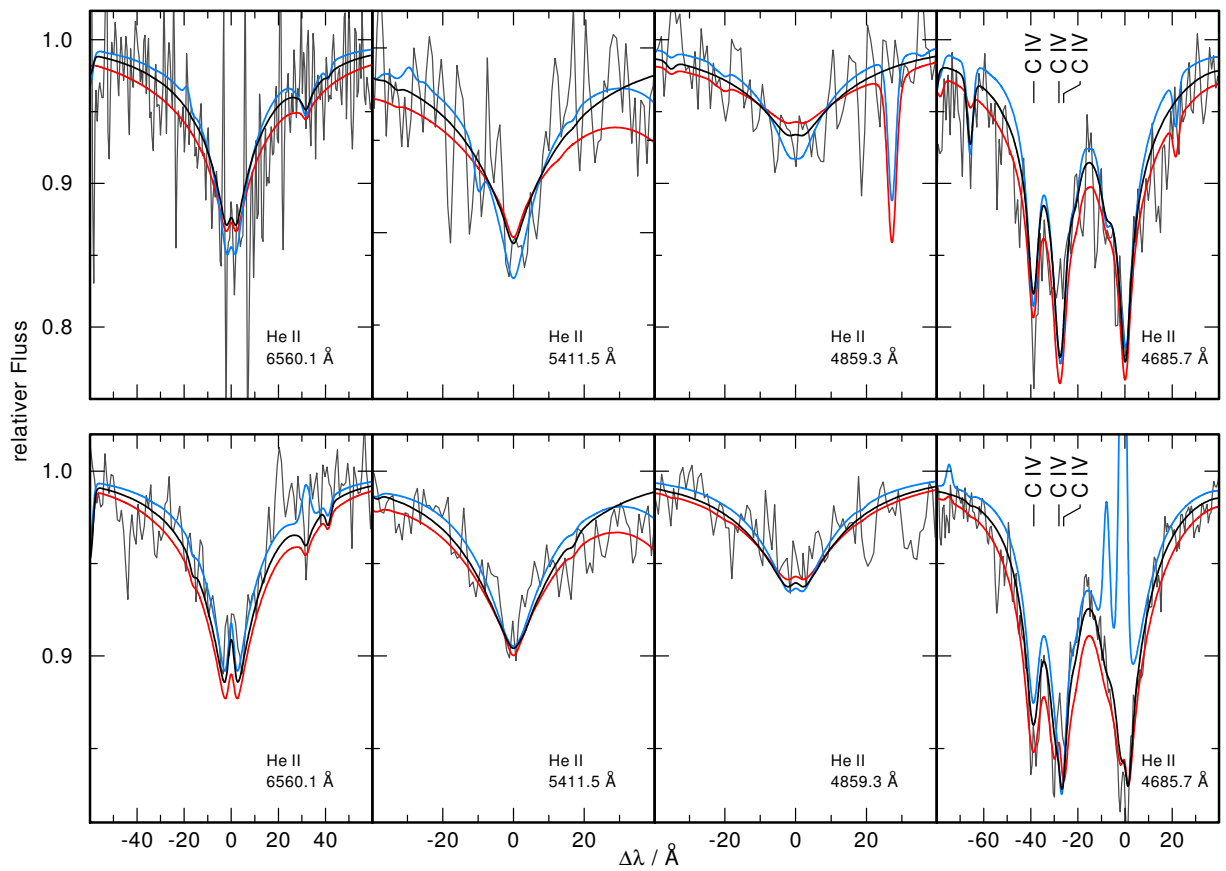


Abbildung 4.37.: Wie Abb. 4.34, für die *TMAW*- anstelle der *HHeCNOFNeSiPSArCaFeNi*-Modelle.

### 4.3. sd(O)B-Sterne mit extrem hohen Eisengruppenhäufigkeiten

Edelmann (2003) analysierte optische Spektren von 146 sdB- und sdOB-Sternen. Insgesamt zeigte diese Analyse, dass die untersuchten Objekte ähnliche Elementhäufigkeiten aufweisen. Drei der Objekte fielen durch ungewöhnliche Häufigkeiten auf. Dies sind die sdOB-Sterne PG 0909+276 ( $T_{\text{eff}} = 36\,900\text{ K}$ ,  $\log g = 5.9$ ) und UVO 0512-08 ( $T_{\text{eff}} = 38\,800\text{ K}$ ,  $\log g = 5.5$ ) und der sdB-Stern UVO 1758+36 ( $T_{\text{eff}} = 32\,000\text{ K}$ ,  $\log g = 5.8$ ). Ihre Spektren zeigen sehr viele, bis dahin nicht beobachtete Linien im optischen Wellenlängenbereich. Viele dieser Linien konnten den Eisengruppenelementen zugeordnet werden, allerdings blieben ca. 23% unidentifiziert. Die anhand dieser Linien bestimmten Häufigkeiten der Eisengruppenelemente waren ungewöhnlich hoch (bis zu 32 000 fach solar). Im Gegensatz zu den anderen Objekten wiesen diese drei Sterne allerdings keine Linien von Fe selbst auf. Später konnte ein weiteres Objekt zu dieser Liste der Sterne mit ungewöhnlich hohen Eisengruppenhäufigkeiten hinzugefügt werden: der sdOB-Stern EC 11481-2303 ( $T_{\text{eff}} = 55\,000\text{ K}$ ,  $\log g = 5.8$ , Rauch et al., 2010) zeigt ein ungewöhnlich flaches UV-Kontinuum, für dessen Verlauf hohe Eisengruppenhäufigkeiten verantwortlich sein könnten (Rauch et al., 2010). Mittlerweile zeigten Geier et al. (2010) durch die Analyse von 139 sdB-Sternen, dass es einen Trend steigender Häufigkeiten der meisten schweren Elemente mit der Temperatur gibt. Eisen selbst macht diesen Trend allerdings nicht mit, sondern ist konstant ca. solar. Die drei von Edelmann (2003) analysierten Objekte wurden von Geier et al. (2010) ebenfalls betrachtet und die extrem supersolaren Eisengruppenhäufigkeiten fallen durch den neu entdeckten Trend nicht mehr (so sehr) aus dem Rahmen. Sie treten aber weiterhin durch eine (super-)solare C-Häufigkeit hervor, was außer bei diesen Objekten nur bei einem weiteren beobachtet wurde. Außerdem zeigen sie mehr He und N als die anderen sdB-Sterne (Edelmann et al., 2001; Edelmann, 2003). Als Erklärungen für diesen Sachverhalt werden von Geier et al. (2010) schwache Konvektion in der Photosphäre, verursacht durch ein Late-Hot-Flasher-Szenario oder einen Merger aus zwei Helium-Weißen Zwergen (wogegen eventuell die niedrige Rotationsgeschwindigkeit spricht) oder eine Entwicklung von He-sdOs zu diesen C-reichen sdBs vorgeschlagen.

Für drei der vier oben erwähnten sdB- und sdOB-Sterne mit extrem hohen Eisengruppenhäufigkeiten sind UV-Spektren vorhanden. Für zwei Objekte zeigen diese einen ungewöhnlich flachen Flußverlauf im IUE-SW-Bereich zwischen 1150 und 1900 Å, der sich mit sinkender Temperatur an den nicht abgeflachten Verlauf anzunähern scheint (Abb. 4.38). Die stärkste Abflachung des Verlaufs scheint sich außerdem mit sinkender Temperatur zu höheren Wellenlängen zu verschieben. Es liegt die Vermutung nahe, dass der flache Flussverlauf mit den ungewöhnlich hohen Eisengruppenhäufigkeiten zusammenhängt und dass dieser Effekt mit sinkender Temperatur abnimmt, was in Übereinstimmung mit den Ergebnissen von Geier et al. (2010) wäre.

Diese Objekte wurden als Testobjekte ausgewählt, um den Einfluss der Eisengruppenelemente auf den UV-Flussverlauf bei sd(O)B-Sternen zu untersuchen. Im Gegensatz zur vorherigen Analyse von LSE 153, LSE 263, PG 1424+535 und PG 1707+427 wird deshalb nur *IrOnIc* und nicht *TMAW* angewandt. Die vorliegende Betrachtung hat nicht den Anspruch genaue Häufigkeiten zu bestimmen, sondern soll lediglich zeigen, ob diese Absenkung des Flussverlaufs mit den Eisengruppenelementen erklärt werden kann und welche Größenordnung ihre Häufigkeiten haben müssen, um die beobachtete Absenkung des UV-Flusses zu verursachen.

Mit Modellen, die Wasserstoff und die Eisengruppenelemente sowie die Literaturwerte für  $T_{\text{eff}}$  und  $\log g$  berücksichtigen wurden für diesen Test verschiedene Häufigkeitsverhältnisse der Eisengruppenelemente simuliert. Da der sdOB-Stern EC 11481-2303 die höchste Temperatur und den am stärksten abgesenkten Flussverlauf zeigt, wurden für dieses Objekt mehrere mögliche Ursachen für den flachen Flussverlauf untersucht (Kapitel 4.3.1). Die am besten zur Reproduktion des Flusses geeignete Ursache wird dann verwendet, um auch den Einfluss auf den Flussverlauf der drei weiteren Objekte abzuschätzen (Kapitel 4.3.2).



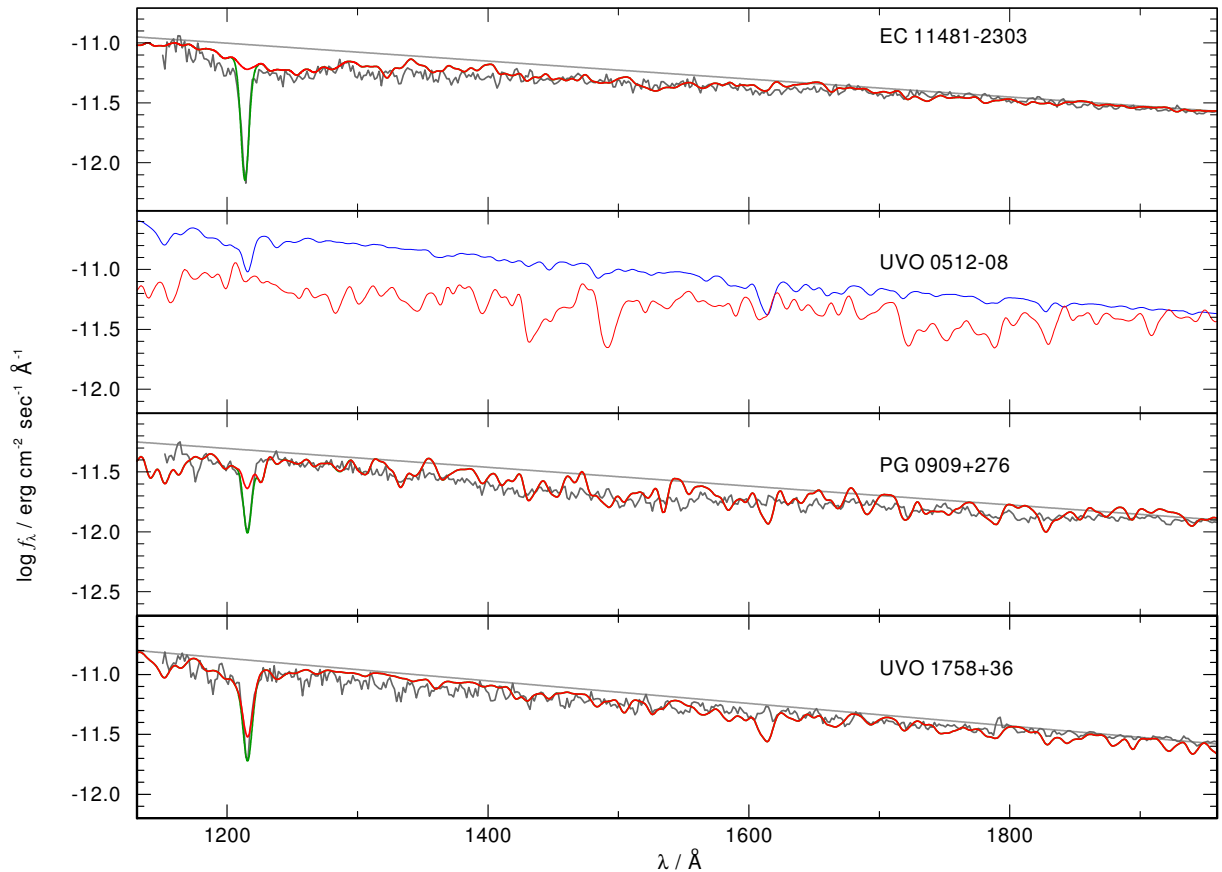


Abbildung 4.38.: UV-Flussverlauf der in diesem Kapitel beschriebenen sd(O)B-Sterne. Rot eingezeichnet ist das Modell, mit dem die Beobachtung am besten wiedergegeben werden kann (siehe Kapitel 4.3.1 und 4.3.2), grün die interstellare Lyman-Linie. Für UVO 0512–08 ist keine Beobachtung vorhanden, deshalb ist das Modell mit Häufigkeiten von Edelmann (2003, rot) und solaren Werten (blau) dargestellt. Die Modelle wurden jeweils auf den rotesten Helligkeitswert (siehe Text) normiert und mit dem jeweils bestimmten  $E_{B-V}$ -Wert gerötet. Um die Abflachung des Flusses besser erkennen zu können wurde zusätzlich eine Gerade eingezeichnet (grau).

### 4.3.1. Der sdOB-Stern EC 11481–2303

#### Entdeckung, frühere Analysen und eigene Vorgehensweise

Der im Edinburgh-Cape Blue Objects Survey von Kilkenny et al. (1997) entdeckte Stern EC 11481–2303 wurde zuerst als DA-Weißer Zwerg klassifiziert. Nach einer ersten Analyse mit LTE-HHe-Modellen von Stys et al. (2000) wurde er neu klassifiziert und gilt seitdem als sdOB-Stern. Die resultierenden Parameter der Analyse sind  $T_{\text{eff}} = 41\,790\text{ K}$ ,  $\log g = 5.84$  und  $\text{He}/\text{H}=0.014$  in Teilchenzahlbruchteilen. Stys et al. (2000) war es allerdings nicht möglich mit ihren Modellen den UV-Flussverlauf zu reproduzieren. Obwohl sie mehrere Szenarien – homogene und geschichtete Atmosphären, einen He-Fleck auf der Oberfläche und eine Mischung aus einem DB- und DA-Weißer Zwerg-Spektrum – aus einer ähnlichen Analyse (Koester et al., 1994 für den DAB-Weißer Zwerg GD 323) in Betracht zogen, blieb diese Diskrepanz ungelöst.

Im Rahmen des ESO Supernovae Type Ia Progenitor Survey (SPY, Napiwotzki et al., 2003) wurde ein optisches Spektrum von EC 11481–2303 aufgenommen. Stroerer et al. (2007) schlossen dieses Spektrum aus ihrer Analyse der sdO-Sterne dieses Surveys aus, vermuteten für EC 11481–2303 aber eine höhere

Tabelle 4.6.: Scheinbare Helligkeiten von EC 11481–2303.

B	V	J	H	K
11.57	11.79	12.512	12.669	12.765

als die bisher bekannte  $T_{\text{eff}}$ .

Rauch et al. (2010) untersuchten dieses SPY-Spektrum zusammen mit den vorhandenen IUE- und FUSE-Beobachtungen. Für diese NLTE-Spektralanalyse wurde der *TMAP*-Code verwendet und die Elemente H, He, C, N, O und Ca-Ni (als generisches Modellatom) berücksichtigt. Das Ergebnis der Analyse des optischen SPY-Spektrums war  $T_{\text{eff}} = 55\,000$  K,  $\log g = 5.8$ ,  $\text{He}/\text{H}=0.0025$  (Teilchenzahlbruchteil) sowie Obergrenzen  $[\text{C}] = -0.875$ ,  $[\text{N}] = -0.875$  und  $[\text{O}] = -0.875$ . Es wurde abgeschätzt, dass die Eisengruppenelementhäufigkeit mindestens zehnfach solar und die Nickelhäufigkeit sogar noch höher sein muss, um den Flussverlauf im UV zu repräsentieren. Für weitere Untersuchungen wurde vorgeschlagen, Diffusionsrechnungen durchzuführen und die Eisengruppenelemente individuell (anstelle des generischen Modellatoms) zu berücksichtigen. Außerdem wurden in dieser Analyse die interstellare Extinktion und die interstellaren Linien nicht modelliert, was ebenfalls als mögliche Erklärung für den UV-Flussverlauf bzw. als Hilfe zur Bestimmung nicht-identifizierter Absorber vorgeschlagen wurde.

Diese Vorschläge werden im Rahmen der vorliegenden Arbeit untersucht. Zu Beginn wurde dafür ein HHeCNO-Modell mit den Parametern von Rauch et al. (2010) berechnet. Mit Hilfe der FUSE- und IUE-Daten, sowie optischen und infraroten Helligkeiten (SIMBAD<sup>9</sup>, Tabelle 4.6) wurde dann die interstellare Extinktion unter Zuhilfenahme des Gesetzes von Fitzpatrick (1999) bestimmt. Der Fluss wurde dafür auf die  $m_K$  Helligkeit normiert. Das beste Ergebnis wurde mit  $E_{B-V} = 0.02 \pm 0.01$  erzielt, was deutlich niedriger als der Literaturwert (0.06, Stys et al., 2000) und der bei GALEX<sup>10</sup> angegebene Wert (0.06) ist.

Für die weitere Analyse wurden Modelle mit den von Rauch et al. (2010) bestimmten  $T_{\text{eff}}$ - und  $\log g$ -Werten berechnet. Diese Modelle berücksichtigen Wasserstoff und alle Eisengruppenelemente individuell mit den Ionen IV-VII bzw. IV-VIII (für Mn, Fe, Co und Ni). Aufgrund numerischer Instabilität wurde V V nur ohne gebunden-gebunden Übergänge berücksichtigt. Als Quelle für die Eisengruppenlinien wurden die LIN-Listen von Kurucz verwendet. Das Grundmodell enthält solare Häufigkeiten für alle Elemente. Ausgehend von diesem wurde die Häufigkeit jeweils eines Eisengruppenelements um bis zu einem Faktor 1000 erhöht, während die anderen Elemente weiterhin mit dem solaren Wert berücksichtigt wurden. Der Flussverlauf dieser Modelle wurde im IUE- und FUSE-Wellenlängenbereich verglichen.

Abb. 4.39 zeigt, dass die Absenkung des Flussverlaufs abhängig vom Element in unterschiedlichen Wellenlängenbereichen verschieden stark ist. An diese Abbildung, sowie auch an alle anderen dieses Kapitels (abgesehen von Abb. 4.38), wurde keine Korrektur der interstellaren Wasserstofflinien angebracht. Die Lyman-Linien werden deshalb nicht reproduziert. Die Abbildung zeigt, dass Eisenlinien den Fluss vor allem im Bereich um ca. 1000 Å und ab ca. 1350 – 1600 Å absenken. Nickellinien sorgen zwischen ca. 900 und 1500 Å für eine Flussreduktion. Eine Erhöhung aller anderen Eisengruppenelemente hat entweder keine signifikanten Auswirkungen auf den Flussverlauf zur Folge oder in Bereichen, in denen die Beobachtung keine Einsenkung zeigt. Das Ergebnis dieser Untersuchung ist also, dass nur Eisen und Nickel für den beobachteten Verlauf verantwortlich sein können.

Für eine genauere Untersuchung wurden anschließend Modelle mit solaren Eisengruppenhäufigkeiten und verschiedenen Häufigkeitskombinationen von Eisen und Nickel erzeugt. Das Ergebnis ist, dass der UV-Flussverlauf am besten von einem Modell reproduziert wird, das eine 1000-fach solare Nickel- und eine 10–100-fach solare Eisenhäufigkeit hat (Abb. 4.39). Werden in diesem Modell die Häufigkeiten der anderen Eisengruppenelemente erhöht, reproduziert das Modell die Beobachtung nicht besser. Stark erhöhte Eisen- und Nickelhäufigkeiten sind also hauptsächlich für diesen UV-Flussverlauf verantwortlich.

<sup>9</sup><http://simbad.u-strasbg.fr>

<sup>10</sup><http://www.galex.caltech.edu/>

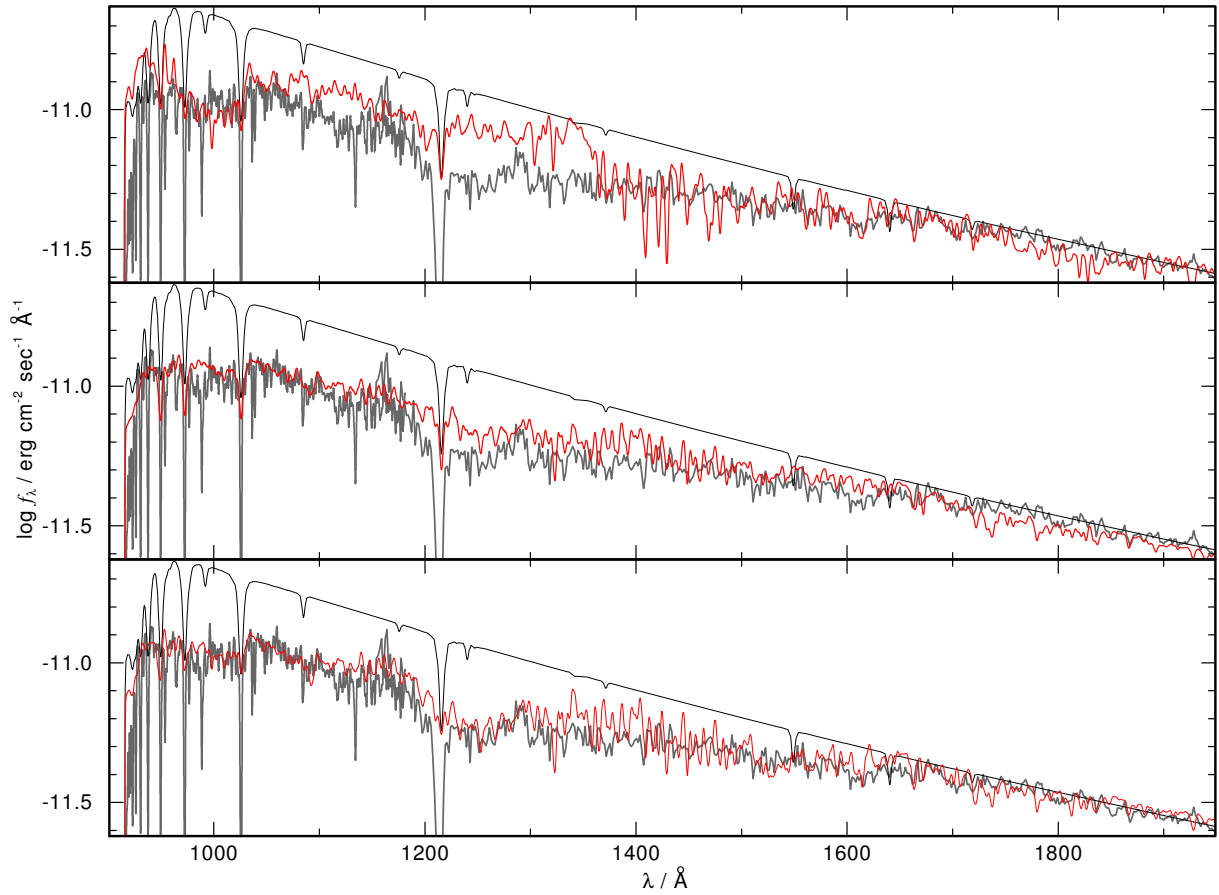


Abbildung 4.39.: UV-Flussverlauf von EC 11481–2303. Schwarz dargestellt ist das HHeCNO-Modell, grau die Beobachtung. Oben: das Wasserstoff-Eisengruppenelemente-Modell mit 100-fach solaren Häufigkeiten für Eisen, solare für die anderen Eisengruppenelemente. Mitte: 1000-fach solare Nickel-Häufigkeiten, solare für die anderen Eisengruppenelemente. Unten: die finale Parameterkombination (1000-fach solare Nickel-, zehnfach solare Eisenhäufigkeit). Es wurde eine interstellare Extinktion mit  $E_{B-V} = 0.01$  (HHeCNO- und Wasserstoff-Eisengruppenelemente-Modell mit 100-facher Eisenhäufigkeit) bzw. 0.03 (mittleres und unteres Wasserstoff-Eisengruppenelemente-Modell) angebracht. Außerdem wurden die Beobachtungen mit einem Gaussprofil der Halbwertsbreite  $1 \text{ \AA}$  gefaltet, die Modelle mit  $3 \text{ \AA}$  und auf  $m_K$  normiert.)

### NGRT – chemisch geschichtete Modelle

Zur Erklärung der beobachteten Elementhäufigkeiten wurde bisher erfolgreich Diffusion (eventuell in Kombination mit einem schwachen Wind) herangezogen (z.B. Geier et al., 2010). Deshalb wird angenommen, dass Diffusion in den Atmosphären der sd(O)B-Sterne eine große Rolle spielt. Die aus diesen Simulationen resultierenden geschichteten Atmosphären sollen im nächsten Schritt untersucht werden, um zu zeigen, ob eine Häufigkeit in der eben abgeschätzten Größenordnung durch Diffusion erklärt werden kann und um zu überprüfen, ob eine geschichtete Atmosphäre den Flussverlauf noch besser reproduziert. Da *TMAP* eine homogene Atmosphäre annimmt, wurde der Next Generation Radiative Transport-Code (*NGRT*, Dreizler, 1999) verwendet, um geschichtete Modelle für diese Analyse zu erzeugen. Dieses Programm basiert auf der Annahme eines Gleichgewichts zwischen gravitativem Absinken und strahlungsgetriebenem Auftrieb der Elemente. Die Theorie dazu basiert auf Chayer et al. (1995) und wurde von Dreizler (1999) im *NGRT*-Programm modifiziert. Für die Berechnung der Atmosphäre

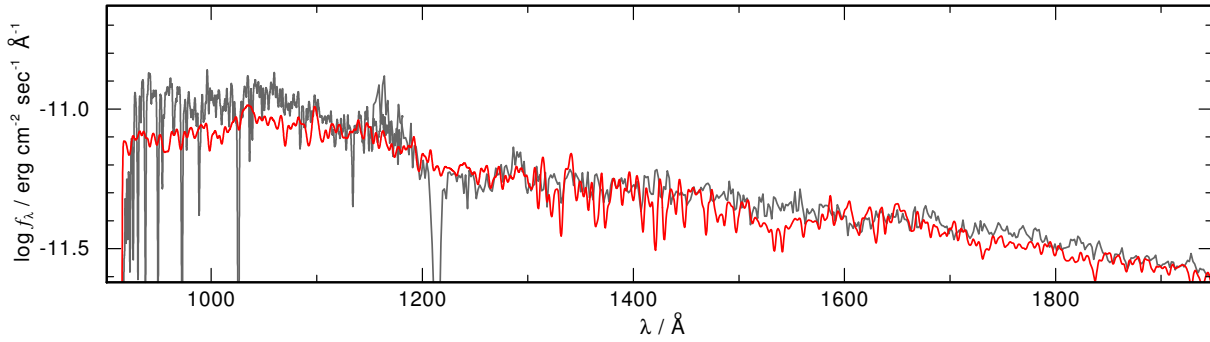


Abbildung 4.40.: Wie Abb. 4.39, mit einem Diffusionsmodell (rot,  $E_{B-V} = 0.03$ ).

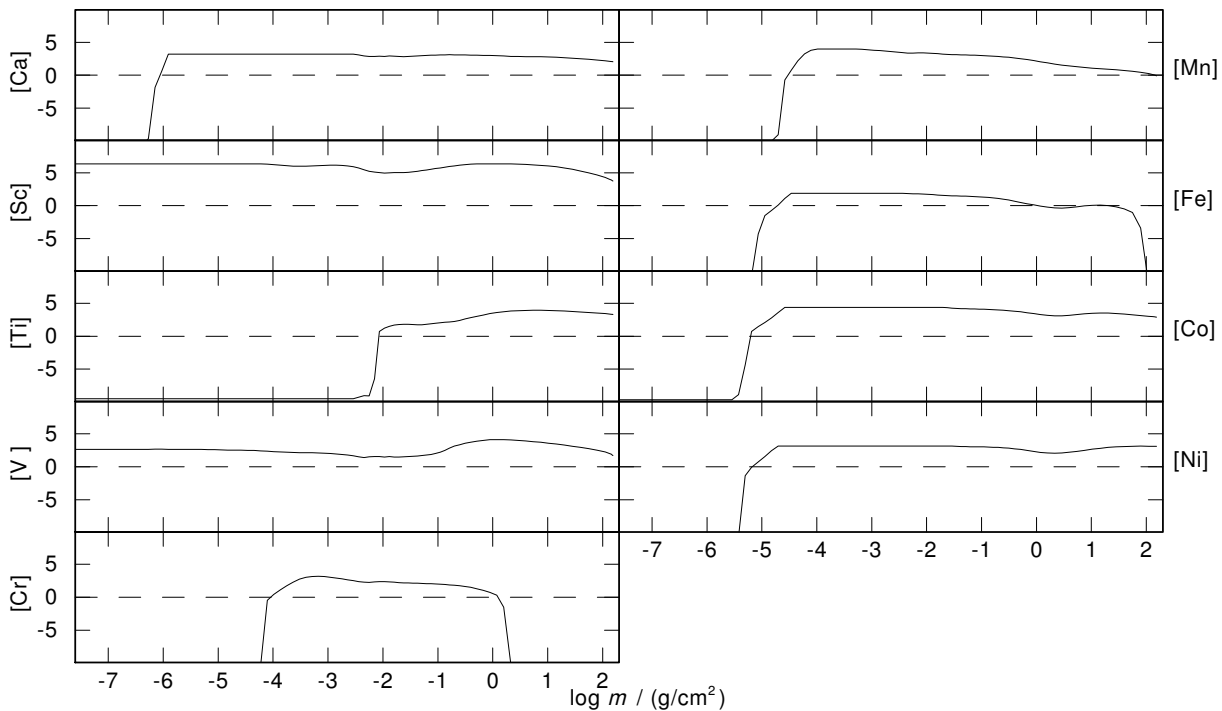


Abbildung 4.41.: Elementhäufigkeiten der Eisengruppenelemente in der Atmosphäre der *NGRT*-Diffusionsmodell von EC 11481–2303.

mit dem *NGRT*-Code dient ein *TMAP*-Modell als Startmodell. In jedem Iterationszyklus der Berechnung wird dann zuerst die radiative Beschleunigung bei festgehaltener Atmosphärenstruktur ermittelt und anschließend das Strahlungsfeld und die Atmosphärenstruktur neu berechnet. Freie Parameter bei dieser Art der Berechnung sind nur  $T_{\text{eff}}$  und  $\log g$ . Die Elementhäufigkeiten stellen sich im Lauf der Modellierung in jedem Tiefenpunkt ein. Die *NGRT*-Berechnung führt somit zu einer Schichtung der Elemente wie sie in Abb. 4.41 dargestellt ist.

Für diesen Test wurde das *TMAP*-Modell, das Wasserstoff und die Eisengruppenelemente mit solaren Häufigkeiten berücksichtigt, als Startmodell für das *NGRT*-Programm verwendet. Das resultierende *NGRT*-Modell reproduziert den Flussverlauf über weite Teile des Wellenlängenbereichs (Abb. 4.40), in einigen Abschnitten (z.B. kurzweilig von  $1100 \text{ \AA}$ ) ist der Flussverlauf allerdings zu flach. Insgesamt bestätigt dieser Test mit der geschichteten Atmosphäre, dass der mit *TMAP* bestimmte Trend der (extrem) supersolaren Häufigkeiten für Eisen und Nickel richtig ist. Die geschichtete Atmosphäre repräsentiert den Flussverlauf allerdings nicht besser als die homogene.

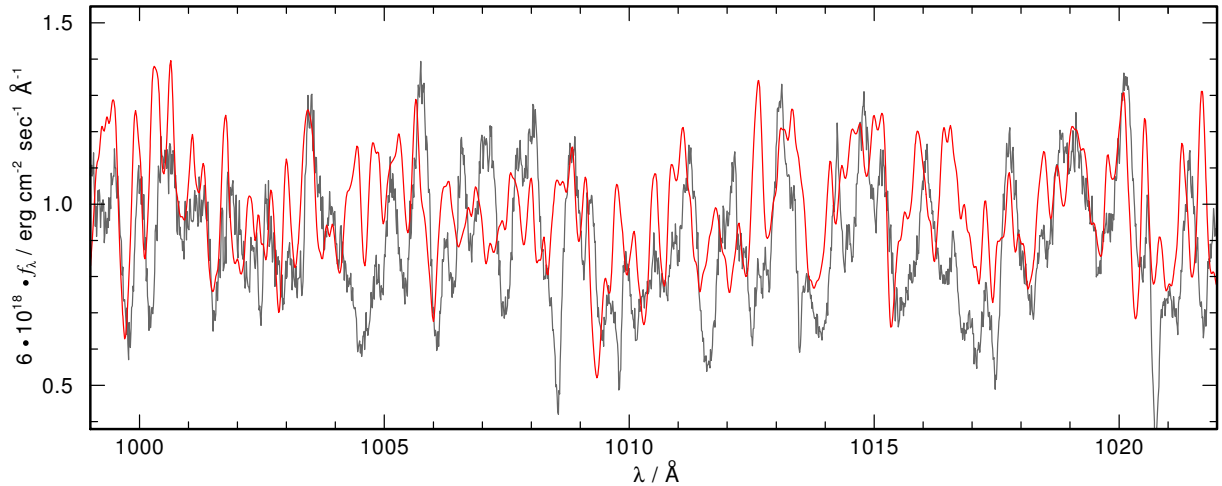


Abbildung 4.42.: Ausschnitt aus der FUSE-Beobachtung von EC 11481–2303 verglichen mit dem *TMAP*-Modell (rot,  $v_{\text{rot}} = 30 \text{ km/sec}$ ,  $E_{B-V} = 0.02$ ).

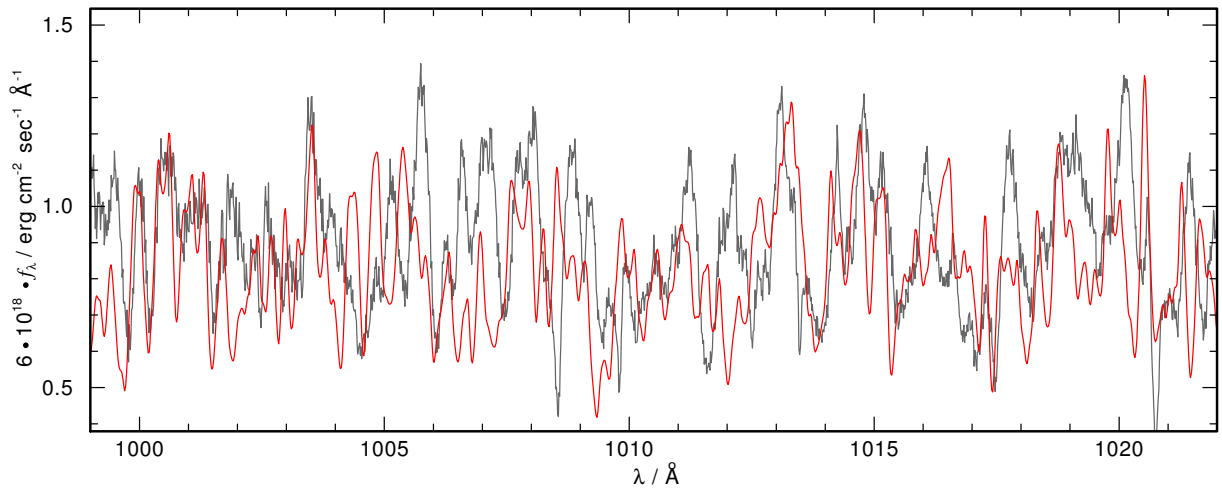


Abbildung 4.43.: Wie Abb. 4.42, mit dem *NGRT*-Modell (rot).

### Modellierung der Linienprofile

In früheren Analysen wurde bereits festgestellt, dass in den FUSE- und IUE-Spektren viele bisher unidentifizierte, breite und tiefe Linien zu finden sind, die nicht von den leichten und mittelschweren Metallen stammen. Es wird vermutet, dass diese Linien von vielen feinen, überlagerten Eisengruppenlinien hervorgerufen werden. Um das zu überprüfen, wurden das finale *TMAP*-Modell (mit 1000-fach solarer Nickel- und zehnfach solarer Eisenhäufigkeit) und das *NGRT*-Diffusionsmodell mit der FUSE-Beobachtung verglichen. Obwohl einzelne Linien bzw. Wellenlängenbereiche (beispielsweise  $999 - 1022 \text{ \AA}$ ) des *TMAP*-Modells gut mit der Beobachtung übereinstimmen, scheinen Modell und Beobachtung ansonsten nicht zueinander zu passen (Abb. 4.42 und 4.45). Das *NGRT*-Modell stimmt insgesamt noch weniger mit der Beobachtung überein (Abb. 4.43).

Als Erklärung für die beobachteten, bisher nicht identifizierten Linien scheiden die Eisengruppenelemente jedoch nicht aus. Bei einem Vergleich des theoretischen und beobachteten Spektrums fällt auf, dass die theoretischen Linien schmäler als die beobachteten sind, obwohl eine Faltung mit einem Gaussprofil der Halbwertsbreite  $0.06 \text{ \AA}$  zur Simulation der Instrumentenauflösung bereits angebracht wurde. Wird die Halbwertsbreite des Gaussprofils erhöht bzw. ein zweites Profil angebracht, so überlagern sich mehrere schmale Linien und werden scheinbar zu einer breiten Linie. Dieses Verhalten könnte ein Hin-

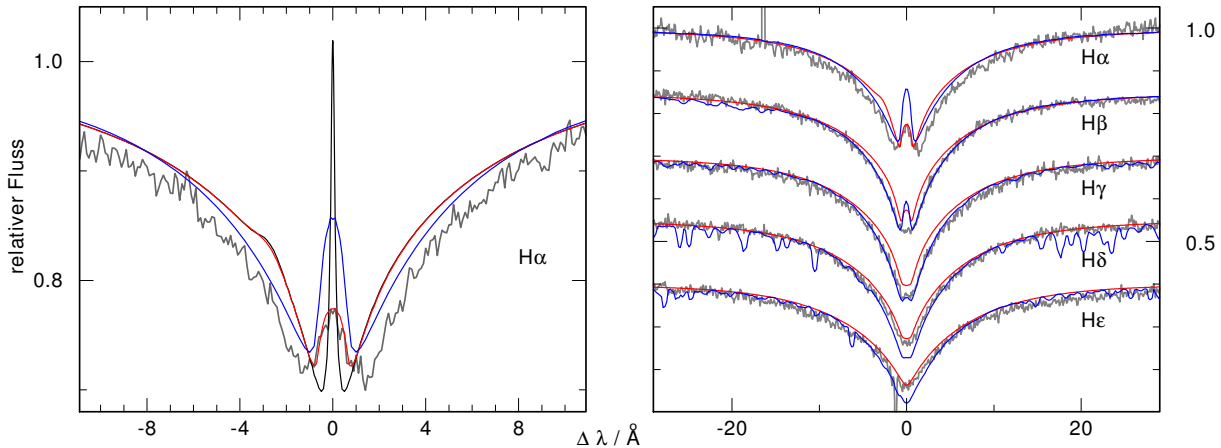


Abbildung 4.44.: Rechts: Die Balmerlinien von EC 11481–2303 verglichen mit dem Modell von Rauch et al. (2010, rot, Rotationsgeschwindigkeit 30 km/sec) und mit dem finalen Modell dieser Analyse ( $v_{\text{rot}} = 30\text{ km/sec}$ , blau). Links: Das Modell von Rauch et al. (2010) mit (rot) und ohne Rotationsverbreiterung (schwarz) und das finale *TMAP*-Modell dieser Analyse ( $v_{\text{rot}} = 30\text{ km/sec}$ , blau). Die Modelle wurden mit einem Gaussprofil der Halbwertsbreite  $0.1 \text{ \AA}$  gefaltet.

weis darauf sein, dass das Objekt rotiert. Die resultierende Rotationsverbreiterung wird durch eine Faltung mit einem Rotationsprofil simuliert. Da in den bisherigen Analysen allerdings keine Hinweise auf eine Rotation des Sterns gefunden wurde, wurden die optischen Linien der Rauch et al. (2010)-Analyse nochmals untersucht. Dabei zeigt sich, dass die beobachtete  $H\alpha$ -Linie einen deutlich schwächeren Emissionskern als das Modell von Rauch et al. (2010) hat. Beim Anbringen einer Rotationsgeschwindigkeit wird der Emissionskern des Modells schwächer je höher die betrachtete Rotationsgeschwindigkeit ist. Auf das Modell von Rauch et al. (2010) wurden deshalb verschiedene Rotationsverbreiterungen angewandt. Das beste Ergebnis wird mit einer Rotationsgeschwindigkeit von  $v_{\text{rot}} = 30 \text{ km/sec}$  erzielt (Abb. 4.44). Wird diese Rotationsgeschwindigkeit an das *TMAP*-Modell mit Wasserstoff und Eisengruppenelementen angebracht, so können im FUSE-Bereich mehr Kennzeichen des Spektrums reproduziert werden (Abb. 4.42). Ein Vergleich mit den Balmerlinien zeigt außerdem, dass das finale Modell dieser Analyse mit den beobachteten Linien konsistent ist, die Temperatur allerdings weiter angepasst bzw. weitere Elemente zur Modellierung der Linienprofile berücksichtigt werden müssen. Für diesen Vergleich des finalen Modells wurde das Wasserstoff-Modellatom erweitert, da das ursprünglich verwendete nicht alle gezeigten Linien beinhaltet.

Insgesamt zeigt diese Analyse, dass EC 11481–2303 eine ungewöhnlich hohe Eisen- und Nickelhäufigkeit aufweist. Der Flussverlauf kann mit diesen hohen Häufigkeiten reproduziert und erklärt werden. Obwohl in dieser Analyse keine Linie der FUSE-Beobachtung eindeutig identifiziert werden kann, sind die Eisengruppenelemente sehr wahrscheinlich die Verursacher dieser Linien. Der Grund dafür sind die verwendeten Atomdaten. Quelle für die Atomdaten der Eisengruppenelemente sind die LIN-Listen von Kurucz (Kapitel 2.2). Diese enthalten, wie bereits erwähnt, hauptsächlich theoretisch berechnete Wellenlängen und Oszillatorenstärken, deren Werte mit einer hohen Unsicherheit behaftet sind ( $\approx 10\%$  für die Wellenlängen,  $\approx 15\%$  bei den Oszillatorenstärken). Würden anstelle der berechneten LIN-Linien die experimentell bestimmten POS-Linien verwendet werden, so würden – vor allem bei diesen hohen Ionisationsstufen – sehr viele (das Verhältnis von LIN- zu POS-Linien ist ca. 300, Tabelle 2.1) Opazitätsquellen fehlen, was einen unrealistischen Flussverlauf verursacht.

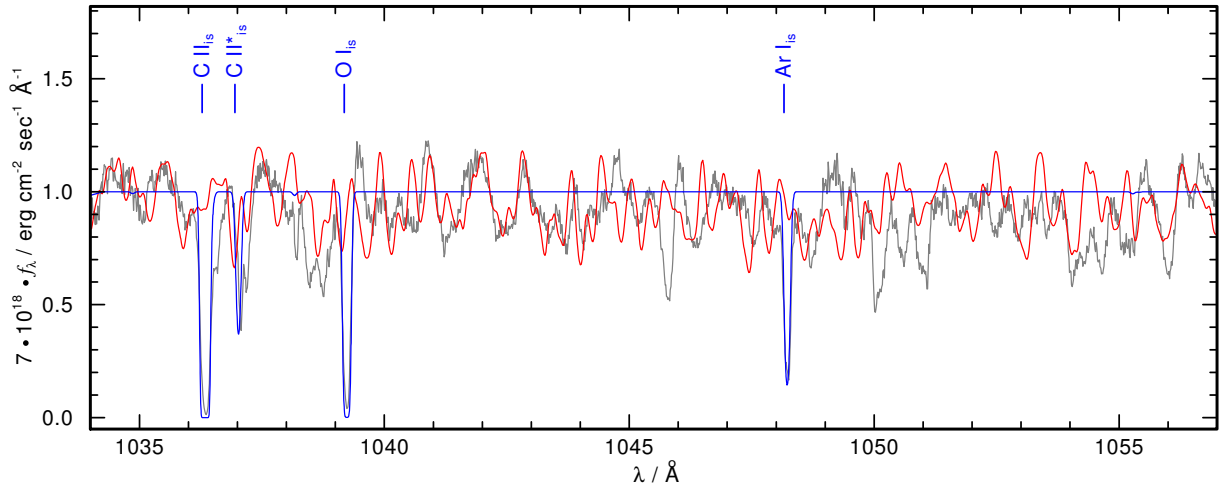


Abbildung 4.45.: Wie Abb. 4.42, für einen anderen Ausschnitt des FUSE-Wellenlängenbereichs. Zusätzlich sind die interstellaren Linien (*OWENS*) dargestellt.

### **OWENS – Die Interstellaren Linien**

Eine weitere Möglichkeit, die bisher unbekanntes spektrale Kennzeichen der FUSE-Beobachtung zu identifizieren, ist die Identifikation und Modellierung der interstellaren Linien. Diese werden von Elementen zwischen uns und dem Stern verursacht und überlagern das beobachtete Spektrum vor allem im UV-Bereich. Um diese Linien zu modellieren, wird in dieser Arbeit das Programm *OWENS* verwendet. Das Ergebnis der Modellierung von EC 11481–2303 ist, dass einige starke Linien des beobachteten Spektrums tatsächlich als interstellare Linien identifiziert werden können. Es kann aber auch gezeigt werden, dass die Mehrheit der Linien stellaren Ursprungs sein muss (Abb. 4.45), was weitere Hinweise auf die Eisengruppenelemente als Verursacher liefert.

Insgesamt kann also geschlossen werden, dass der ungewöhnlich flache Verlauf des UV-Flusses von EC 11481–2303 von Nickel- und Eisenlinien verursacht wird. Das *NGRT*-Modell, das eine geschichtete Atmosphäre simuliert, bestätigt das Ergebnis der (extrem) supersolaren Häufigkeiten. Wenige Linien des beobachteten Spektrums haben interstellaren Ursprung, die Mehrheit stammt von den Eisengruppenelementen. Dass diese nicht eindeutig zugeordnet werden können liegt vermutlich an den Unsicherheiten der zugrunde liegenden Atomdaten.

### **4.3.2. Die sd(O)B-Sterne UVO 1758+36 und PG 0909+276**

Der ungewöhnliche Flussverlauf von EC 11481–2303 – des heißesten der vier ausgewählten sd(O)B-Sterne – kann durch eine Atmosphäre erzeugt werden, die einen 1000-fach solaren Nickel- und einen zehnfach solaren Eisenwert aufweist. Wie bereits erwähnt, scheint dieser Trend der überhäufigen schweren Metalle mit sinkender Temperatur abzunehmen. Die drei anderen Vertreter dieser Gruppe – UVO 1758+36 ( $T_{\text{eff}} = 32\,000\text{ K}$ ,  $\log g = 5.8$ ), PG 0909+276 (36 900 K, 5.9) und UVO 0512–08 (38 800 K, 5.5), Literaturwerte von Edelmann (2003) – haben niedrigere Temperaturen. Aufgrund dessen ist zu erwarten, dass der UV-Flussverlauf dieser Objekte in geringerem Maße von den Eisengruppenelementen beeinträchtigt wird, was nun untersucht wird.

Für diese Analyse wurden die verfügbaren UV-Beobachtungen verwendet. Für PG 0909+276 existiert eine IUE-Beobachtung im relevanten Wellenlängenbereich mit niedriger Auflösung (ID: SWP27469) und eine langwellig davon (ID: LWP07464 – niedrige Auflösung). In diesem Bereich wurde ebenfalls für UVO 1758+36 eine Beobachtung mit niedriger (ID: SWP08078), eine mit hoher Auflösung (ID: SWP20668) und zwei langwellig davon (ID: LWR16578 – niedrige Auflösung, ID: LWR16579 – hohe

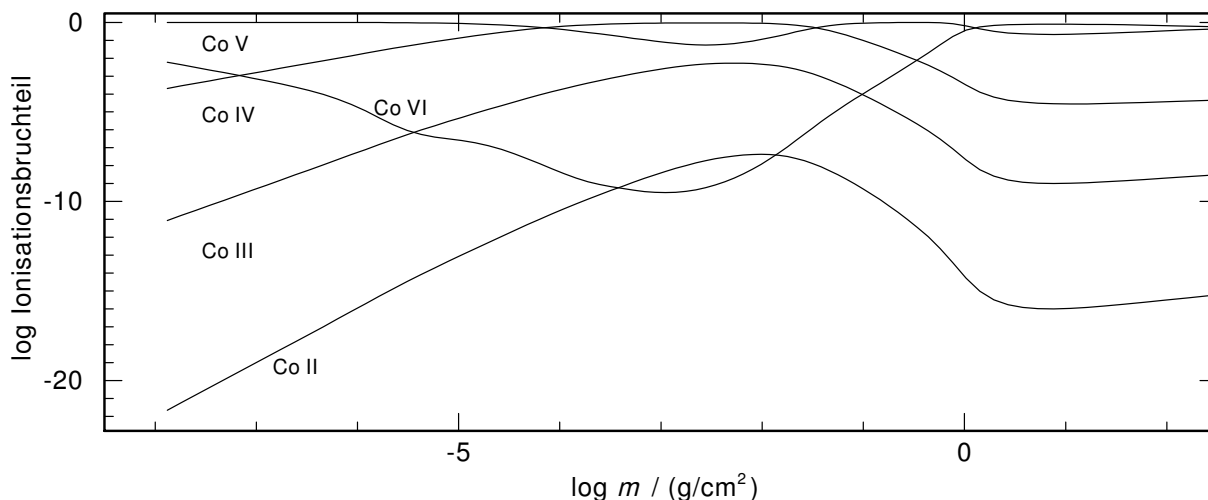


Abbildung 4.46.: Ionisationsbruchteile von Co ( $T_{\text{eff}} = 37\,000\text{ K}$ ,  $\log g = 5.9$ , Häufigkeiten von Edelmann, 2003).

Tabelle 4.7.: Scheinbare Helligkeiten der drei sd(O)B-Sterne (SIMBAD).

	B	V	R	J	H	K
UVO 0512–08	10.97		11.34	11.84	11.97	12.096
PG 0909+276	10.74	12.1				
UVO 1758+36	11.17	11.39	11.38	11.956	12.121	12.226

Auflösung) aufgenommen. Für UVO 0512–08 existiert keine UV-Beobachtung.

Wie bei der Analyse von EC 11481–2303 wurden für jedes Objekt Modelle berechnet, die Wasserstoff und die individuellen Eisengruppenelemente enthalten. Als Startwert für die Häufigkeiten wurden die von Edelmann (2003) bestimmten Werte verwendet oder, falls nicht vorhanden, solare Werte angenommen. Für die Atomdaten der Eisengruppenelemente wurden die Ionisationsstufen II–VI (II und VI je nur mit einem Level) berücksichtigt und ein Frequenzgitter mit ca. 9000 Punkten verwendet. Obwohl sowohl Edelmann (2003) als auch Geier et al. (2010) hauptsächlich Linien der Ionisationsstufen III (und IV) analysiert haben, zeigt Abb. 4.46 am Beispiel von Co, dass die Ionisationsstufen IV und V bei unseren Modellen dominant sind.

Für PG 0909+276 ist der theoretische Flussverlauf mit den Häufigkeiten von Edelmann (2003) bereits deutlich zu flach (Abb. 4.47 und 4.48), einzelne Eisengruppenhäufigkeiten sind also schon zu hoch. Um die Elemente zu identifizieren, die für diesen Verlauf verantwortlich sind, wurde zusätzlich ein Modell mit solaren Eisengruppenhäufigkeiten angesetzt. Bei der Berechnung des synthetischen Spektrums wurden dann testweise einzelne Elementhäufigkeiten (wieder) erhöht.

Für eine realistische Analyse des UV-Bereichs muss die interstellare Extinktion berücksichtigt werden. Durch sie wird der Flussverlauf vor allem im UV-Bereich abgesenkt. Im optischen Wellenlängenbereich sind die Auswirkungen deutlich geringer und im Infraroten vernachlässigbar. Wie bei der Analyse von EC 11481–2303 wurde deshalb die interstellare Extinktion anhand des Extinktionsgesetz von Fitzpatrick (1999) an das (auf die roteste scheinbare Helligkeit (Tabelle 4.7) normierte) Modell angebracht (Abb. 4.47). Der Fehler dieses Werts ist aufgrund der variierten Zusammensetzung der Atmosphäre jedoch groß, was deutliche Auswirkungen auf die bestimmten Häufigkeiten der Eisengruppenelemente hat. Wird z.B. die interstellare Extinktion komplett ignoriert ( $E_{B-V} = 0.0$ ), so müssen die Häufigkeiten um bis zu einem Faktor zehn erhöht werden, um den Verlauf zu reproduzieren.



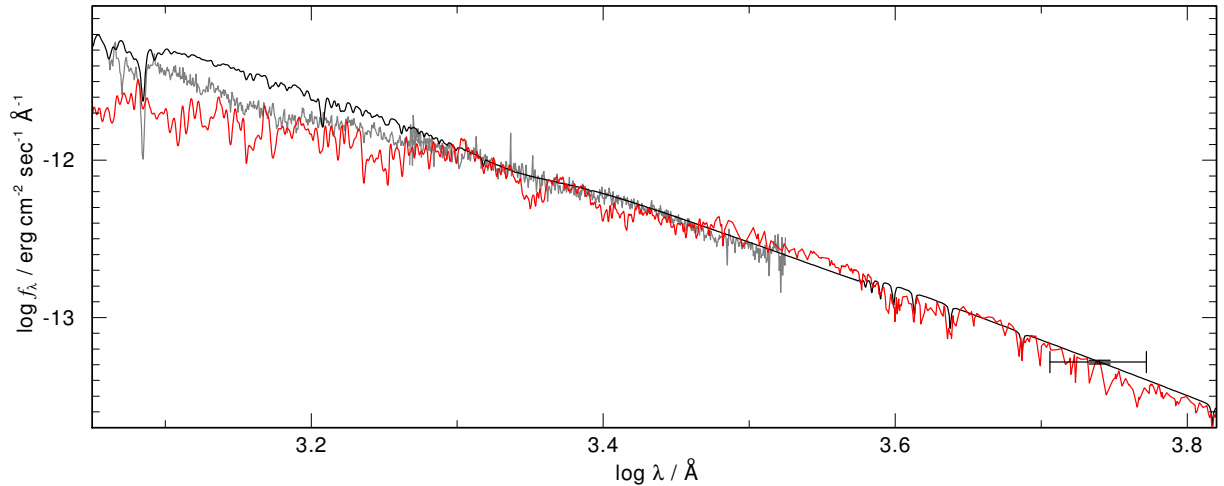


Abbildung 4.47.: Bestimmung der interstellaren Extinktion von PG 0909+276. An das Modell mit Parametern von Edelmann (2003, rot) ist eine interstellare Extinktion mit  $E_{B-V} = 0.06$  angebracht, an das mit solaren Häufigkeiten (schwarz) mit  $E_{B-V} = 0.07$ . Die Modelle sind auf die ebenfalls dargestellte V-Helligkeit normiert.

Abb. 4.48 zeigt die Auswirkungen der Erhöhung der individuellen Elementhäufigkeiten im solaren Modell auf den Flussverlauf. Die einzelnen synthetischen Spektren sind dabei so normiert, dass sie den beobachteten Flusswert bei  $3000 \text{ \AA}$  reproduzieren. Es ist gut zu erkennen, dass nur die Elemente Cr, Mn, Fe, Co und Ni eine merkliche Auswirkung auf den UV-Flussverlauf im relevanten Bereich ( $1130$  bis  $2000 \text{ \AA}$ ) haben. Eine Erhöhung des Cr-Wertes hat die stärksten Auswirkungen im Bereich zwischen  $1300$  und  $1400 \text{ \AA}$ , um  $1600$  und zwischen  $1700$  und  $1850 \text{ \AA}$ . Bei Mn sind diese zwischen  $1250$  und  $1350 \text{ \AA}$ , sowie zwischen  $1550$  und  $1800 \text{ \AA}$ . Eine Erhöhung der Eisenhäufigkeit führt zu stärksten Absenkungen von  $1150$  bis  $1250 \text{ \AA}$  und von  $1500$  bis  $1700 \text{ \AA}$ . Co ist für eine Absenkung um ca.  $1150 \text{ \AA}$ , zwischen  $1350$  und  $1600 \text{ \AA}$  und von  $1800$  bis  $1850 \text{ \AA}$  verantwortlich. Ni senkt den Fluss vor allem zwischen  $1130$  und  $1500 \text{ \AA}$ . Es ist gut zu erkennen, dass der UV-Flussverlauf beinahe von dem Modell mit der  $10\,000$ -fachen solaren Co-Häufigkeit reproduziert wird. Bei ca.  $1150 \text{ \AA}$  sowie zwischen  $2000$  und  $2200 \text{ \AA}$  ist dieser theoretische Flussverlauf beinahe zu flach, bei ca.  $1700 \text{ \AA}$  noch nicht flach genug, was durch eine Erhöhung der Co-, Cr-, Mn- und Ni-Häufigkeiten eventuell verbessert werden kann. Aufgrund dieser Erkenntnisse wurden Modelle mit verschiedenen Häufigkeitskombinationen von Ni, Mn, Co und Cr berechnet. Das Ergebnis dieser Modellierung ist, dass ein Modell mit einer  $1000$ -fachen solaren Kobalt-,  $100$ -fachen solaren Chrom-, zehnfach solarer Mangan- und  $100$ -fachen solarer Nickelhäufigkeit die Beobachtung am besten repräsentiert (Abb. 4.48, Tabelle 4.8, die im EM-Modell mit den Häufigkeiten von Edelmann (2003) abweichenden Werte werden durch die Normierung verursacht). Bei einer weiteren Erhöhung der Häufigkeiten dieser Elemente um einen Faktor zehn wird der Flussverlauf in bestimmten Wellenlängenbereichen flacher als beobachtet. Bei etwa  $1150 \text{ \AA}$  ist der theoretische Flussverlauf etwas zu flach. Ein Reduzieren der Kobalthäufigkeit um einen Faktor zehn hat zur Folge, dass das Modell die Beobachtung in diesem Bereich besser reproduziert, zwischen  $1400$  und  $1600 \text{ \AA}$  aber nicht mehr flach genug ist, weshalb die höhere Häufigkeit gewählt wird.

Die Größe der Variationsschritte der Häufigkeiten (Faktor zehn) macht noch einmal deutlich, dass hier keine genaue Häufigkeitsbestimmung erfolgt, sondern nur eine Aussage gemacht wird, ob der Flussverlauf von erhöhten Eisengruppenelementwerten verursacht werden kann. Die resultierenden Werte sind mit großen Fehlergrenzen versehen.

Für UVO 1758+36 ist die beobachtete Absenkung des UV-Flussverlaufs deutlich schwächer bzw. kaum merklich. Wie für die Analyse von PG 0909+276 wurde vor der Variation der Elementhäufigkeiten die interstellare Extinktion bestimmt. Mit  $E_{B-V} = 0.02$  ist sie deutlich geringer als für PG 0909+276.

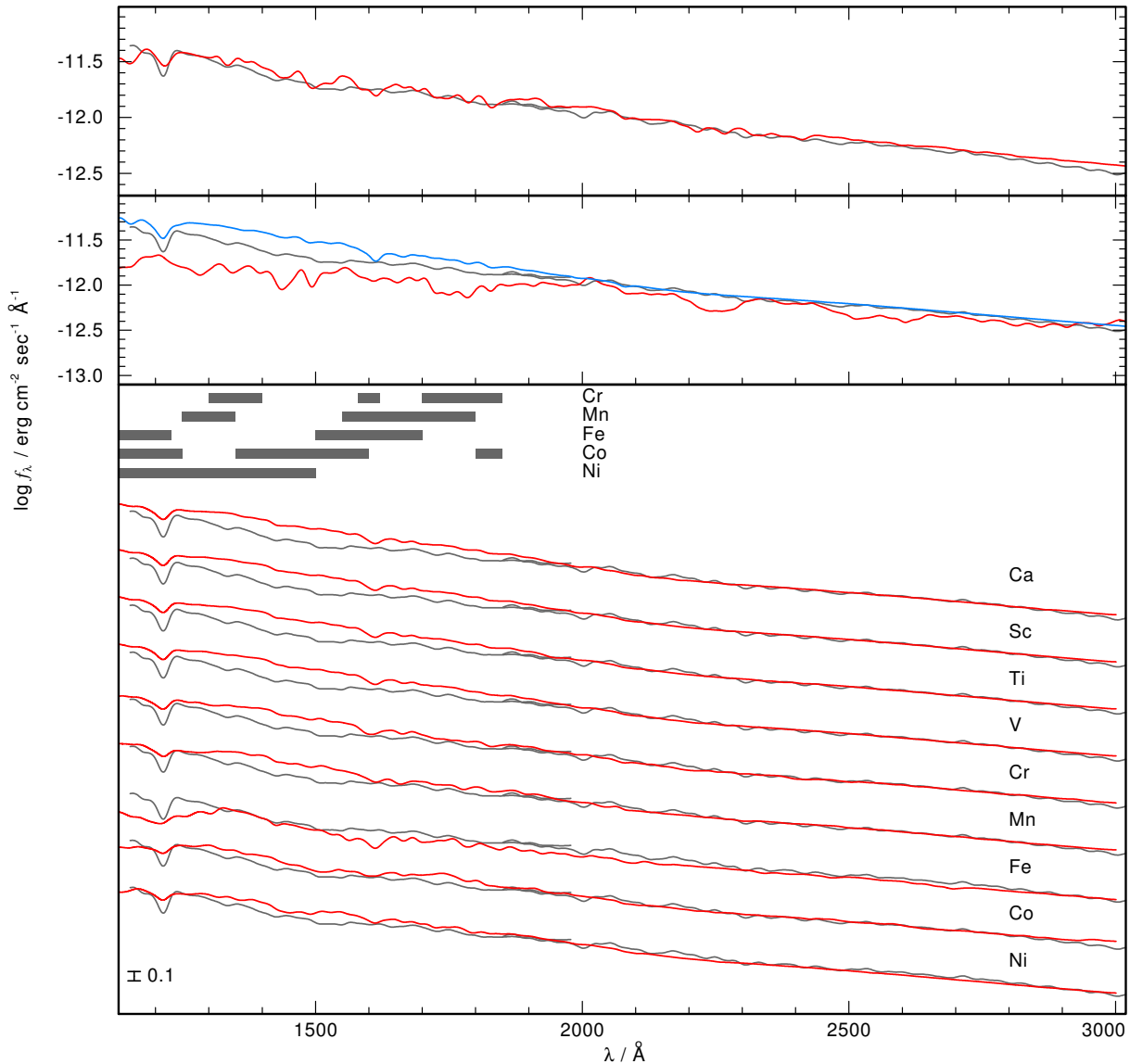


Abbildung 4.48.: Oben: Der beobachtete UV-Flussverlauf von PG 0909+276 (grau) verglichen mit dem Modell mit aus der Analyse resultierenden Häufigkeiten (rot). Mitte: Die IUE-Beobachtung (grau) mit einem Modell mit solaren Häufigkeiten (blau) und den Häufigkeiten von Edelmann (2003, rot). Unten: Die synthetischen Spektren mit individuell erhöhten Elementhäufigkeiten (rot, Teilchenzahlbruchteile, jeweils 1000-fach solarer Wert, 100-fach solarer Wert für Fe und Ni, 10 000-fach solarer Wert für Co und 2000-fach für Cr) verglichen mit der Beobachtung (grau). Dunkelgrau eingezeichnete Balken sind die Bereiche, in denen die für eine Absenkung verantwortlichen Elemente die größte Auswirkung haben. An die Beobachtung wurde jeweils ein Gaussprofil mit Halbwertsbreite  $20 \text{ \AA}$  angebracht, an die Modelle ein Gaussprofil mit Halbwertsbreite  $21 \text{ \AA}$  sowie  $E_{B-V} = 0.07$  (nach Normierung auf  $m_V$ ).

Zuerst wurde wie für PG 0909+276 ein Modell mit den Häufigkeitswerten von Edelmann (2003) bzw. solaren Häufigkeiten für die Elemente ohne Literaturwert berechnet (Tabelle 4.9). Dieses Modell repräsentiert die Beobachtung bereits gut. Wie bei PG 0909+276 hat eine supersolare Ca-, Sc-, Ti- oder V-Häufigkeit (hier 100-fach) keine Auswirkungen auf den Flussverlauf im relevanten Wellenlängenbereich. Eine Änderung der individuellen Häufigkeiten von Cr, Mn, Fe, Co oder Ni bei der Berechnung des

Tabelle 4.8.: Die Elementhäufigkeiten der HCaScTiVCrMnFeCoNi-Modelle von PG 0909+276 verglichen mit dem Literaturwert (Edelmann, 2003). EM steht für das Modell, das die Häufigkeiten von Edelmann (2003) zusätzlich zu solaren für Elemente, zu denen kein Literaturwert vorhanden ist, verwendet.  $\log \epsilon$  steht hier für den Logarithmus des Verhältnisses der Teilchenzahlfrequenz des jeweiligen Elements zur Wasserstoffhäufigkeit, addiert mit 12.

	Literatur	EM	diese Arbeit	
$T_{\text{eff}} / \text{kK}$	36 900	37 000	37 000	
$\log g / \text{cm/sec}^2$	5.9	5.9	5.9	
	$\log \epsilon$	$\log \epsilon$	$\log \epsilon$	[X]
Ca	7.81	7.347	7.938	1.584
Sc	7.10	6.637	7.228	4.064
Ti	7.87–7.99	7.467	8.058	3.094
V	8.10–9.12	8.147	8.738	4.794
Cr		5.177	7.654	2.000
Mn	9.67	9.207	6.444	1.000
Fe		7.037	7.628	0.114
Co		4.527	8.004	3.000
Ni	10.68	10.217	8.234	2.000

synthetischen Spektrums auf einen 100-fach solaren Wert zeigt, dass kein Testmodell den Verlauf besser reproduziert (Abb. 4.49). In dem Modell mit 100-fach solarer Nickelhäufigkeit ist zwar der theoretische Fluss im Bereich zwischen 1250 und 1420 Å näher an der Beobachtung als mit den Werten von Edelmann (2003), zwischen 1420 und 1600 Å allerdings deutlich zu tief. Zum Vergleich wurde ein Spektrum mit solaren Elementhäufigkeiten berechnet. Der resultierende Flussverlauf unterscheidet sich kaum von dem mit Häufigkeiten von Edelmann (2003). Die Elementhäufigkeiten von Edelmann (2003) können also nicht widerlegt werden.

## Diffusion

Das *NGRT*-Modell konnte für EC 11481–2303 supersolare Eisengruppenhäufigkeiten bestätigen. Da die Eisen- und Nickelhäufigkeiten für die beiden kühleren Objekte PG 0909+276 und UVO 1758+36 jeweils kleiner sind, wurden auch für diese beiden Objekte Diffusionsmodelle berechnet. Als Startmodell wurde jeweils das Modell gewählt, das die Eisengruppenelemente mit solaren Häufigkeiten berücksichtigt. Die supersolaren Werte der Eisengruppenhäufigkeiten können mit dem resultierenden Modell für PG 0909+276 ebenfalls bestätigt werden (Abb. 4.50). Der Flussverlauf beider Objekte (Abb. 4.51) wird aber mit dem *TMAP*-Modell besser reproduziert. Vor allem für UVO 1758+36 sind die Eisengruppenhäufigkeiten des Diffusionsmodells deutlich zu hoch, obwohl Abb. 4.50 zeigt, dass sie im Vergleich zum heißeren Modell von PG 0909+276 etwas kleiner sind. Das Diffusionsmodell kann also wieder supersolare Häufigkeiten bestätigen, eine homogene Atmosphäre reproduziert die Beobachtung jedoch besser.

## Auswirkungen auf die Balmerlinien

Bereits im vorherigen Kapitel 4.1 konnte gezeigt werden, dass die Berücksichtigung von Eisengruppenelementen Einfluss auf die Linienprofile der von  $n = 4$  ausgehenden optischen HeII-Serie hat. Bei EC 11481–2303 ändern sich die Linienprofile der Balmerlinien durch die Berücksichtigung der Eisengruppenelemente. PG 0909+276 und UVO 1758+36 zeigen niedrigere Effektivtemperaturen, ein Einfluss der erhöhten Eisengruppenhäufigkeit auf die Balmerlinien wird dennoch erwartet. Um das zu überprüfen wurde in das finale Modell beider Objekte zusätzlich He, C, N und O mit den Häufigkeiten

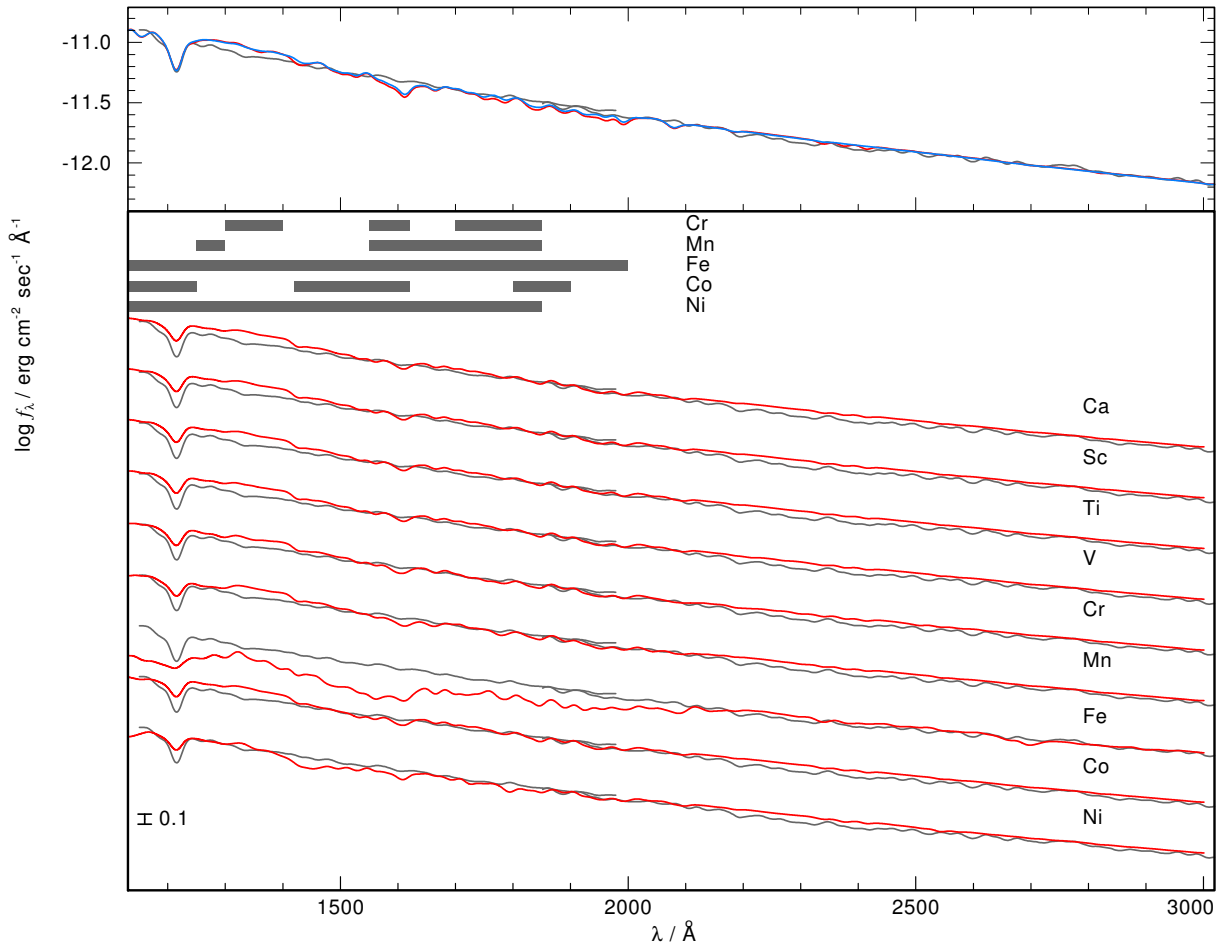


Abbildung 4.49.: Oben: Ausschnitt aus der IUE-Beobachtung von UVO 1758+36 (grau) verglichen mit Modellen mit Parametern von Edelmann (2003, rot) und mit solaren Häufigkeiten (blau). Unten: Modelle (rot) mit individuell auf einen 100-fach solaren Wert (Teilchenzahlbruchteile) geänderte Elementhäufigkeit verglichen mit der Beobachtung. In beiden Abbildungen wurde an die Beobachtung ein Gaussprofil mit Halbwertsbreite  $20 \text{ \AA}$  angebracht, an die Modelle ein Gaussprofil mit Halbwertsbreite  $21 \text{ \AA}$  sowie  $E_{B-V} = 0.02$  (normiert auf  $m_K$ ).

von Edelmann (2003) eingefügt. Zum Vergleich wurde zusätzlich ein reines HHeCNO-Modell mit den Literaturhäufigkeiten berechnet. Die Häufigkeiten aller Modelle sind in Tabelle 4.10 dargestellt. Die Unterschiede zwischen den einzelnen Werten eines Elements werden durch die Normierung aller Elemente eines Modells verursacht.

Die Linienprofile der Balmerreihe der Modelle mit Eisengruppenelementen weichen von denen des HHeCNO-Modells für PG 0909+276 deutlich ab (Abb. 4.52). Am stärksten ist diese Abweichung für die  $H\alpha$ -Linie. Die Unterschiede zwischen den Linienprofilen des HCaScTiVCrMnFeCoNi- und des HHeCNOCaScTiVCrMnFeCoNi-Modells sind deutlich kleiner, nehmen aber für höhere Serienmitglieder zu. Ein Blick auf die Temperaturstrukturen zeigt die Ursache für diesen Sachverhalt (Abb. 4.53). Bei den Entstehungstiefen der Linienkerne der höheren Serienmitglieder weicht die Temperatur der Modelle ab der  $H\gamma$ -Linie zunehmend ab. Die Unterschiede im Bereich der Entstehung der  $H\alpha$ -Linie ist insgesamt für alle Modelle noch stärker.

Für UVO 1758+36 fallen die Abweichungen der Balmerlinienprofile deutlich kleiner aus. Die Liniinflügel sind weniger beeinflusst als bei PG 0909+276. Für die  $H\alpha$ -Linie sind die Unterschiede wieder

Tabelle 4.9.: Die Elementhäufigkeiten der HCaScTiVCrMnFeCoNi-Modelle von UVO 1758+36 verglichen mit dem Literaturwert (Edelmann, 2003).

	Literatur	diese Arbeit	
$T_{\text{eff}} / \text{kK}$	32 000	32 000	
$\log g / \text{cm/sec}^2$	5.8	5.8	
	$\log \epsilon$	$\log \epsilon$	[X]
Ca	8.18	8.323	1.969
Sc		3.293	0.129
Ti	7.26	7.403	2.439
V		4.073	0.128
Cr		5.783	0.129
Mn		5.573	0.129
Fe		7.643	0.128
Co		5.133	0.128
Ni		6.363	0.129

Tabelle 4.10.: Die Elementhäufigkeiten der für den Vergleich der Balmerlinien verwendeten HHeCNOCaScTiVCrMnFeCoNi- (jeweils links), HCaScTiVCrMnFeCoNi- (Mitte) und HHeCNO-Modelle (rechts) von PG 0909+276 und UVO 1758+36 in Massenbruchteilen.

	PG 0909+276			UVO 1758+36		
H	$6.79 \times 10^{-1}$	$9.59 \times 10^{-1}$	$7.09 \times 10^{-1}$	$9.27 \times 10^{-1}$	$9.91 \times 10^{-1}$	$9.32 \times 10^{-1}$
He	$2.76 \times 10^{-1}$		$2.88 \times 10^{-1}$	$6.70 \times 10^{-2}$		$6.74 \times 10^{-2}$
C	$1.90 \times 10^{-3}$		$1.98 \times 10^{-3}$	$1.05 \times 10^{-4}$		$1.06 \times 10^{-4}$
N	$8.61 \times 10^{-4}$		$8.98 \times 10^{-4}$	$4.17 \times 10^{-4}$		$4.19 \times 10^{-4}$
O	$2.47 \times 10^{-4}$		$2.58 \times 10^{-4}$	$1.07 \times 10^{-4}$		$1.07 \times 10^{-4}$
Ca	$2.46 \times 10^{-3}$	$2.46 \times 10^{-3}$		$4.04 \times 10^{-3}$	$5.97 \times 10^{-3}$	
Sc	$5.38 \times 10^{-4}$	$5.38 \times 10^{-4}$		$4.23 \times 10^{-8}$	$6.25 \times 10^{-8}$	
Ti	$3.88 \times 10^{-3}$	$3.88 \times 10^{-3}$		$5.80 \times 10^{-4}$	$8.57 \times 10^{-4}$	
V	$1.97 \times 10^{-2}$	$1.97 \times 10^{-2}$		$2.89 \times 10^{-7}$	$4.26 \times 10^{-7}$	
Cr	$1.66 \times 10^{-3}$	$1.66 \times 10^{-3}$		$1.51 \times 10^{-5}$	$2.23 \times 10^{-5}$	
Mn	$1.08 \times 10^{-4}$	$1.08 \times 10^{-4}$		$9.85 \times 10^{-6}$	$1.46 \times 10^{-5}$	
Fe	$1.68 \times 10^{-3}$	$1.68 \times 10^{-3}$		$1.18 \times 10^{-3}$	$1.74 \times 10^{-3}$	
Co	$4.21 \times 10^{-3}$	$4.21 \times 10^{-3}$		$3.84 \times 10^{-6}$	$5.66 \times 10^{-6}$	
Ni	$7.13 \times 10^{-3}$	$7.13 \times 10^{-3}$		$6.49 \times 10^{-5}$	$9.59 \times 10^{-5}$	

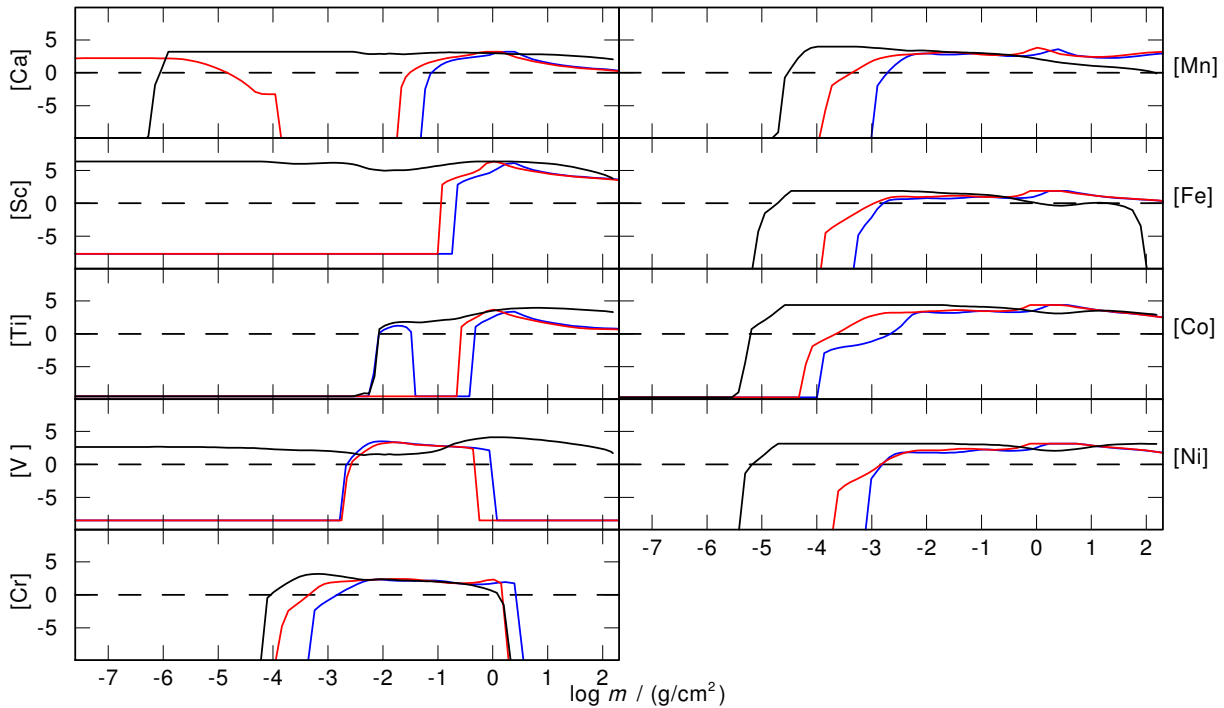


Abbildung 4.50.: Elementhäufigkeiten in der Atmosphäre der Diffusionsmodelle von PG 0909+276 (rot) und UVO 01758+36 (blau). Zum Vergleich ist das Modell von EC 11481–2303 (schwarz) eingezeichnet.

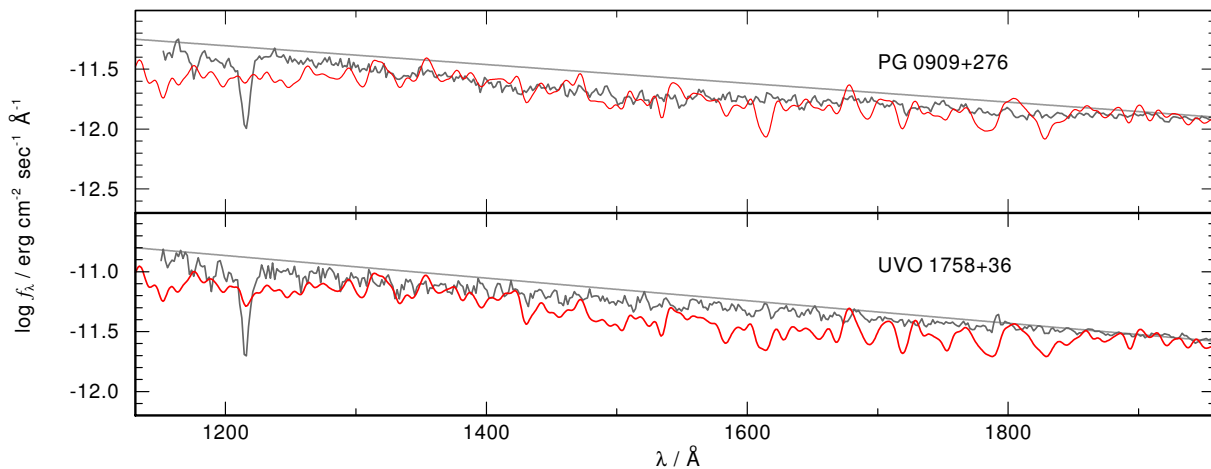


Abbildung 4.51.: UV-Flussverlauf von PG 0909+276 und UVO 1758+36 (grau) verglichen mit dem Diffusionsmodell (rot), an das  $E_{B-V} = 0.07$  bzw.  $E_{B-V} = 0.02$  angebracht ist.

am stärksten. Bei der Entstehungstiefe des Kerns dieser Linie weichen auch die Temperaturstrukturen der Modelle im Vergleich zu den Entstehungstiefen der anderen Balmerlinien am stärksten ab. Für sinkende Häufigkeiten der Eisengruppenelemente, die den Flussverlauf beeinträchtigen, reduziert sich also insgesamt auch die Abweichung der optischen Linienprofile.

## Ergebnis

Zusammenfassend lässt sich sagen, dass bestätigt werden konnte, dass der flache UV-Flussverlauf der hier untersuchten sd(O)B-Sterne von Eisengruppenelementen verursacht wird. Die Ergebnisse sind in

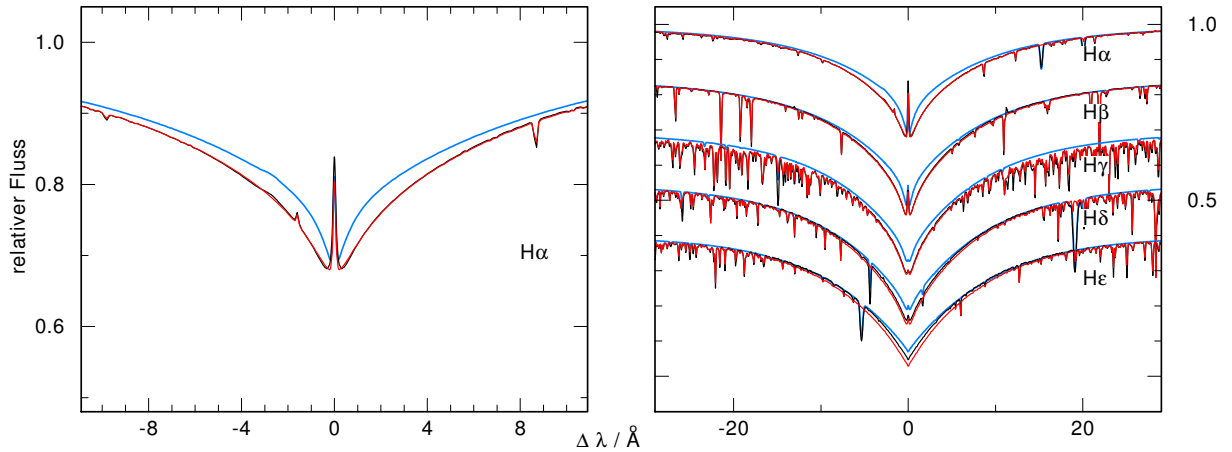


Abbildung 4.52.: Die Linienprofile der Balmerserie von PG 0909+276 des HHeCNO- (blau), des HCaScTiVCrMnFeCoNi- (rot) und des HHeCNOCaScTiVCrMnFeCoNi-Modells (schwarz). An die Modelle wurde ein Gaussprofil der Halbwertsbreite  $0.1 \text{ \AA}$  angebracht.

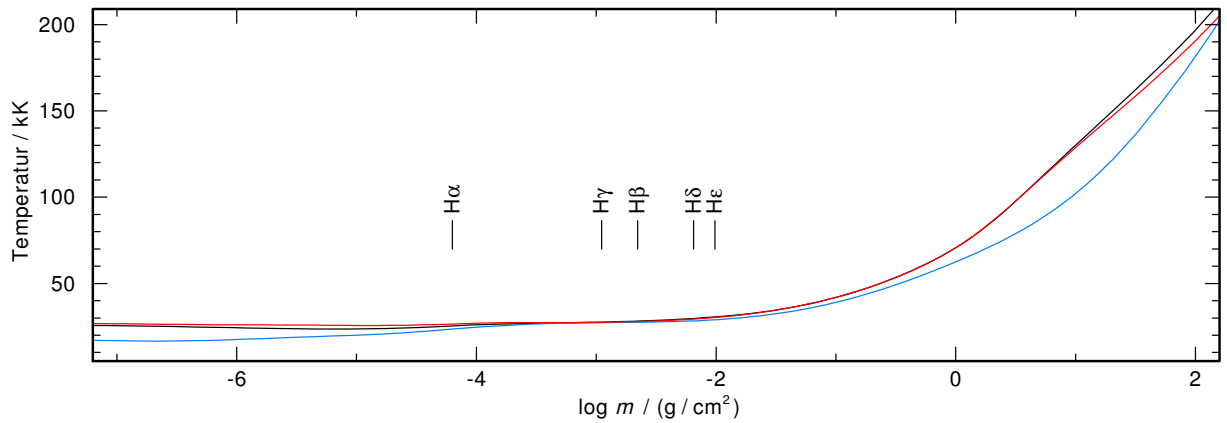


Abbildung 4.53.: Temperaturstrukturen des HHeCNO- (blau), HCaScTiVCrMnFeCoNi- (rot) und HHeCNOCaScTiVCrMnFeCoNi-Modells (schwarz) von PG 0909+276. Die Entstehungstiefen der Balmerlinienkerne sind zusätzlich eingezeichnet.

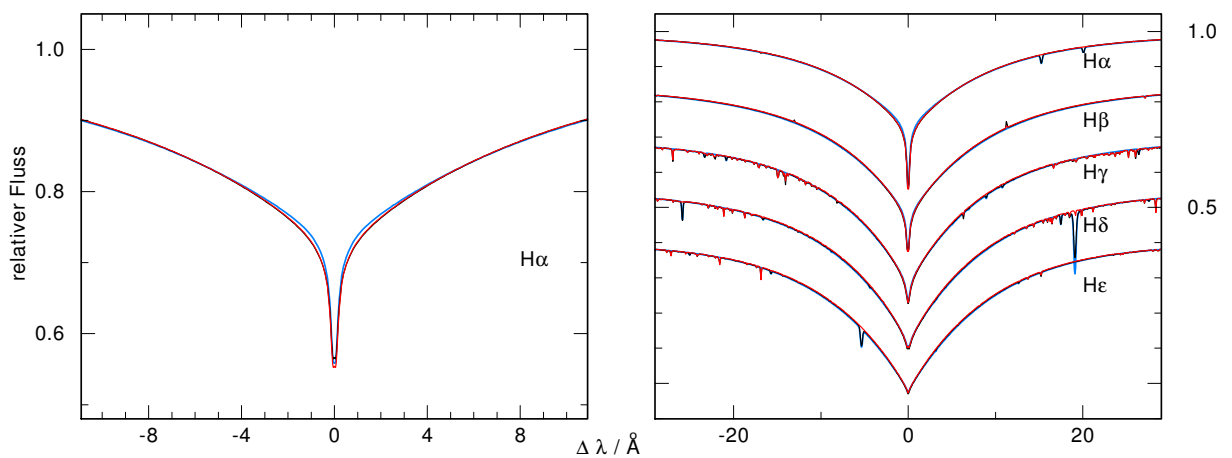


Abbildung 4.54.: Wie Abb. 4.52, für UVO 1758+36.

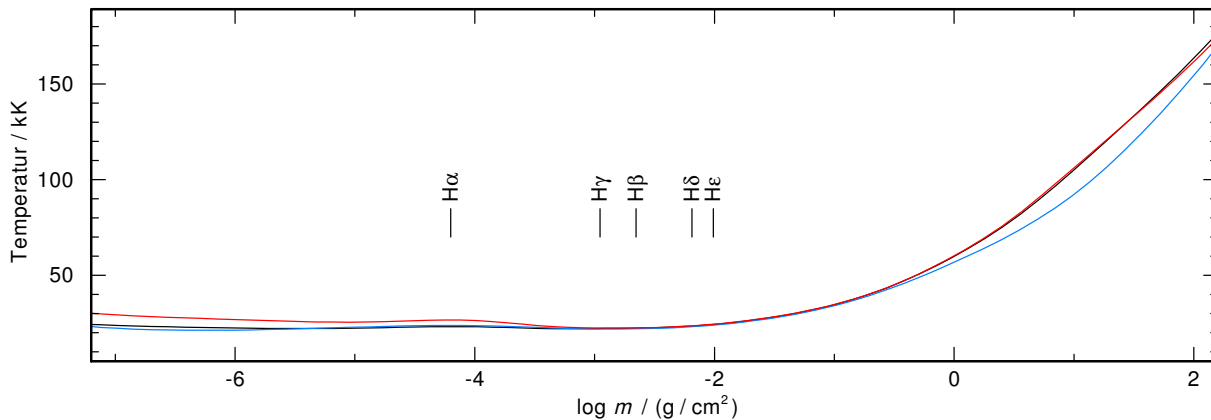


Abbildung 4.55.: Wie Abb. 4.53, für UVO 1758+36.

Übereinstimmung mit dem von Geier et al. (2010) bestimmten Trend der steigenden Eisengruppenhäufigkeiten mit der Temperatur. Allerdings scheinen abhängig von der Temperatur unterschiedliche Elemente für das Absinken des Flussverlaufs verantwortlich zu sein. Beim heißesten Objekt, EC 11481–2303, ist eine Kombination aus erhöhtem Ni- und Fe-Wert zur Reproduktion des Verlaufs notwendig, während im kühleren Objekt PG 0909+276 Co und Ni verantwortlich sind und im kühlpsten der beobachteten Objekte (UVO 1758+36) keines der fünf flussabsenkenden Elemente (Cr, Mn, Fe, Co oder Ni) den Verlauf abzusinken scheint.

Die von Edelmann (2003) bestimmten Häufigkeitswerte sind für PG 0909+276 eindeutig zu hoch. Für das dritte Objekt aus dieser Gruppe – UVO 0512–08 – sind keine Beobachtungsdaten im IUE-Bereich vorhanden, weshalb hier keine Aussage über die Häufigkeiten getroffen werden kann. Für UVO 1758+36 können die Literaturwerte generell bestätigt werden.

Co und Cr wurden von Edelmann (2003) nicht analysiert. Die Werte von Geier et al. (2010) für deren Auswahl von sdB-Sternen bilden Grenzen (Abb. 4.56), in denen die hier bestimmten supersolaren Werte von PG 0909+276 – trotz der großen Unsicherheiten durch die interstellare Extinktion und einer Analyse des Gesamtverlaufs anstelle einzelner Linien – für die meisten Elemente liegen. Für UVO 1758+36 wären Häufigkeiten innerhalb dieser Grenzen zu hoch. Der hier für dieses Objekt bestimmte solare Wert weicht deutlich von den erwarteten Werten ab. Die Fe-Häufigkeit liegt für beide Objekte in den erwarteten Grenzwerten. Da Sc, Ti und V kaum merkliche Auswirkungen auf den Flussverlauf haben, kann über deren Häufigkeiten nur bedingt eine Aussage getroffen werden. Die supersolaren Werte von Edelmann (2003) stehen nicht im Widerspruch zu den Ergebnissen dieser Analyse. Das selbe gilt für Ca.

Ni und Mn wurden von Geier et al. (2010) nicht analysiert. Der Flussverlauf des Modells zeigt allerdings, dass die Werte von Edelmann (2003) für PG 0909+276 deutlich zu hoch sind. Für UVO 1758+36 sind sie mit den Erkenntnissen aus dieser Arbeit konsistent. Die Häufigkeiten dieser Elemente liegen aber für beide Objekte innerhalb der Literaturwerte von O’Toole & Heber (2006) für deren Auswahl an sdB-Sternen.

Für diese Analyse wurden die in Kapitel 2.2 beschriebenen LIN-Atomdaten verwendet. Damit sind die Opazitätsquellen ausreichend berücksichtigt. Einzelne Linien haben allerdings große Unsicherheiten – sowohl in den Oszillatorenstärken als auch in der Position. Aus diesem Grund wurden einzelne spektrale Kennzeichen ignoriert und nur der gesamte Verlauf betrachtet. Ein Grund für die Abweichungen zu den Ergebnissen von Edelmann (2003) und Geier et al. (2010) können diese verwendeten Atomdaten sein. In den beiden früheren Analysen wurden Atomdaten für die Eisengruppenelemente von Kurucz (1992), Ekberg (1993) (Fe) und Hirata & Horaguchi (1995) verwendet. Mittlerweile gibt es aber z.B. von Kurucz verbesserte Atomdaten, wodurch mehr experimentell bestätigte Linien für eine Analyse zur Verfügung stehen.

Obwohl die Objekte im NLTE-Bereich liegen, wurden in den beiden früheren Analysen LTE-Modelle



zur Häufigkeitsbestimmung verwendet, was vor allem bei der Betrachtung von Resonanzlinien zu Abweichungen gegenüber NLTE-Modellen führt. Napiwotzki (1997) zeigte, dass für  $T_{\text{eff}} \gtrsim 30\,000\text{ K}$  NLTE-Modelle für sdBs und vor allem für sdOBs verwendet werden sollten, dass aber die Berücksichtigung des Blanketing der Metalllinien einen ähnlich großen Effekt hat. Da in der Studie der NLTE-Grenztemperatur aber hauptsächlich H und He untersucht wurden, wurden in dem am besten passenden Wasserstoff-Eisengruppenelementmodell testweise die Stoßraten erhöht, um LTE-Bedingungen zu erreichen. Die resultierenden Unterschiede zwischen beiden Modellen haben ihren Ursprung allein in LTE-NLTE-Abweichungen, nicht in Unterschieden zweier Modellatmosphären-codes. Diese Unterschiede sind zwar zu erkennen, haben aber bei der Betrachtung des UV-Flusses, solange die spektralen Kennzeichen nicht analysiert werden, vernachlässigbare Auswirkungen.

Die beiden früheren Analysen wurden im optischen Wellenlängenbereich durchgeführt, in welchem bei diesen  $T_{\text{eff}}$  deutlich weniger Eisengruppenlinien als im UV-Bereich zu finden sind. Abweichungen durch diese Tatsache sind allerdings nicht zu erwarten, da UV-Analysen bisher – trotz der großen Streuung der Häufigkeiten der einzelnen Sterne – mit optischen Betrachtungen konsistente Ergebnisse liefern (z.B. O’Toole & Heber, 2006).

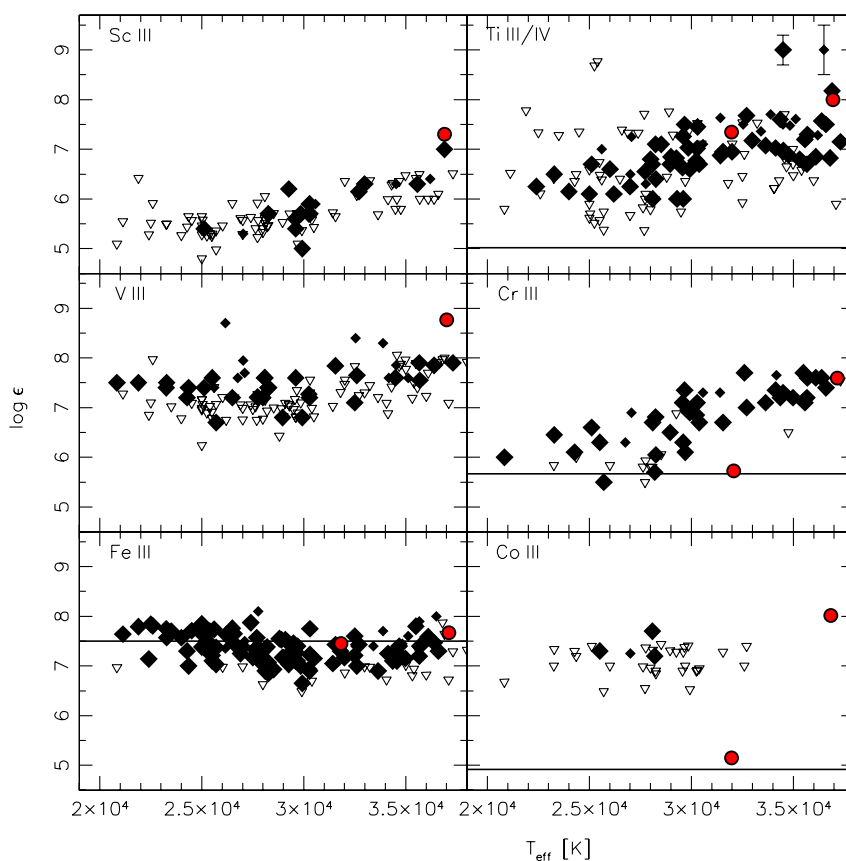


Abbildung 4.56.: Abb. 4 aus Geier et al. (2010, modifiziert). Dargestellt sind die Elementhäufigkeiten oder Obergrenzen (Dreiecke) verschiedener sdB-Sterne. Der solare Wert wird durch die horizontale Linie repräsentiert. Rot eingezeichnet sind die hier bestimmten Häufigkeitswerte (Tabelle 4.10).

## 4.4. (Interstellare) Absorption im EUV-Bereich

In diesem Kapitel wird der DO-Weiße Zwerg RE 0503–289 untersucht. Bei diesem Objekt wurde ein extrem niedriger EUV-Flussverlauf beobachtet, der bisher nicht konsistent reproduziert werden kann. Als Ursache für diesen Verlauf wurde bisher angenommen, dass die Modelle wichtige Absorptionsquellen vernachlässigen. Vielversprechende Kandidaten für diese Absorptionsquellen sind (wieder) die Eisen-Gruppenlemente, weshalb der Einfluss von Modellen mit erneuerten Atomdaten für die Eisen-Gruppenelemente für die Literaturparameter von RE 0503–289 untersucht wird. Im Lauf dieser Untersuchung zeigt sich jedoch, dass die Eisen-Gruppenelemente nicht die (alleinige) Ursache für den beobachteten EUV-Flussverlauf sein können. Es wird deshalb der Einfluss interstellarer Absorption auf das Modellspektrum untersucht. Dafür wird ein eigenes VO-Tool entwickelt, mit dem der Einfluss interstellarer Absorption durch Ionen von H, He, C, N und O untersucht werden kann. Dieses neue Tool wird anschließend zur Analyse des DO-Weißen Zwergs RE 0503–289 eingesetzt.

In diesem Kapitel werden zuerst die Eigenschaften und Entwicklung von DO-Weißen Zwergen allgemein kurz beschrieben sowie RE 0503–289 und die Untersuchungen zu diesem Objekt vorgestellt. Dann wird das in dieser Arbeit entwickelte VO-Tool zur Berücksichtigung interstellarer Absorption sowie die Anwendung auf RE 0503–289 beschrieben.

### 4.4.1. Eigenschaften von DO-Weißen Zwergen

Im Gegensatz zu den zahlenmäßig überlegenen, wasserstoffdominierten DA-Weißen Zwergen besitzen DO-Weiße Zwerge eine fast reine Heliumatmosphäre. Nur in zwei DO-Weißen Zwergen konnte bisher Wasserstoff nachgewiesen werden und Metalle sind mit deutlich niedrigeren Häufigkeiten in der Atmosphäre vertreten als z.B. bei PG 1159-Sternen. Eine aktuelle Übersicht über die bekannten DO-Weißen Zwerge wird in (Sion, 2011) gegeben. Ihre Parameter liegen in einem Bereich mit  $T_{\text{eff}} > 45\,000\text{ K}$  und  $\log g > 7$ . Das heißeste Objekt dieser Klasse besitzt eine Effektivtemperatur von  $200\,000\text{ K}$  (KPD 0005+5106, Wassermann et al., 2010). Analysen aller bis dahin bekannten DO-Weißen Zwerge wurden beispielsweise von Dreizler & Werner (1996) durchgeführt. Es wird angenommen, dass sich in den Atmosphären dieser Sterne ein Gleichgewicht zwischen radiativem Auftrieb und gravitativem Absinken der einzelnen Elemente eingestellt hat. Allerdings stimmen die beobachteten Häufigkeiten der DO-Weißen Zwerge nicht mit den aus Diffusionsmodellen vorhergesagten überein (z.B. Chayer et al., 1995; Werner et al., 1997). Dreizler (1999) versuchte die Atmosphären der DO-Weißen Zwerge mit Diffusionsmodellen zu reproduzieren, allerdings stimmen die Spektren homogener Modelle besser mit der Beobachtung überein. Vermutlich ist in diesen Objekten ein schwacher Wind vorhanden, der keine Anzeichen im Spektrum hinterlässt, die Diffusion aber beeinflusst (Unglaub & Bues, 2000). Durch die Diffusion ändern sich die Elementhäufigkeiten, sodass schwere Elemente in der Atmosphäre angereichert, leichtere abgereichert werden.

Vorläufer der DO-Weißen Zwerge sind vermutlich die PG 1159-Sterne. Durch das Einsetzen des gravitativen Absinkens während ihrer Entwicklung verschwinden zuerst die schweren Elemente aus der Atmosphäre der PG 1159-Sterne und ein DO-Weißer Zwerg entsteht (Unglaub & Bues, 2000). Im Lauf der Entwicklung sinkt die Massenverlustrate, der Sternwind der PG 1159-Sterne verschwindet langsam und der Auftrieb wird im Lauf der Zeit schwächer.

Die Tatsache, dass die beobachteten Häufigkeiten theoretisch (mit Diffusionsmodellen) noch nicht erklärt werden können, zeigt, dass auf diesem Gebiet noch viel Arbeit nötig ist, um ein konsistentes Entwicklungsmodell zu finden. Eine alternative Entwicklungssequenz, die darauf beruht, dass KPD 0005+5106 Häufigkeiten ähnlich der R Coronae Borealis-Sterne (RCB) zeigt, wäre von diesen RCB-Sternen zu DO-Weißen Zwergen (Wassermann et al., 2010). Da RCB-Sterne wahrscheinlich durch Verschmelzung zweier Sterne (Merger) entstanden sind, ist es möglich, dass auch (einige) DO-Weiße Zwerge Mergerprodukte sind.

Tabelle 4.11.: Frühere Analysen von RE 0503–289.  $T_{\text{eff}}$  ist in kK,  $\log g$  in  $\text{cm}/\text{sec}^2$  angegeben. <sup>a</sup> kennzeichnet Werte, die nicht selbst bestimmt, sondern festgesetzt wurden. PM steht für Photometrie, gs für geschichtet.

$T_{\text{eff}}$	$\log g$	Modell-atmosphäre	Methode	Bemerkung	Referenz
60–90			EUV, PM	niedriges $n_{\text{H I}}$	Vennes et al. (1992)
60–80			OPT EUV	niedriges $n_{\text{H I}}$	Barstow et al. (1993)
70	7.5	He, HHeC	NLTE UV+OPT EUV	EUV-Problem	Barstow et al. (1994)
60–70			EUV, PM	niedriges $n_{\text{H I}}$	Vennes et al. (1994)
70	7.0	HeCNOSiFeNi	LTE EUV+UV		Polomski et al. (1995)
65 <sup>a</sup>	7.5	HHeC	NLTE OPT	kein H nachweisbar	Werner (1996)
70	7.5	HHeCNOSi	NLTE OPT+UV	$0.49 M_{\odot}$	Dreizler & Werner (1996)
68.6	7.20	HHe	LTE UV	$0.4 M_{\odot}$	Vennes et al. (1998)
70	7.5		NLTE gs	kein guter Fit	Dreizler (1999)
69–75	7.26– 7.63	HHeC, HHeCNOSiFeNi	NLTE UV+OPT		Barstow et al. (2000)
63			NLTE EUV	EUV-Problem	Barstow et al. (2000)
65–70	7.5	HeCNi, HeONi	NLTE EUV	EUV-Problem	Werner et al. (2001)
70 <sup>a</sup>	7.5 <sup>a</sup>	HHeCNOSiFeNi +PS(LTE)	NLTE UV	EUV-Problem	Barstow et al. (2007)

#### 4.4.2. RE 0503–289 und das EUV-Problem

##### Bisherige Analysen von RE 0503–289

RE 0503–289 ist der einzige DO-Weiße Zwerg (WD 0501-289), der im ROSAT EUV Survey detektiert wurde (Barstow et al., 1993) und einer von nur zwei DOs, die mit dem Extreme Ultraviolet Explorer (EUVE<sup>11</sup>) beobachtet werden konnten (Vennes et al., 1992). Im Gegensatz zu DA-Weißen Zwergen sorgen die Opazitäten von Helium und Metallen in DO-Weißen Zwergen sowie interstellare Elemente dafür, dass der Fluss im EUV-Bereich deutlich abgeschwächt wird, wodurch die meisten DO-Weißen Zwerge nicht im EUV-Bereich beobachtbar sind. Bei den beiden Ausnahmen (HD 149499 B und RE 0503–289) handelt es sich um den hellsten DO-Weißen Zwerg ( $m_V = 9.75$ ) bzw. um ein Objekt mit einer extrem niedrigen interstellaren H I-Säulendichte ( $n_{\text{H I}} \approx 10^{17} - 6 \times 10^{18} \text{cm}^{-2}$ , Vennes et al., 1992; Barstow et al., 1994). Damit ist RE 0503–289 der einzige DO-Weiße Zwerg für den Beobachtungen über den gesamten Wellenlängenbereich vom Infraroten zum extremen UV ohne Unterbrechungen möglich sind.

Identifiziert wurde RE 0503–289 mit dem Montreal-Cambridge-Tololo-Survey (Vennes et al., 1992) und mit dem Edinburgh-Cape-Survey (Barstow et al., 1993). Die optischen Spektren von RE 0503–289 zeigen den für PG 1159-Sterne typischen C IV- und He II-Absorptionstrog bei  $4645 - 4686 \text{Å}$ , die C-Häufigkeit ist allerdings mit 1–3% (Teilchenzahlbruchteil) deutlich geringer als bei PG 1159-Sternen (Barstow et al., 1993, 1994). Unglaub & Bues (2000) gehen deshalb davon aus, dass es sich bei diesem Objekt um ein Übergangsobjekt zwischen PG 1159-Sternen und DO-Weißen Zwergen handeln könnte. Eventuell zeigt RE 0503–289 auch noch Anzeichen eines Sternwinds (Barstow & Sion, 1994), was ebenfalls dafür sprechen würde, dass er noch nicht ganz im DO-Bereich angekommen ist und sich noch kein Gleichgewicht zwischen gravitativem Absinken und strahlungsgetriebenem Auftrieb eingestellt hat.

<sup>11</sup><http://www.ssl.berkeley.edu/euve/>

Die Beobachtungen von RE 0503–289 decken mittlerweile den gesamten Wellenlängenbereich beinahe lückenlos ab. Analysen dieser Spektren (aufgenommen mit SOFI<sup>12</sup>, CASPEC, EMMI<sup>13</sup>, IUE, GHRS, FUSE, ORFEUS<sup>14</sup>, Voyager<sup>15</sup>, ROSAT<sup>16</sup>, EUVE<sup>17</sup>) führten zu  $T_{\text{eff}} \approx 70\,000\text{ K}$  und  $\log g \approx 7.5$  (siehe Tabelle 4.11). Dabei stellte sich unter anderem heraus, dass geschichtete Modelle die Beobachtung deutlich schlechter reproduzieren als homogene (Dreizler, 1999), was noch nicht erklärt werden kann. Insgesamt zeigt RE 0503–289 auch höhere Metallhäufigkeiten als die meisten anderen DOs und die beobachteten Häufigkeiten weichen stark von den Vorhersagen von Diffusionsmodellen ab (Dreizler & Werner, 1996).

Bisher ist es nicht gelungen, alle Beobachtungen von RE 0503–289 konsistent mit einem Modell wiederzugeben. Modelle mit den im optischen und UV-Bereich bestimmten Temperaturen können den EUV-Bereich nicht reproduzieren (EUV-Problem, Barstow et al., 1993). Um den Flussverlauf im EUV anzupassen, muss die Temperatur der Modelle auf  $60\,000 - 65\,000\text{ K}$  gesenkt werden. Dadurch wird allerdings die optische He I  $\lambda 4471\text{ \AA}$ , die nicht beobachtet ist, deutlich zu stark. Zur Lösung dieses EUV-Problems wurde vorgeschlagen, weitere Elemente in die Modelle einzufügen, die durch zusätzliche Opazitäten eventuell den Fluss im EUV-Bereich senken. In den folgenden Jahren konnten immer mehr Elemente in den Beobachtungen identifiziert werden, der theoretische EUV-Fluss wird durch die Berücksichtigung dieser Elemente jedoch nicht genug abgesenkt um die Beobachtung zu reproduzieren.

Durch ihre Vielzahl an Linien haben die Eisengruppenelemente besonders großes Potential den Flussverlauf zu senken. Erste Identifikationsversuche von Eisen und Nickel waren allerdings erfolglos (Polomski et al., 1995, Tabelle. 4.12). In den GHRS-Spektren konnte später Nickel nachgewiesen werden, für Eisen wurde eine Obergrenze bestimmt bzw. bestätigt (Barstow et al., 2000). Die Berücksichtigung von Fe und Ni führte jedoch nicht zur Lösung des EUV-Problems. Die von Barstow et al. (2000) bestimmte Ni-Häufigkeit musste sogar aufgrund eines nicht beobachteten spektralen EUV-Kennzeichens um einen Faktor zehn auf  $1 \times 10^{-5}$  reduziert werden (Werner et al., 2001).

Auch die Analyse des seit Anfang 2000 verfügbaren FUSE-Spektrums führte nicht zu einer Lösung des EUV-Problems (Barstow et al., 2007), allerdings konnten in diesem Spektrum ungewöhnlich viele Linien nicht identifiziert werden. Werner et al. (2012a) gelang es, einen Teil dieser Linien zuzuordnen. Zum ersten Mal konnten die Elemente Kr, Xe, Ga und Mo in einem Weißen Zwerg nachgewiesen werden. Bei einer Analyse der Ge-, Kr- und Xe-Linien konnte die Häufigkeit dieser Elemente zu extrem supersolaren Werten bestimmt werden (Ge 650-fach solar, Kr 4500-fach solar, Xe 3800-fach solar, Werner et al., 2012a; Rauch et al., 2012). Linien der Elementen Ge, As, Se, Sn, Te und I wurden ebenfalls identifiziert und die Stärke dieser Linien impliziert auch supersolare Häufigkeiten. Diese Elemente wurden bereits in zwei weiteren DO-Weißen Zwergen – mit  $T_{\text{eff}} = 49\,500\text{ K}$  und  $53\,000\text{ K}$  – ebenfalls mit supersolaren Häufigkeiten gefunden (Chayer et al., 2005), jedoch nicht in anderen Weißen Zwergen detektiert. Das Auftreten dieser Elemente stellt eine weitere Herausforderung an die Theorie der Entwicklung der DO-Weißen Zwerge dar. Sowohl ein Merger-Szenario als auch die Entwicklung von PG 1159-Sternen können das Auftreten dieser Elemente mit diesen Häufigkeiten noch nicht erklären.

#### 4.4.3. Neue Ansätze zur Lösung des EUV-Problems

Da für RE 0503–289 noch kein konsistentes Modell erzeugt werden konnte, das den gesamten Wellenlängenbereich reproduziert, wurde das EUV-Problem in dieser Arbeit weiter untersucht. Es wurde bereits vermutet, dass die Absenkung des Flusses durch Opazitäten verursacht wird, die bisher nicht in den Modellen berücksichtigt wurden. Das Modell mit supersolaren Ge-, Kr- und Xe-Häufigkeiten und

<sup>12</sup><http://www.eso.org/sci/facilities/lasilla/instruments/sofi/>

<sup>13</sup><https://www.eso.org/lasilla/instruments/emmi/>

<sup>14</sup><http://archive.stsci.edu/orfeus/>

<sup>15</sup><http://voyager.jpl.nasa.gov/>

<sup>16</sup><http://heasarc.gsfc.nasa.gov/docs/rosat/roskof.html>

<sup>17</sup><https://www.ssl.berkeley.edu/~euve/>

Tabelle 4.12.: Obergrenzen der Eisen- und Nickelhäufigkeit von RE 0503–289 aus früheren Analysen. Die Werte sind in Teilchenzahlbruchteilen relativ zu Helium angegeben.

Element	Polomski et al. (1995)	Barstow et al. (2000)	Werner et al. (2001)
Fe	$< 1.0 \times 10^{-6}$	$< 1.0 \times 10^{-6}$	$< 1.0 \times 10^{-6}$
Ni	$< 1.0 \times 10^{-6}$	$< 1.0 \times 10^{-5}$	$< 1.0 \times 10^{-6}$

$T_{\text{eff}} = 70\,000\text{ K}$  reproduziert den beobachteten Verlauf ebenfalls nicht. Ideale Kandidaten für die zusätzlichen Opazitätsquellen sind deshalb, wie bei den bereits beschriebenen sd(O)B-Sternen, die Eisen-gruppenelemente. Um zu untersuchen, ob eine ähnliche Flussabsenkung wie bei EC 11481–2303 und PG 0909+276 auch im EUV-Bereich bei fast reinen Heliumatmosphären von Eisengruppenelementen verursacht werden kann, wird nun auf eine mögliche Lösung des EUV-Problems mit Hilfe von Eisen-gruppenelementen eingegangen.

### Eisengruppenelemente

Eisen und Nickel wurden bereits in früheren Analysen von RE 0503–289 berücksichtigt und Obergrenzen der Häufigkeiten bestimmt (Tabelle 4.12). Zur Lösung des EUV-Problems hat die Einbeziehung dieser beiden Elemente bisher nicht beigetragen. Allerdings sind seit der Aktualisierung und Erweiterung der Linienlisten von Kurucz einige neue Linien für Eisen und Nickel hinzugekommen. Außerdem wurden die anderen Eisengruppenelemente bisher nicht in den Rechnungen berücksichtigt. Für diese Arbeit wurden deshalb Modelle berechnet, die H, He, C, N, O sowie die Eisengruppenelemente individuell berücksichtigen. Die Häufigkeitswerte sowie  $T_{\text{eff}}$  und  $\log g$  wurden Dreizler & Werner (1996) entnommen. Die Eisengruppenelemente wurden solar berücksichtigt, Fe ein Zehntel solar. Die dadurch angenommenen Häufigkeiten von Fe und Ni sind mit einem Teilchenzahlbruchteil relativ zu Helium von  $9.35 \times 10^{-6}$  und  $4.86 \times 10^{-6}$  etwas höher als die Literaturwerte von Polomski et al. (1995) und Werner et al. (2001). Ausgehend von diesem Modell wurden weitere Modelle berechnet, in denen die Häufigkeiten der Eisengruppenelemente einzeln auf einen zehnfach solaren Wert erhöht wurden. Zusätzlich wurde ein HHeCNO-Modell mit den selben Häufigkeiten berechnet. Die resultierenden Modellflüsse wurden mit der Beobachtung im FUSE- (ID: P2041601000)- und EUVE-Bereich (ID: mct\_0501-2858\_9310202237N, 70–760 Å) verglichen (Abb. 4.57). In dieser Abbildung sind die Modellflüsse mit erhöhter Sc-, Ti- und V-Häufigkeit nicht dargestellt, da die Erhöhung dieser Elemente keine sichtbaren Auswirkungen auf den Flussverlauf im betrachteten Bereich hat. Cr-, Mn- und Co-Linien verursachen bei Erhöhung der Häufigkeit um einen Faktor zehn nur eine sehr geringe Absenkung des Modellflusses. Signifikanten Einfluss auf den Flussverlauf haben die Elemente Ca, Fe und Ni. Eine zehnfach solare Fe- oder Ni-Häufigkeit steht allerdings im Widerspruch zu den Literaturwerten und kann deshalb ausgeschlossen werden. Insgesamt zeigt sich, dass der Wellenlängenbereich, in dem die Eisengruppenelemente Einfluss auf den EUV-Fluss haben, nicht groß genug ist, um eine Absenkung des Flussverlaufs im beinahe gesamten (abgesenkten) EUVE-Bereich zu verursachen. Der beobachtete flache EUV-Flussverlauf kann also auch mit den neuen Kurucz-Daten nicht reproduziert werden. Die Eisengruppenelemente fallen somit als (alleinige) Erklärung für den beobachteten Verlauf aus.

#### 4.4.4. TEUV– Korrektur der interstellaren Absorption

In beinahe jedem Spektrum (zumindest für Wellenlängenbereiche kurzweilig vom optischen Bereich) sind außer den photosphärischen Linien des Sterns auch Anzeichen von interstellarer Materie (ISM) vorhanden. Diese können sowohl von Übergängen zwischen Niveaus interstellarer Ionen als auch von Photoionisationen dieser Ionen stammen. Von der Tübinger Arbeitsgruppe wird das Programm *OWENS* verwendet, um interstellare Linien im UV-Wellenlängenbereich zu modellieren und so von photosphä-

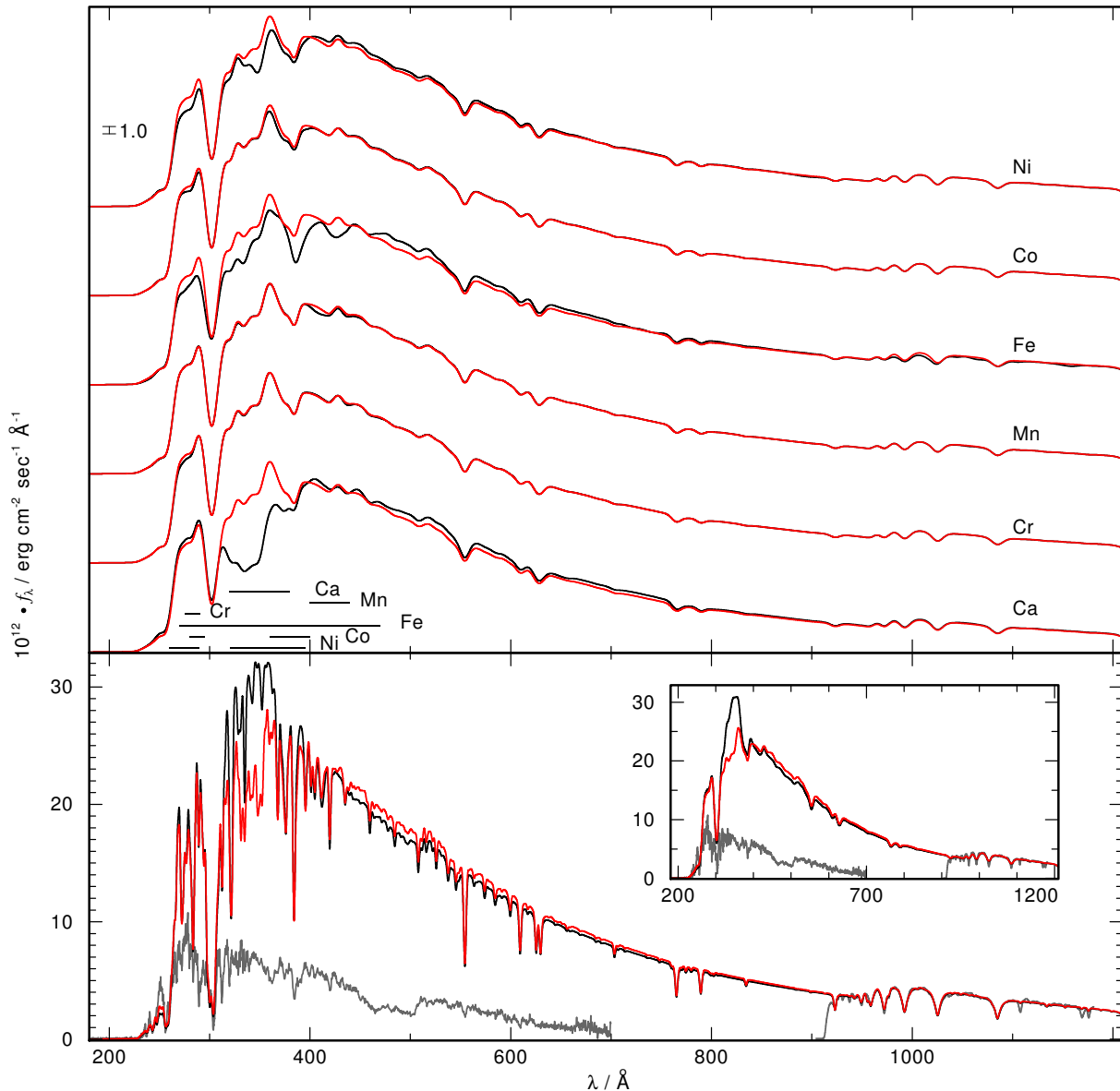


Abbildung 4.57.: Oben: Der FUSE- und EUVE-Bereich des HHeCNOCaScTiVCrMnFeCoNi-Modells (rot, Gaussprofil der Halbwertsbreite  $10 \text{ \AA}$ ) verglichen mit Modellen, bei denen je ein Eisengruppenelement auf einen zehnfach solaren Wert gesetzt wurde (schwarz). Zusätzlich sind die Bereiche eingezeichnet, in denen die Eisengruppenelemente die größten Auswirkungen haben. Unten: Das FUSE- und EUVE-Spektrum von RE 0503–289 verglichen mit einem auf den FUSE-Bereich normierten HHeCNO- (schwarz) und einem HHeCNOCaScTiVCrMnFeCoNi-Modell, wobei ein Gaussprofil der Halbwertsbreite  $2.0 \text{ \AA}$  an die Modelle und an die FUSE-Beobachtung angebracht wurde, um die Instrumentenauflösung von EUVE zu modellieren. Die kleine Abbildung zeigt denselben Bereich. Die Modelle sind mit einem Gaussprofil der Halbwertsbreite  $10 \text{ \AA}$  gefaltet.

rischen zu unterscheiden. Zusätzlich zu diesen Linien sind auch sprunghafte Absenkungen des Flussverlaufs möglich, die von gebunden-frei-Absorptionen stammen (Tabelle 4.13). Die *OWENS* zugrunde liegenden Atomdaten decken Linien im UV-Bereich (FUSE und IUE) ab, gebunden-frei-Absorption ist in diesem Bereich vernachlässigbar und wird von *OWENS* nicht berücksichtigt. Kurzweilig von  $911 \text{ \AA}$  (EUV-Bereich) sind die gebunden-frei-Absorptionen – vor allem von Wasserstoff und Helium – jedoch

Tabelle 4.13.: Grundzustandswellenlängen der von *TEUV* verwendeten Ionen.

Ion	Wellenlänge / Å
H I	911.75
He I	504.26
He II	227.32
C I	1101.07
C II	508.48
C III	258.91
N I	853.06
N II	418.85
N III	261.30
O I	910.44
O II	353.06
O III	225.69

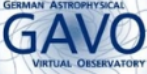
nicht zu vernachlässigen. Sie können eine deutliche Absenkung des theoretischen Flussverlaufs verursachen. Zur Korrektur der gebunden-frei-Absorptionen des interstellaren Mediums im EUV-Bereich können die Absorptionsquerschnitte von Wasserstoff und Helium zum Beispiel mit dem euv-Paket des Programms IRAF modelliert werden, das auf Rumph et al. (1994) basiert.

Außer Wasserstoff und Helium sind auch viele Metalle im interstellaren Medium zu finden, wie z.B. an den mit *OWENS* modellierten interstellaren Linien der vorhergehenden Kapitel zu sehen ist. In früheren Analysen wurde der Einfluss der Photoionisationsquerschnitte der Metalle auf maximal 10% abgeschätzt (Cruddace et al., 1974; Rumph et al., 1994), weshalb sie bei der Korrektur der interstellaren Absorption meist ignoriert wurden. Mit steigender Genauigkeit der Sternatmosphärenmodelle und der zugrunde liegenden Atomdaten muss auch die Modellierung interstellarer Korrekturen immer präziser werden. Eventuell können dadurch bisher ungeklärte spektrale Kennzeichen identifiziert oder die Diskrepanz zwischen theoretischem und beobachtetem Flussverlauf des DO Weißen Zwergs RE 0503–289 erklärt werden.

Im Rahmen dieser Arbeit wurde deshalb ein neues *VO*-Tool – genannt Tübingen EUV-Absorption (*TEUV*<sup>18</sup>)– entwickelt, das interstellare Absorption, verursacht durch RBF-Übergänge vom Grundzustand der Ionen H I, He I–II, C I–III, N I–III, O I–III, an das theoretische Spektrum anbringt. Grundlage dieses Analysewerkzeugs ist ein FORTRAN-Programm, das mit Hilfe von Unix-Shell-Skripten über eine Webseite gesteuert wird (Abb. 4.59). Dafür wurde eine Webseite erstellt (Abb. 4.58), auf der der Nutzer die Säulendichten der gewählten Ionen für zwei Komponenten des Interstellaren Mediums – die die heiße und kühle Komponente des ISM simulieren – sowie deren Temperaturen, Radial- und Turbulenzgeschwindigkeiten eingeben kann. Ein Upload für das Modell, an das die interstellare Korrektur angebracht werden soll, sowie der EUV-Beobachtung wurde ebenfalls in die Webseite eingebaut. Außerdem kann der Nutzer wählen, ob eine Faltung mit einem Gaussprofil angebracht werden soll, um eine Instrumentenauflösung zu simulieren. Zusätzlich ist eine Angabe des beobachteten Flusses bei 928 Å oder bei einer selbst gewählten EUV-Wellenlänge nötig.

Das Absenden des Requests startet das PHP-Skript `upload.php`, das die eingegebenen Parameter prüft und Modell und Beobachtung hochlädt. Danach wird mit der Prozedur `TEUV.sh` ein Verzeichnis erstellt, in das Modell und Beobachtung kopiert werden. `TEUV.sh` startet anschließend für die vom Nutzer gewählten Parameter sowie für eine um einen Faktor zehn variierte H I- und He I-Säulendichte die ausführbare Datei des FORTRAN-Programms (`TEUV1.Linux_x86`). Anschließend prüft `TEUV.sh`, ob bei der Ausführung des FORTRAN-Programms Fehler aufgetreten sind und benachrichtigt gege-

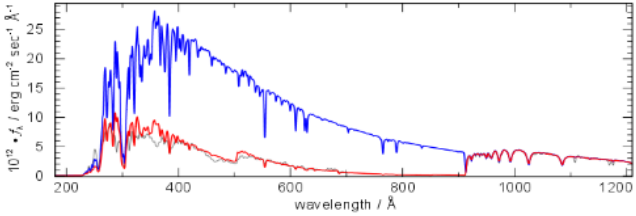
<sup>18</sup><http://astro.uni-tuebingen.de/~TEUV>



GERMAN ASTROPHYSICAL  
**GAVO**  
VIRTUAL OBSERVATORY

German Astrophysical Virtual Observatory

TEUV - Tübingen EUV Absorption - WWW Interface



$10^{12} \cdot f_{\lambda} / \text{erg cm}^{-2} \text{sec}^{-1} \text{Å}^{-1}$   
wavelength / Å

TEUV corrects synthetic stellar fluxes for interstellar absorption for  $\lambda < 911 \text{ Å}$ . It simulates radiative bound-free absorption of the ground states of H, He, C, N, and O using [Opacity Project](#) data. Two interstellar components with different radial ( $v_{\text{rad}}$ ) and turbulent velocities ( $v_{\text{turb}}$ ), temperatures ( $T_{\text{comp}}$ ), and column densities are considered. For further information hover the mouse over the respective labels.

**Personal Information**

Last Name

First Name

Email

Model

Observation

**Interstellar Components**

	Cool	Hot
$v_{\text{turb}}$	<input type="text" value="10"/>	<input type="text" value="30"/>
$v_{\text{rad}}$	<input type="text" value="10"/>	<input type="text" value="-10"/>
$T_{\text{comp}}$	<input type="text" value="100"/>	<input type="text" value="1000"/>

**Instrumental Resolution**

Resolution

**Observed flux**

1.0E-12 (928 Å)

at

**Column Densities / cm<sup>-2</sup>**

	Cool	Hot
H I	<input type="text" value="1.0E+18"/>	<input type="text" value="1.0E+18"/>
He I	<input type="text" value="1.0E+16"/>	<input type="text" value="1.0E+16"/>
He II	<input type="text" value="1.0E+16"/>	<input type="text" value="1.0E+16"/>
C I	<input type="text" value="1.0E+12"/>	<input type="text" value="1.0E+12"/>
C II	<input type="text" value="1.0E+12"/>	<input type="text" value="1.0E+12"/>
C III	<input type="text" value="1.0E+12"/>	<input type="text" value="1.0E+12"/>
N I	<input type="text" value="1.0E+12"/>	<input type="text" value="1.0E+12"/>
N II	<input type="text" value="1.0E+12"/>	<input type="text" value="1.0E+12"/>
N III	<input type="text" value="1.0E+12"/>	<input type="text" value="1.0E+12"/>
O I	<input type="text" value="1.0E+12"/>	<input type="text" value="1.0E+12"/>
O II	<input type="text" value="1.0E+12"/>	<input type="text" value="1.0E+12"/>
O III	<input type="text" value="1.0E+12"/>	<input type="text" value="1.0E+12"/>

This page was visited **10** times since December 18, 2012

[Please do not hesitate to contact us](#) in case of any question.

Abbildung 4.58.: TEUV-Webseite.

benenfalls den *TEUV*-Administrator. Vom Programm *TEUV1.Linux\_X86* wird eine Datei im json-Format erzeugt, die die Beobachtung, das originale Modell sowie mit der interstellaren Absorption korrigierte Modelle enthält (eins mit den vom Nutzer gewählten Säulendichten, eins für eine zehnfache H I-Säulendichte, eins für eine 0.1-fache H I-Säulendichte, eins für eine zehnfache He I-Säulendichte, eins für eine 0.1-fache He I-Säulendichte). Alle Modelle sind dabei auf den vom Nutzer angegebenen Fluss normiert und enthalten jeweils maximal 400 Punkte. Die Radialgeschwindigkeitskorrektur wurde ebenfalls an die Modelle angebracht, die Temperatur und Turbulenzgeschwindigkeit werden in diesem ersten Schritt ignoriert. Der Nutzer bekommt die json-Datei mit Hilfe von Javascript-Funktionen direkt auf der Webseite angezeigt (Abb. 4.60). Die variierten Modelle können interaktiv über die Auswahlbuttons dargestellt werden und geben einen Überblick, wie stark sich der Fluss bei einer Variation der H I- oder He I-Säulendichte ändert. Das nach den Parametern des Nutzers korrigierte Modell kann über den Button „Download your ISM\_choice“ heruntergeladen werden. Nach Auswahl dieses Buttons wird das PHP-Skript *download.php* gestartet, das dem Nutzer das bei Verwendung des Modells in Publikationen zu zitierende Acknowledgement anzeigt. Gleichzeitig führt die Webseite das FORTRAN-Programm *TEUV.Linux\_X86* aus, das die interstellare Korrektur an das hochgeladene Modell anbringt. Im Gegensatz zu *TEUV1.Linux\_X86* wird dabei keine Normierung angebracht, das Modell wird mit der Turbulenzgeschwindigkeit und Temperatur der Komponenten versehen und das komplette Modell in eine Textdatei geschrieben, die der Nutzer (mit Hilfe von *get.php*) speichern oder sich anzeigen lassen kann.

Die Funktionsweise des FORTRAN-Programms *TEUV.Linux\_X86* ist in Abb. 4.61 dargestellt. Das Programm liest zuerst die in der Webseite eingegebenen Parameter sowie das hochgeladene Modell ein.



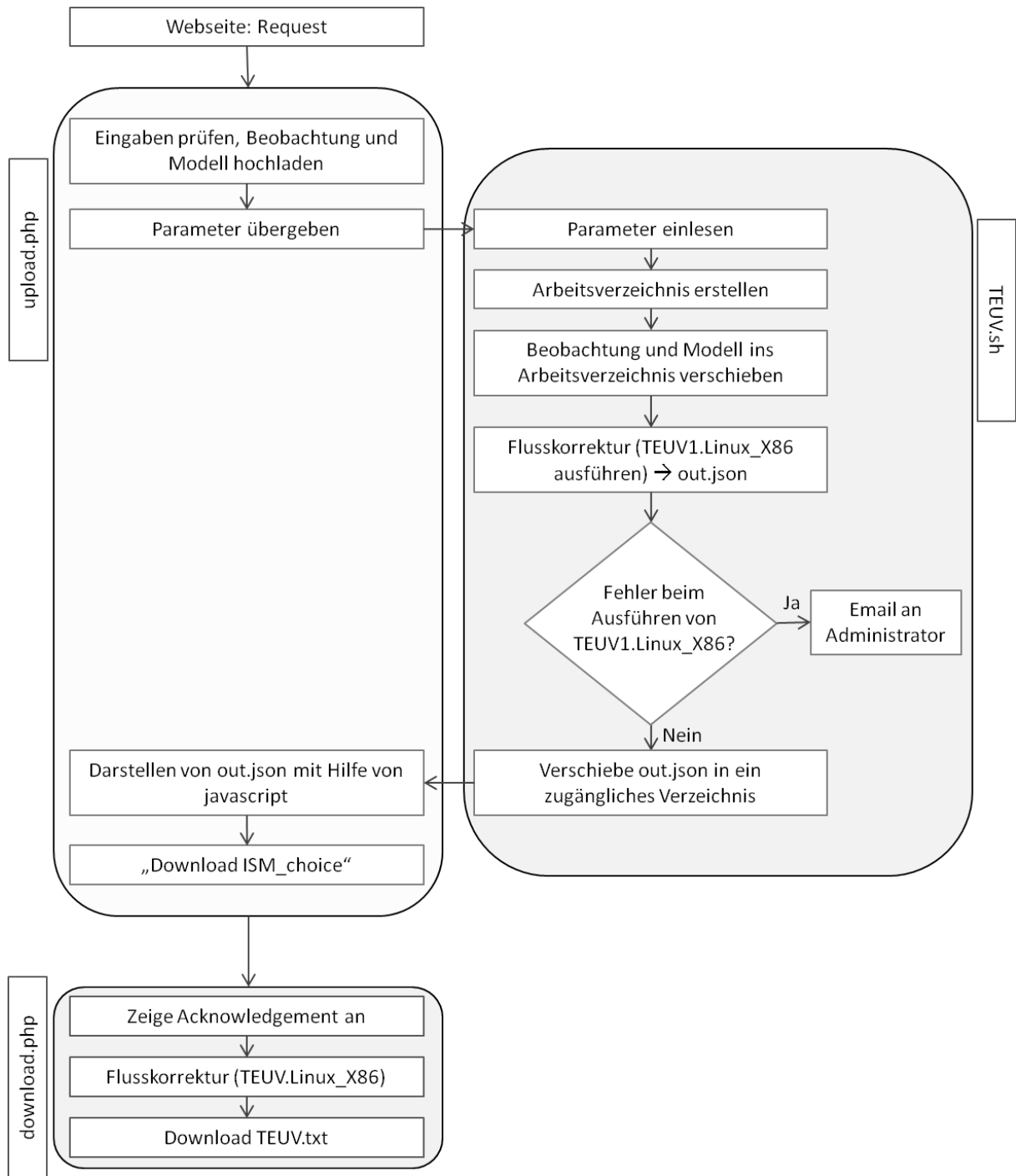


Abbildung 4.59.: Schematische Darstellung des neuen *TEUV*-Dienstes.

Dann werden die Opacity Project-Querschnitte der gewünschten Ionen eingelesen und bis zur Grundzustandswellenlänge extrapoliert, ein Dopplerprofil zur Korrektur der Temperatur und Turbulenzgeschwindigkeit angebracht sowie die resultierenden Querschnitte auf die eingelesenen Wellenlängen interpoliert. Anschließend werden die optischen Tiefen der heißen und kühlen Komponente berechnet, anhand der Radialgeschwindigkeit verschoben und wieder auf das ursprüngliche Wellenlängengitter interpoliert. Zum Schluss wird ein Gaussprofil mit der Halbwertsbreite der Instrumentenauflösung angebracht und das Resultat in eine Datei geschrieben.

Mit *TEUV* kann nun eine Absenkung des Flusses im EUV-Bereich, verursacht durch gebunden-frei-

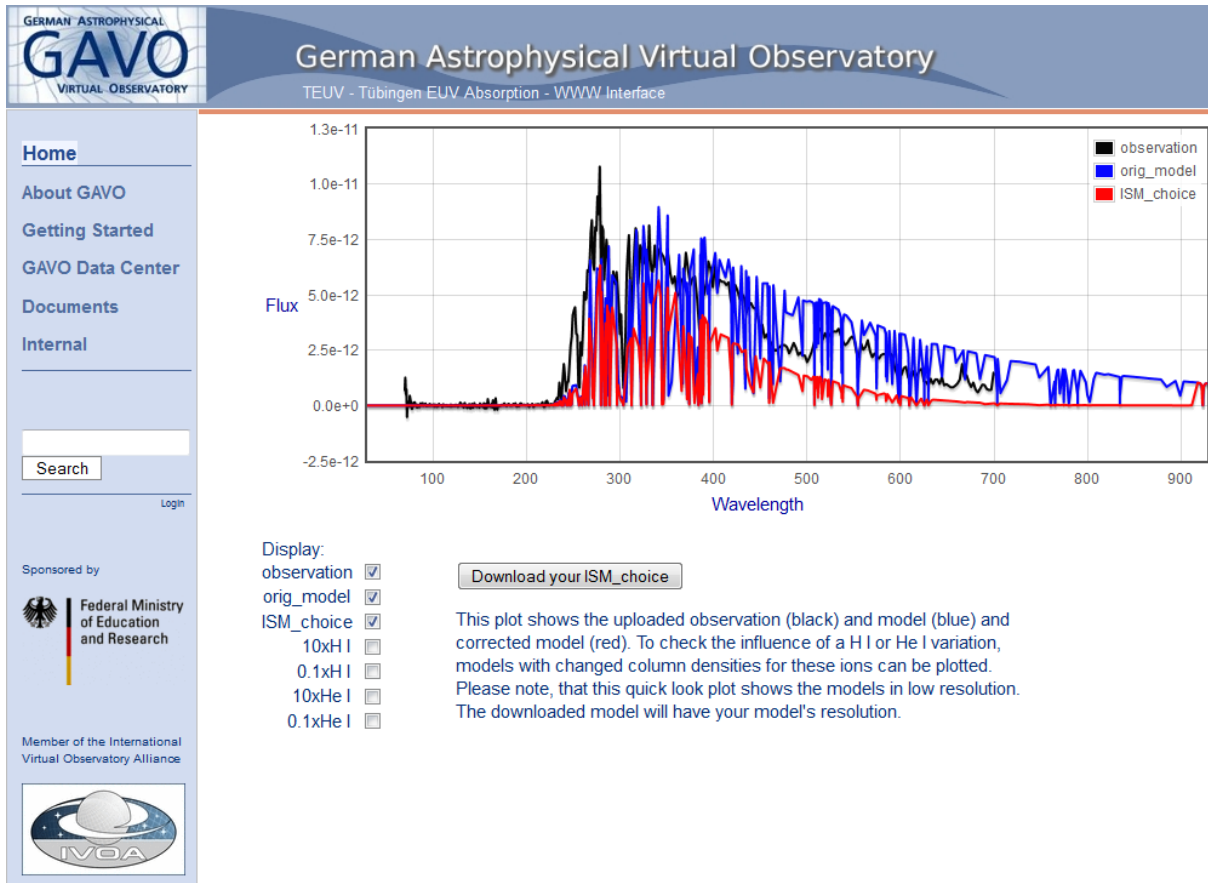


Abbildung 4.60.: Webseite des neuen *TEUV*-Dienstes nach Upload des Modells .

Absorptionsquerschnitte interstellarer Elemente, mit *VO*-Mitteln modelliert werden. Außerdem können die durch Resonanzen der BF-Absorptionsquerschnitte verursachten, linienähnlichen Komponenten des EUV-Flusses modelliert werden, was bei der Analyse hochaufgelöster EUV-Beobachtungen hilfreich sein kann. Die Bedienung dieses Tools ist einfach und intuitiv. Für weitere Hilfestellungen bei der Bedienung werden zusätzliche Erklärungen zu den einzelnen Eingabefeldern angezeigt, wenn der Nutzer mit dem Mauszeiger über die Bezeichnung der Eingabefelder fährt.

### Anwendung des neuen *TEUV*-Programms auf RE 0503–289

Als erste Anwendung dieses Analysewerkzeugs wird das beobachtete EUV-Spektrum des DO-Weißen Zwergs RE 0503–289 modelliert. Als Vorarbeit dafür wurden die interstellaren Linien der FUSE-Beobachtung von RE 0503–289 mit dem Programm *OWENS* simuliert. Die resultierenden Säulendichten der Elemente H, C, N und O sind in Tabelle 4.14 dargestellt. Bei der Modellierung wurde eine heiße Komponente des interstellaren Mediums mit 4000 K und einer Turbulenzgeschwindigkeit von 5 km/sec modelliert. Die Temperatur der interstellaren Komponente sowie die Säulendichte von Wasserstoff wurden mit dem WRPLOT-Programm, welches ebenfalls über eine Funktion zur Anbringung einer Korrektur der interstellaren Wasserstofflinien an das Modell verfügt, überprüft (Abb. 4.62). Die Werte beider Programme sind in Übereinstimmung (bis auf die Lyman  $\alpha$ - und Lyman  $\beta$ -Linie, die von *OWENS* unzureichend modelliert werden). Für die Bestimmung der Säulendichten mit *OWENS* ist anzumerken, dass manche der modellierten Ionen nur eine einzige Linie zeigen und diese beispielsweise für C III von einer photosphärischen überlagert ist. Die Werte sind dadurch mit einem großen Fehler behaftet und dienen hier nur zur Überprüfung der Größenordnung.

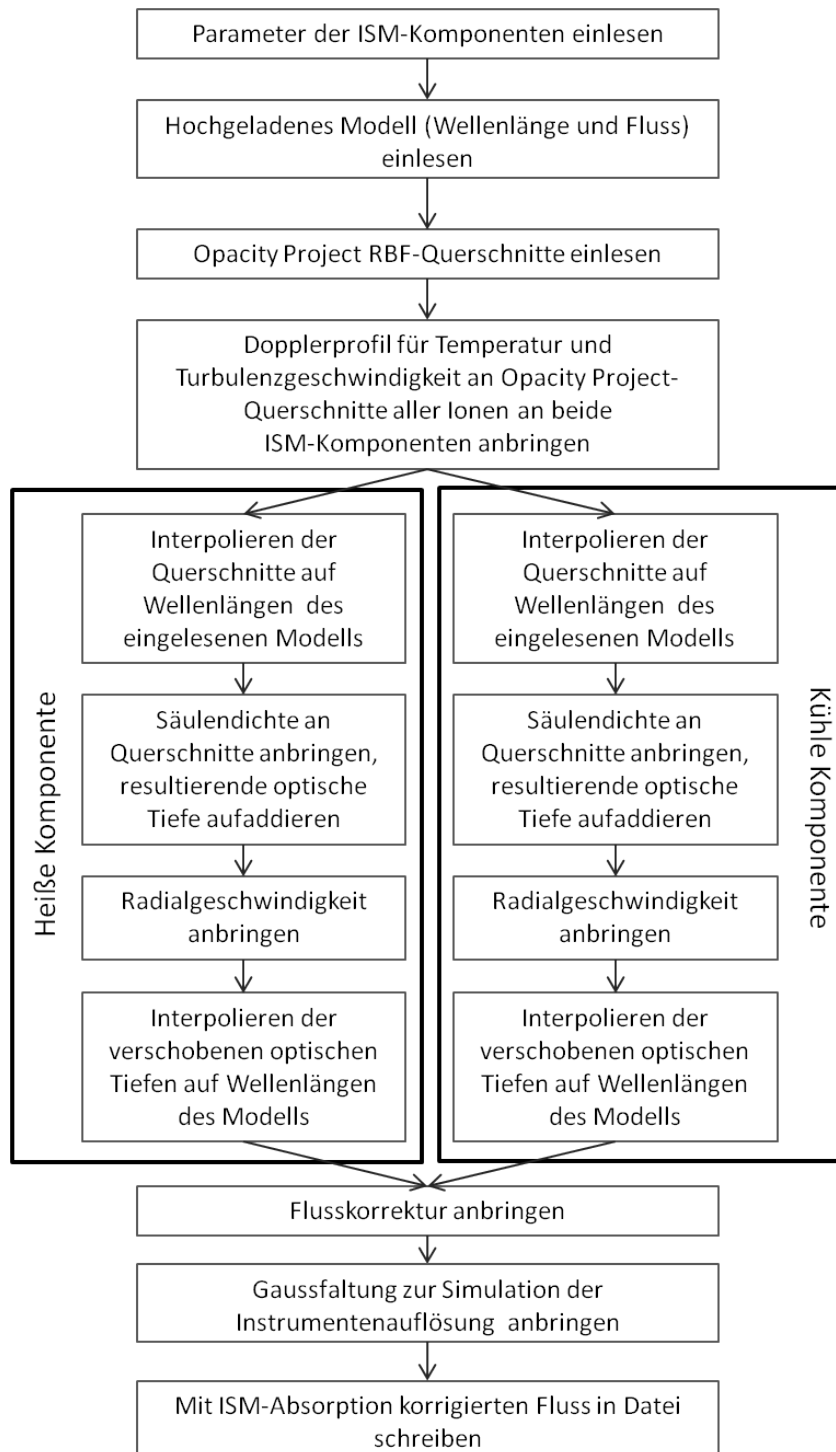


Abbildung 4.61.: Funktionsweise des neuen *TEUV*-FORTRAN-Programms.

Als weitere Vorarbeit wurde die interstellare Absorption mit dem *euv*-Paket des Programms *IRAF* modelliert. Dabei werden die  $\text{H I}$ -Säulendichte sowie das Verhältnis der  $\text{He I}$ - bzw.  $\text{He II}$ -Säulendichten zur  $\text{H I}$ -Säulendichte angegeben. Abb. 4.63 zeigt das Ergebnis. Das korrigierte Modell stimmt deutlich besser mit der Beobachtung überein. Die  $\text{H I}$ -Säulendichte ist gut getroffen, die  $\text{He I}$ -Säulendichte führt zu einer deutlich stärkeren Absenkung als beobachtet, was jedoch nötig ist, um den Flussverlauf kurzweilig davon besser zu treffen. Dennoch kann das Modell die Beobachtung nicht reproduzieren.

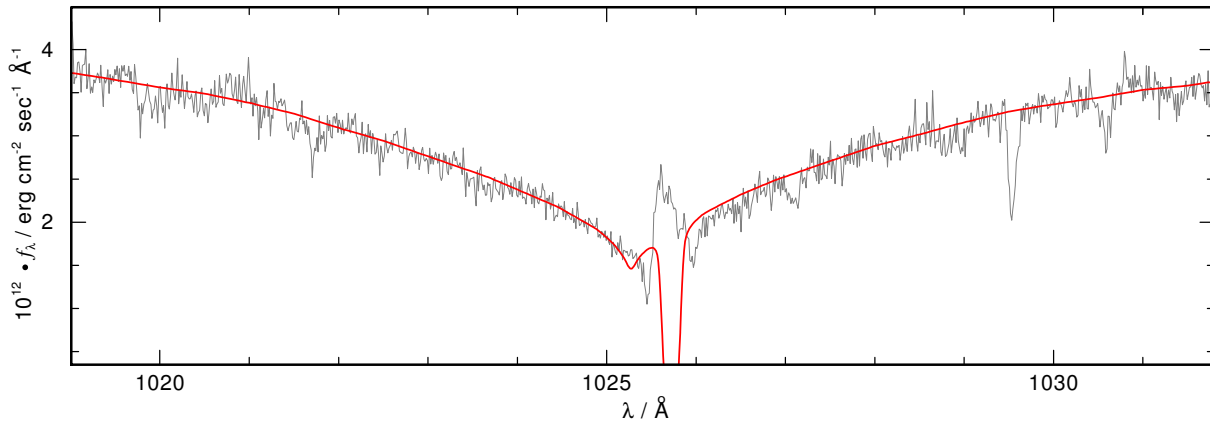


Abbildung 4.62.: Die Lyman  $\beta$ -Linie des HHeCNO-Modells von RE 0503–289, an das die interstellare Korrektur der Wasserstofflinien mit dem Programm WRPLOT angebracht wurde. Die Temperatur ist dabei 4000 K, die Turbulenzgeschwindigkeit 5 km/sec und die Säulendichte  $1.5 \times 10^{18} \text{ cm}^{-2}$ .

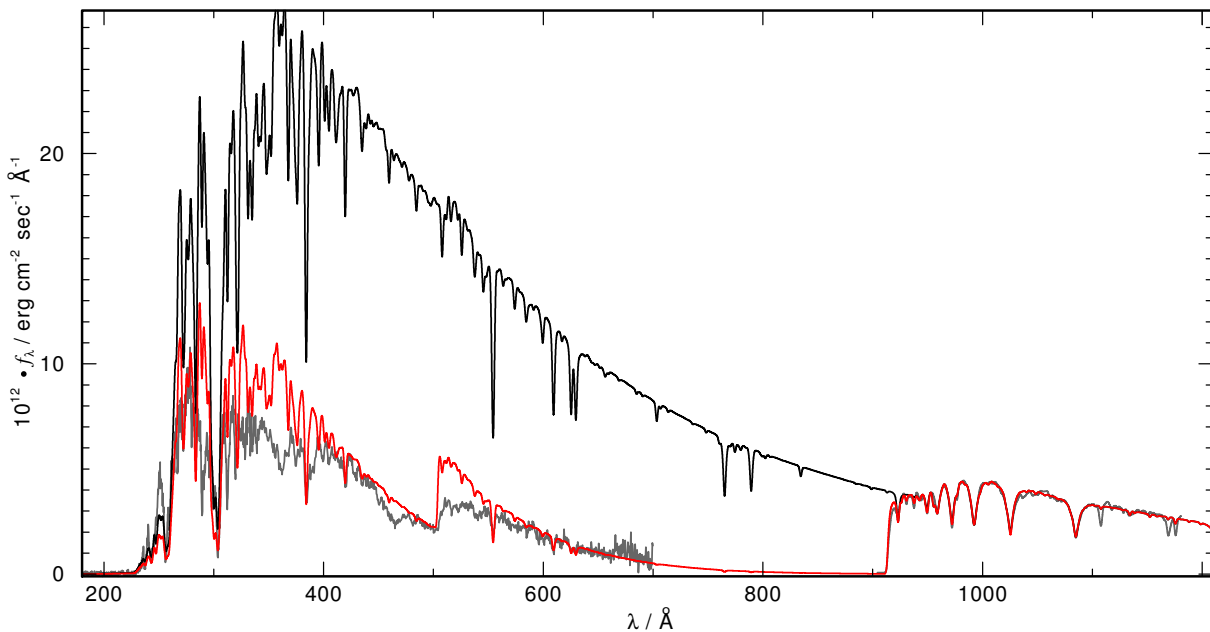


Abbildung 4.63.: Das FUSE- und EUVE-Spektrum von RE 0503–289 verglichen mit einem auf den FUSE-Bereich normierten HHeCNOCaScTiVCrMnFeCoNi-Modell (schwarz). Zusätzlich wurde an dieses Modell eine interstellare H I-Säulendichte von  $9 \times 10^{17}$  sowie He I- und He II-Säulendichten von  $1.3 \times 10^{17}$  angebracht (rot). Die Modelle und die FUSE-Beobachtung wurden mit einem Gaussprofil der Halbwertsbreite  $2.0 \text{ \AA}$  gefaltet.

Mit *TEUV* wurden die interstellaren Absorptionsquerschnitte der gebunden-frei Grundzustandsübergänge aller verfügbaren Ionen für RE 0503–289 modelliert. Der dem *TEUV*-Programm übergebene Fluss wurde zuvor im FUSE-Bereich um eine interstellare Komponente mit einer Temperatur von 4000 K und einer H I-Säulendichte von  $1.5 \times 10^{18} \text{ cm}^{-2}$  (ab  $\lambda = 910 \text{ \AA}$  zu höheren Wellenlängen hin) versehen, um einen realistischen, kontinuierlichen Übergang des Modells von höheren Wellenlängen zur Wasserstoffgrundzustandskante hin zu bekommen.

Werden mit *TEUV* die mit *OWENS* bestimmten Säulendichten angebracht, so fällt auf, dass die H I-Säulendichte bereits zu hoch ist und dass die He I-Grundzustandskante nicht modelliert wird (Abb. 4.64).

Tabelle 4.14.: Die mit *OWENS* bestimmten Säulendichten in  $\text{cm}^{-2}$  der Elemente des Interstellaren Mediums. Angenommen wurde dabei eine heiße Komponente des interstellaren Mediums mit einer Temperatur von 4000 K und einer Turbulenzgeschwindigkeit von 5 km/sec. Die Literaturverhältnisse wurden Cruddace et al. (1974) entnommen und auf die mit *OWENS* bestimmte Säulendichte von Wasserstoff angewandt.

Ion	<i>OWENS</i>	<i>OWENS</i> modifiziert	<i>WRPLOT</i>	<i>TEUV</i>	Literatur
H I	$1.50 \times 10^{18}$	$9.00 \times 10^{17}$	$1.50 \times 10^{18}$	$5.50 \times 10^{17}$	$9.00 \times 10^{17}$
He I		$1.30 \times 10^{17}$		$9.00 \times 10^{16}$	$7.50 \times 10^{16}$
He II		$1.30 \times 10^{17}$		$9.00 \times 10^{16}$	$7.50 \times 10^{16}$
C I	$9.02 \times 10^{12}$	$9.02 \times 10^{12}$		$9.00 \times 10^{12}$	$3.60 \times 10^{14}$
C II	$8.00 \times 10^{14}$	$8.00 \times 10^{14}$		$8.00 \times 10^{14}$	$3.60 \times 10^{14}$
C III	$7.00 \times 10^{13}$	$7.00 \times 10^{13}$		$7.00 \times 10^{13}$	$3.60 \times 10^{14}$
N I	$6.12 \times 10^{13}$	$6.12 \times 10^{13}$		$6.10 \times 10^{16}$	$1.00 \times 10^{14}$
N II	$2.92 \times 10^{13}$	$2.92 \times 10^{13}$		$2.90 \times 10^{13}$	$1.00 \times 10^{14}$
N III	$7.92 \times 10^{13}$	$7.92 \times 10^{13}$		$7.90 \times 10^{13}$	$1.00 \times 10^{14}$
O I	$1.62 \times 10^{14}$	$1.62 \times 10^{14}$		$1.60 \times 10^{14}$	$8.00 \times 10^{14}$
O II		$1.62 \times 10^{14}$		$1.60 \times 10^{14}$	$8.00 \times 10^{14}$
O III		$1.62 \times 10^{14}$		$1.60 \times 10^{14}$	$8.00 \times 10^{14}$

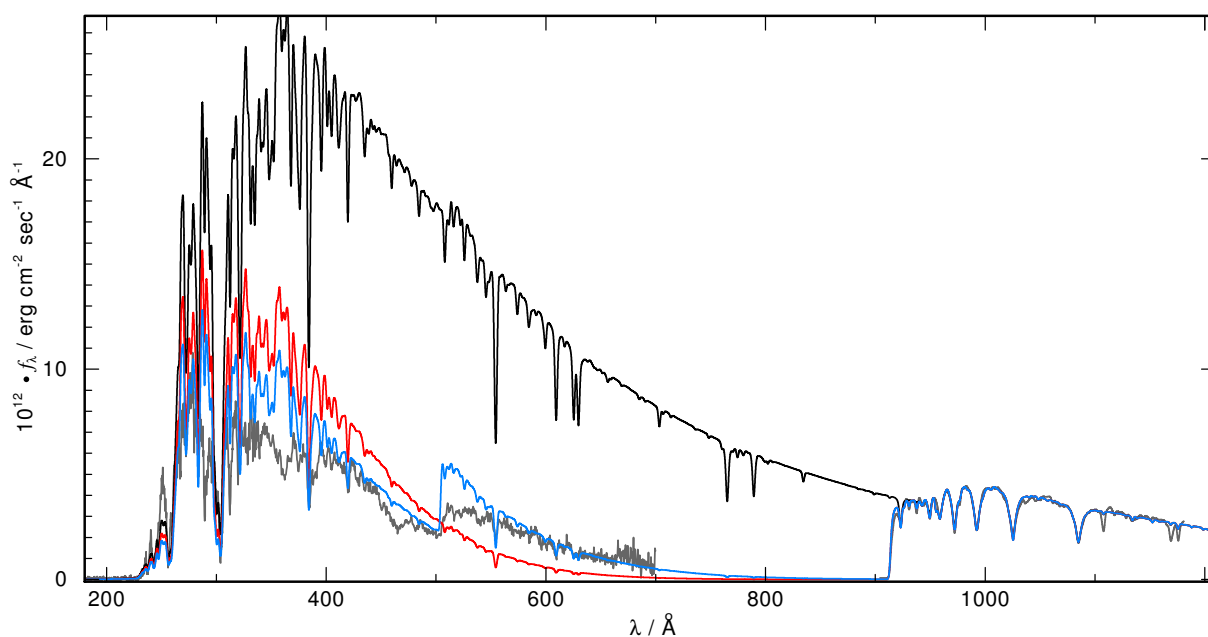


Abbildung 4.64.: Das FUSE- und EUVE-Spektrum von RE 0503–289 verglichen mit einem auf den FUSE-Bereich normierten HHeCNOCaScTiVCrMnFeCoNi-Modell (schwarz). Zusätzlich wurde dieses Modell mit einer interstellaren Komponente mit den Häufigkeiten der *OWENS*-Analyse (rot) sowie mit den modifizierten *OWENS*-Häufigkeiten (blau, Tabelle 4.14) versehen. An die Modelle und die FUSE-Beobachtung wurde ein Gaussprofil der Halbwertsbreite 2.0 Å angebracht.

Ursache dafür ist, dass wir mit *OWENS* im UV-Bereich keine Heliumsäulendichten bestimmen, weshalb diese für die erste Modellierung mit *TEUV* auf Null gesetzt ist. In einem weiteren Schritt wurde deshalb eine He-Säulendichte von  $1.3 \times 10^{17} \text{cm}^{-2}$  angebracht sowie die H I-Säulendichte reduziert, wodurch

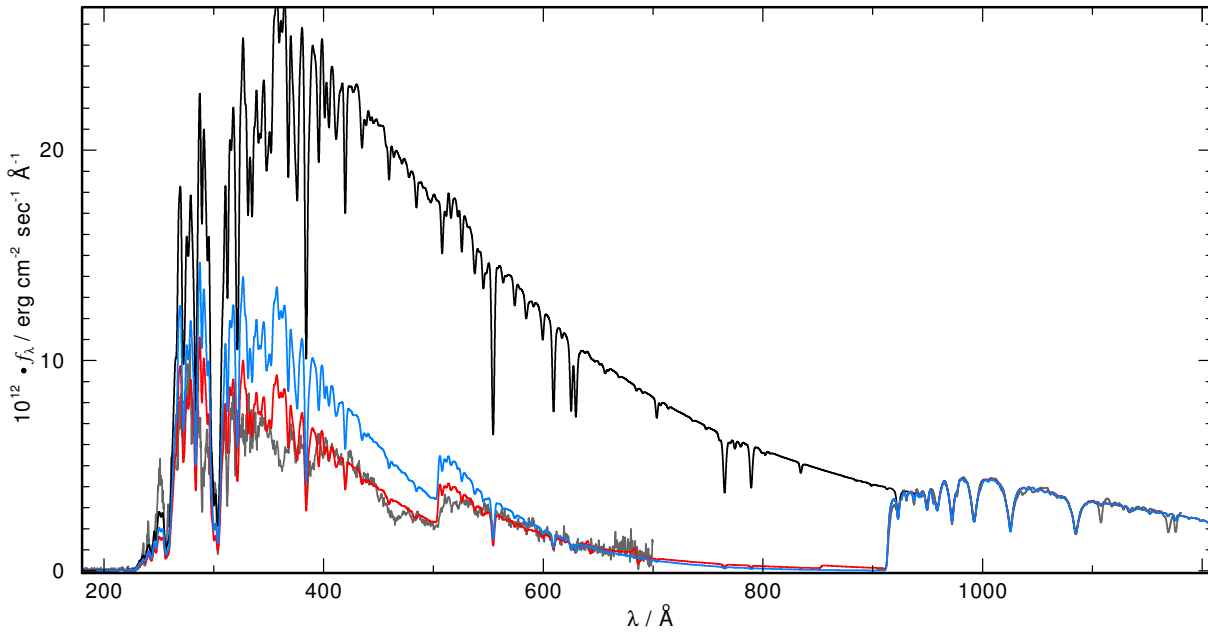


Abbildung 4.65.: Das FUSE- und EUVE-Spektrum von RE 0503–289 verglichen mit einem auf den FUSE-Bereich normierten HHeCNOCaScTiVCrMnFeCoNi-Modell (schwarz). Das Modell mit Häufigkeitsverhältnissen von Cruddace et al. (1974) (blau) sowie das Modell mit *TEUV*-Häufigkeiten (rot) ist ebenfalls dargestellt. An die Modelle und die FUSE-Beobachtung wurde ein Gaussprofil der Halbwertsbreite  $2.0 \text{ \AA}$  angebracht.

sich das Modell der Beobachtung annähert (Abb. 4.64, blau). Die He I-Säulendichte ist wie bei der Modellierung mit IRAF zu hoch, jedoch nötig um das Modell dem beobachteten Flussverlauf kurzweilig von  $530 \text{ \AA}$  anzunähern. Für O II und O III sind in *OWENS* keine Linien im FUSE-Bereich angegeben, die Werte der Säulendichten wurden deshalb auf den Wert von O I gesetzt. Die mit *TEUV* korrigierten Modelle mit (variierten) *OWENS*-Häufigkeiten reproduzieren den Flussverlauf deutlich besser als ohne Berücksichtigung der interstellaren Absorption, weichen jedoch kaum von den mit IRAF korrigierten Modellen ab.

Um den Bereich kurzweilig von etwa  $530 \text{ \AA}$ , der bei den korrigierten Modellen noch nicht flach genug ist, weiter abzusenken, wurde die Säulendichten der Elemente verändert. Einfluss auf den eben beschriebenen, noch nicht perfekt reproduzierten Bereich hat vor allem die N I-Säulendichte. Am besten reproduziert der Fluss mit den in Tab.4.14 angegebenen Werten für das *TEUV*-Programm den beobachteten Verlauf (Abb. 4.65). Für einen Vergleich mit in der Literatur angegebenen Häufigkeitsverhältnissen des interstellaren Mediums (Cruddace et al., 1974, Tabelle 4.14) wurden diese relativ zur H I-Säulendichte der mit *TEUV* bestimmten Werte modelliert. Durch die den *OWENS*-Werten ähnliche N I-Säulendichte ähnelt der Verlauf des Modells mit den Werten von Cruddace et al. (1974) dem mit den *OWENS*-Werten (Abb. 4.65), die He I-Säulendichte reproduziert jedoch die Höhe des beobachteten Sprungs besser.

Die Elemente und Ionen des interstellaren Mediums sind nicht homogen verteilt. Eine Variation der Säulendichten aller Elemente wird je nach Blickrichtung beobachtet. In der Sichtlinie zu einem Objekt sollten allerdings alle Programme zur Modellierung interstellaren Materials auf ähnliche Werte für die Säulendichten kommen. Für alle Ionen außer H I und N I konnten die mit *OWENS* bestimmten Werte bestätigt werden. Werden sie mit *TEUV* modelliert, stehen sie nicht im Widerspruch zur Beobachtung. Der mit *TEUV* bestimmte Wert von H I ist zwar etwas niedriger als die beiden durch *OWENS* und *WRPLOT* erhaltenen Werte, stimmt aber innerhalb der Fehlergrenzen mit diesen überein. Um die Beobachtung zu reproduzieren, muss die N I-Häufigkeit aber um ca. einen Faktor 1000 erhöht werden. Dies steht im klaren Widerspruch zum mit *OWENS* bestimmten Wert und ist unrealistisch. Das Verhältnis der N I- zur

HI-Säulendichte liegt für *TEUV* bei 0.089, für *OWENS* bei  $4.08 \times 10^{-5}$ . Dieses *OWENS*-Verhältnis ist ca. solar ( $6.76 \times 10^{-5}$ ) und liegt näher am Literaturwert (Verhältnis zu Wasserstoff) von Cruddace et al. (1974) ( $1.12 \times 10^{-4}$ ) als der *TEUV*-Wert. Sowohl Cruddace et al. (1974) als auch Rumph et al. (1994) geben an, dass die Metalle maximal 10% der Gesamtopazität ausmachen, was mit dieser Analyse bestätigt werden kann. Die Literaturannahme, dass Helium ca. 10% der Zusammensetzung ausmacht wird von *TEUV* mit *OWENS*-Werten (Verhältnis zu H  $1.4 \times 10^{-1}$ ) und mit eigens bestimmten Werten ( $3.4 \times 10^{-1}$ ) etwas übertroffen. Diese Verhältnisse sind etwas supersolar (solar wäre  $8.51 \times 10^{-2}$ ). Das größere Verhältnis des mit *TEUV* bestimmten Heliumwerts resultiert aus der Verringerung der Wasserstoffsäulendichte. Damit ist sie kleiner als der Literaturwert aus früheren Analysen von RE 0503–289 (maximal  $10^{18} \text{cm}^{-2}$ ), wird jedoch durch die erhöhten NI-Säulendichte ausgeglichen.

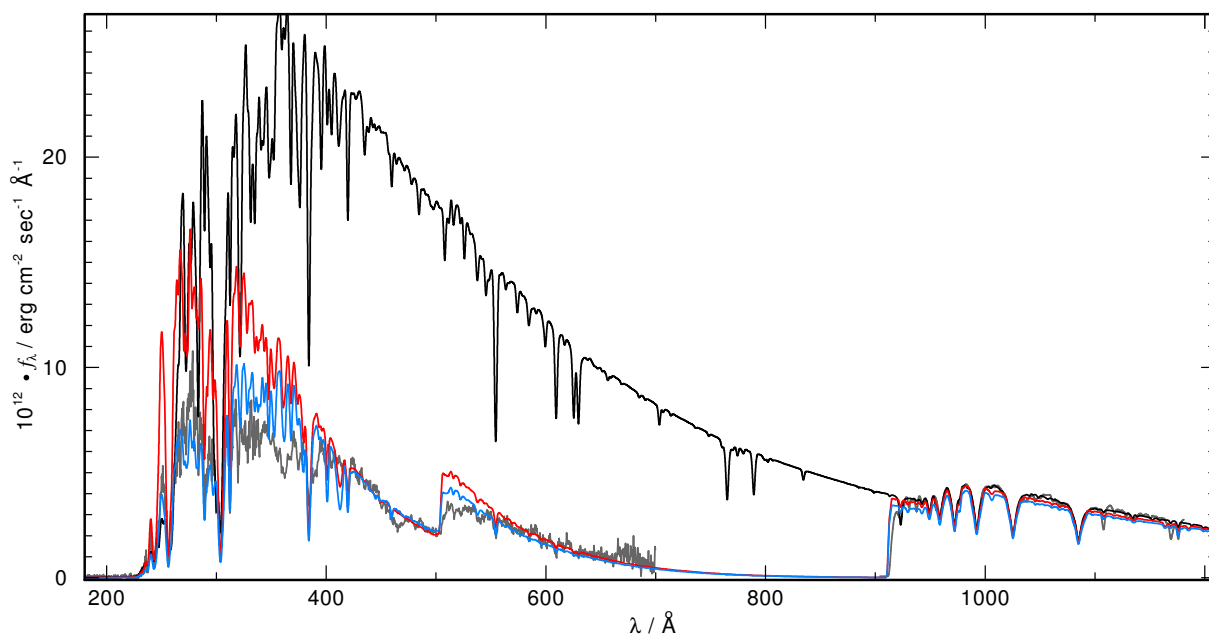


Abbildung 4.66.: Das FUSE- und EUVE-Spektrum von RE 0503–289 verglichen mit auf den FUSE-Bereich normierten Modellen aus der Literatur (siehe Text, schwarz). Die Modelle wurden mit *TEUV* korrigiert, wobei eine Säulendichte von  $1.3 \times 10^{17} \text{cm}^{-2}$  für He I (rot,  $T_{\text{eff}} = 70\,000 \text{K}$ ) bzw.  $9.0 \times 10^{16} \text{cm}^{-2}$  für He I (blau,  $T_{\text{eff}} = 65\,000 \text{K}$ ) und die mit *OWENS* bestimmten Häufigkeiten für die anderen Ionen angenommen wurden. An die Modelle und die FUSE-Beobachtung wurde ein Gaussprofil der Halbwertsbreite  $2.0 \text{Å}$  angebracht.

Insgesamt wird bestätigt, dass die interstellare Materie einen großen Einfluss auf den Fluss im EUV-Bereich hat. Durch die neu modellierten Ionen kann der Fluss zusätzlich abgesenkt werden und nähert sich so der Beobachtung an. Die dabei angenommenen Säulendichten können jedoch nicht durch die Ergebnisse der Analyse des FUSE-Bereichs bestätigt werden und sind unrealistisch. Die zusätzlichen gebunden-frei-Übergänge von C, N, und O haben also im Vergleich zu H und He – wie bereits von Cruddace et al. (1974) und Rumph et al. (1994) festgestellt – eine untergeordnete Rolle und helfen hier nicht weiter. Die fehlende Abflachung des Flusses wird deshalb vermutlich durch weitere Elemente der Photosphäre verursacht, die bisher noch nicht identifiziert bzw. noch nicht ausreichend modelliert werden können. Dafür spricht zum einen, dass bereits schwerere Elemente als die Eisengruppenelemente mit supersolaren Häufigkeiten identifiziert werden konnten, und zum anderen, dass das beobachtete Spektrum weitere Absorptionskennzeichen zeigt (beispielsweise bei  $365 \text{Å}$  oder bei  $465 \text{Å}$ ), die bei keiner bekannten Wellenlänge einer gebunden-frei-Absorption irgendeines Elements liegen. Abb. 4.66 zeigt die Modelle von Rauch et al. (2012); Werner et al. (2012b). Dabei wurden die Elemente He (mit einer Häu-

figkeit von  $9.531\text{E}-01$  in Massenbruchteilen), C ( $4.434\text{E}-02$ ), N ( $1.695\text{E}-04$ ), O ( $1.925\text{E}-03$ ), Ca (generisch,  $9.779\text{E}-07$ ), Fe ( $1.273\text{E}-05$ ), Ni ( $1.478\text{E}-04$ ), Ge ( $1.556\text{E}-04$ ), Kr ( $4.942\text{E}-05$ ) und Xe ( $6.164\text{E}-05$ ) modelliert. Durch die Berücksichtigung der schweren Elemente dieser Analyse kann der Fluss noch nicht reproduziert werden. Wie in den vorherigen Analysen reproduziert das Modell mit kleinerer  $T_{\text{eff}}$  die Beobachtung besser, zeigt jedoch im Bereich um etwa  $280 \text{ \AA}$  eine zu starke Abflachung.



## 5. Zusammenfassung und Diskussion

### 5.1. Die neuen und weiterentwickelten Tübinger VO-Dienste

Im Rahmen des *GAVO*-Projekts wurden in Tübingen Dienste entwickelt, die unterschiedlichen Nutzergemeinden Spektralanalysen über das Internet ermöglichen. Der Dienst *TheoSSA* bietet direkten Zugriff auf bereits berechnete synthetische Spektren. Für detailliertere Analysen kann der Nutzer Spektren mit individuell gewählten Parametern über den *TMAW*-Dienst automatisch berechnen lassen. Fortgeschrittene Nutzer können Modellatome aus der Datenbank *TMAD* herunterladen und für alle Modellatmosphärenprogramme benutzen. Momentan basieren diese Dienste auf dem Tübinger Modellatmosphärencode *TMAP* für heiße, kompakte Objekte, können aber um andere Modellatmosphärencodes ergänzt werden. Diese Arbeit beschäftigt sich mit der Erweiterung der *TMAD*- und *TMAW*-Dienste. Zusätzlich wurde ein neuer Dienst (*TIRO*) entwickelt, mit dem Modellatome und gebunden-gebunden- und gebunden-frei-Querschnitte für Eisengruppenelemente dem *VO*-Nutzer zur Verfügung gestellt werden. Dieser Dienst basiert auf dem bereits bestehenden Programm *IrOnIc*, das vor der eigentlichen Modellatmosphärenberechnung ausgeführt wird. Er ermöglicht noch detailliertere und präzisere Spektralanalysen über das *VO*. Die Erweiterungen und Neuentwicklungen der Tübinger Dienste werden in diesem Kapitel zusammengefasst.

#### Atomdaten

Wichtiger Input für alle Modellatmosphärenprogramme sind Atomdaten. Für präzise Spektralanalysen sind aktuelle und möglichst vollständige Atomdaten von fundamentaler Bedeutung. Im Rahmen dieser Arbeit wurde deshalb sowohl die *TMAD*-Datenbank erweitert, als auch ein Teil der *IrOnIc* zugrunde liegenden, von Kurucz erweiterten Eisengruppenatomdaten aktualisiert. Damit stehen nun zusätzliche Levels und Linien zur Verfügung, die die Modelle realistischer machen. Die neuen *TMAD*-Ionen können außerdem für Häufigkeitsbestimmungen weiterer Elemente verwendet werden, was z.B. Hinweise auf die Fusionsprozesse in Sternen während früherer Entwicklungsphasen liefern kann.

#### *TMAW*

Für die Berechnung individueller, synthetischer Spektren kann der *TMAW*-Dienst verwendet werden. Mit ihm können Modellatmosphären, die die Opazitäten der Elemente H, He, C, N und O berücksichtigen, automatisch berechnet werden. Im Rahmen der Einführung einer Qualitätskontrolle wurde ein erster Benchmarktest für theoretische Daten eingeführt, in dem ein großes Gitter von (527) *TMAW*-Modellen berechnet und zu einem Vergleich mit der *TMAP*-Analyse des sdOB-Primärsterns von AA Doradus von Klepp & Rauch (2011) herangezogen wurde. Dieser Vergleich liefert wichtige Erkenntnisse über die Genauigkeit von *TMAW*-Modellen, die, verglichen mit *TMAP*-Modellen, deutlich weniger Iterationen durchlaufen haben und auf kleineren Modellatomen mit weniger Elementen basieren. Aufgrund dieses Tests wurde *TMAW* um zusätzliche Berechnungsschritte erweitert, in denen eine exaktere Temperaturkorrektur sowie eine Lineinformationrechnung mit größeren Modellatomen durchgeführt wird. Durch diese Nachberechnungsschritte konnte die Genauigkeit von *TMAW*- im Vergleich zu *TMAP*-Analysen deutlich gesteigert werden. Aufgrund der größeren Modellatome können mit *TMAW* nun mehr Linien im optischen Wellenlängenbereich modelliert werden. Um die Bedienung von *TMAW* auch dem Administrator zu erleichtern, wurden Prozeduren entwickelt, die ihn automatisch über Fehler informieren und

eine schnellere Bearbeitung und Kontrolle der vom *VO*-Nutzer gestellten Requests ermöglichen.

Große Gitter von *TMAW*-Requests, die viele Modelle beinhalten, können zur Entlastung des eigenen (IAAT) Rechenclusters auf Computern, die über das *AstroGrid-D* zugänglich sind, berechnet werden. Die dafür nötigen Skripte und Prozeduren wurden erweitert und verbessert. Zusätzlich wurden auch für die Berechnung auf über das *AstroGrid-D* zugänglichen Rechnern Prozeduren eingeführt, die die Kontrolle durch den *TMAW*-Administrator vereinfachen.

### ***IrOnIc* und *TIRO***

Gebunden-gebunden- und gebunden-frei-Querschnitte und Modellatome für Eisengruppenelemente werden von der Tübinger Arbeitsgruppe „Optische und UV-Astronomie“ mit dem selbst entwickelten *IrOnIc*-Programm erzeugt. Um diese dem *VO*-Nutzer zugänglich zu machen, wurde der *TIRO*-Dienst entwickelt. Der Nutzer gibt auf der zugehörigen Webseite die gewünschten Parameter ein und erhält die Modellatome und Querschnitte nach deren automatischer Berechnung oder durch Interpolation zwischen Werten einer speziell für diesen Zweck aufgebauten Datenbank. Im Vergleich zur Bedienung der ursprünglichen *IrOnIc*-Version können weniger Parameter gewählt werden. Die Energiebänder, in die die realen Levels aufgrund des verwendeten statistischen Verfahrens einsortiert werden, werden nun logarithmisch äquidistant gesetzt. Die Ionisationsenergie eines generischen Ions wird realistischer gewählt und entspricht nun der maximalen Ionisationsenergie aller berücksichtigten Elemente. Die neuen Modellatome können die Ionisationsstufe I zusätzlich berücksichtigen, was den *TIRO*-Dienst um die Nutzung für kühle Objekte erweitert. Eine Datenbank mit Querschnitten für individuelle und generische Modellatome wurde mit Hilfe eines sehr feinen Frequenzgitters (1 000 000 Punkte) aufgebaut, wodurch die eben erwähnte Interpolation möglich ist. Sollte dennoch eine neue Berechnung der Querschnitte nötig sein, kann diese auf einem hochgeladenen oder nach Nutzervorgaben automatisch erzeugten Frequenzgitter erfolgen. Die neue Bedienung des *IrOnIc*-Programms über den *TIRO*-Dienst stellt für alle Nutzer (vor allem die der eigenen Arbeitsgruppe) eine deutliche Vereinfachung dar. Während früher für jedes Element Input-Dateien angepasst und jeweils drei Programmdurchläufe durchgeführt werden mussten, ist nun nur noch die Eingabe der gewünschten Parameter auf der Webseite erforderlich. Auch die lange Rechenzeit von bis zu mehreren Tagen wurde (neben der nun möglichen Interpolation der Querschnitte) verkürzt. Eine Beschleunigung der Berechnung wurde zum einen durch eine parallele Behandlung der einzelnen Ionen, zum anderen durch den Test verschiedener Voigttroutinen – dem Flaschenhals der Querschnittsberechnung – und dem Implementieren der schnellsten erreicht. Der neue Dienst kann ohne Hintergrundkenntnisse benutzt werden und liefert dem *VO*-Nutzer realistische Modellatome und Querschnitte ohne selbst Rechenzeit für deren Erzeugung investieren zu müssen.

## **5.2. Anwendung der Dienste in der Spektralanalyse**

Der weiterentwickelte *TMAW*-Dienst wurde nicht nur für den Vergleich mit der Analyse von AA Doradus getestet. Um dem *VO*-Nutzer die Genauigkeit von *TMAW*- im Vergleich zu *TMAP*-Rechnungen zu verdeutlichen und so zu überprüfen, inwiefern *TMAW* für zuverlässige Spektralanalysen geeignet ist, wurden die Helium-sdOs LSE 153 und LSE 263 sowie die PG 1159-Sterne PG 1707+427 und PG 1424+535 analysiert und die resultierenden Modelle mit *TMAW* nachgerechnet und verglichen. Als Anwendung für das *IrOnIc*-Programm und Motivation für den neu entwickelten *TIRO*-Dienst wurden zusätzlich Eisengruppenelemente in die Analysen dieser Objekte aufgenommen und ihre Auswirkungen auf das Ergebnis der Analysen untersucht. Zusätzlich wurde der Effekt der Eisengruppenelemente auf den Flussverlauf von sd(O)B-Sternen und eines DO-Weißen Zwergs untersucht. Diese Objekte zeigen einen beobachteten UV- bzw. EUV-Flussverlauf, der bisher nicht erklärt werden konnte und für dessen Ursache Eisengruppenelemente vermutet wurden. Aufgrund der Untersuchung des DO-Weißen Zwergs wurde außer-

dem ein neuer VO-Dienst (*TEUV*) entwickelt, mit dem interstellare Absorption im Wellenlängenbereich  $\lambda < 911 \text{ \AA}$  modelliert werden kann. Diese Analysen und Entwicklungen werden nun kurz zusammengefasst.

### **LSE 153, LSE 263, PG 1707+427 und PG 1424+535**

Für die beiden He-sdO-Sterne LSE 153 und LSE 263 wurden zuerst HHe-Modelle mit Literaturhäufigkeiten berechnet und mit der optischen Beobachtung verglichen. Anhand dieser Modelle konnte geschlossen werden, dass die Effektivtemperatur der beiden Objekte niedriger als die Literaturwerte sein müssen. Die erzeugten Modelle wurden ebenfalls über den *TMAW*-Dienst angefordert. Bei der folgenden Berechnung zeigte sich, dass nicht alle Modelle in diesem Parameterbereich mit dem automatischen Verfahren ein Ergebnis liefern. Tests zur Überprüfung einer möglichen Verbesserung des *TMAW*-Dienstes zeigen, dass eine Verbesserung nur mit einer deutlich höheren Rechenzeit erreicht werden kann, was bei einem Einbau in das Standardverfahren zu lange Wartezeiten verursachen würde.

Aufgrund des in beiden Objekten beobachteten Balmerlinienproblems wurden im Rahmen der *TMAP*-Analyse zusätzlich die Elemente C, N und O eingefügt und die Parameter angepasst. Ein Vergleich des mit den finalen Parametern berechneten *TMAW*-Modells mit den *TMAP*-Modellen, die die Fehlergrenzen repräsentieren, zeigt, dass eine *TMAW*-Analyse der optischen und UV-Spektren zu einem Ergebnis führt, das weniger als 4% bzw. 8% in  $T_{\text{eff}}$  (60 000 K für LSE 153 und LSE 263) und als ca. 5% in  $\log g$  (5.0 für LSE 153 und 5.25 für LSE 263) vom *TMAP*-Ergebnis abweicht.

In die finalen *TMAP*-Modelle wurden zusätzlich die Eisengruppenelemente eingefügt (Fe und Ni individuell, Ca, Sc, Ti, V, Cr, Mn und Co generisch). Für LSE 153 stehen die stärksten Fe- und Ni-Linien des Modells mit solaren Häufigkeiten nicht im Widerspruch zur Beobachtung. Für LSE 263 sind die Häufigkeiten solar bis zehnfach solar. Im Vergleich zum HHeCNO-Modell zeigt das Modell mit Eisengruppenelementen eine höhere Temperatur im Linienentstehungsgebiet. Die Berücksichtigung der Eisengruppenelemente in der Analyse führt zu einem Ergebnis, das noch innerhalb der Fehlergrenzen (4% bzw. 8% in  $T_{\text{eff}}$  und ca. 5% in  $\log g$ ) liegt, aber stärker vom *TMAP*-HHeCNO-Modell abweicht als ein mit *TMAW* berechnetes Modell.

Für die beiden PG 1159-Sterne PG 1424+535 ( $T_{\text{eff}} = 115\,000 \text{ K}$ ,  $\log g = 7.3$ ) und PG 1707+427 ( $T_{\text{eff}} = 85\,000 \text{ K}$ ,  $\log g = 7.5$ ) wurde eine vollständige Analyse mit den Elementen H, He, C, N, O, F, Ne, Si, P, S, Ar, Fe und Ni sowie Ca-Co als generisches Modellatom durchgeführt. Begonnen wurde mit HHeCNO-Modellen für die Bestimmung von  $T_{\text{eff}}$  und  $\log g$ . Bei diesen Objekten macht Neon einen großen Bestandteil der Atmosphäre aus, was sich ebenfalls auf deren Struktur auswirkt. Der Vergleich mit den *TMAW*-HHeCNO-Modellen, berechnet mit den finalen Parametern der HHeCNO-Analyse zeigt, dass eine *TMAW*-Analyse zu Ergebnissen innerhalb der Fehlergrenzen (4% bzw. 12% für  $T_{\text{eff}}$  und 4% bzw. 7% für  $\log g$ ) führt. Der Vergleich der HHeCNO-Modelle mit dem Modell mit allen Elementen (H, He, C, N, O, F, Ne, Si, P, S, Ar, Fe, Ni, Ca-Co) zeigt, dass eine Vernachlässigung der zusätzlichen Elemente zu Abweichungen von maximal 4% bzw. 12% in  $T_{\text{eff}}$  und 4% bzw. 7% in  $\log g$  führt. Für beide Objekte sind die Modelle mit solaren Eisen- und Nickelhäufigkeiten mit der Beobachtung konsistent.

### **sd(O)B-Sterne**

Die ersten vier sd(O)B-Sterne, bei denen ungewöhnlich hohe Eisengruppenelementhäufigkeiten gefunden wurden, zeigen – falls eine Beobachtung vorhanden ist – einen UV-Flussverlauf, der ungewöhnlich flach ist. Dieses Phänomen nimmt mit sinkender Effektivtemperatur ab. Um die Vermutung zu prüfen, dass dieser Verlauf durch Eisengruppenelemente verursacht wird, wurden *TMAP*-Modelle mit individuell erhöhten und später Kombinationen von erhöhten Eisengruppenhäufigkeiten berechnet. Es zeigt sich, dass für die heißesten Objekte EC 11481–2303 ( $T_{\text{eff}} = 55\,000 \text{ K}$ ,  $\log g = 5.8$ ) und PG 0909+276 ( $T_{\text{eff}} = 36\,900 \text{ K}$ ,  $\log g = 5.9$ ) eine deutlich erhöhte Fe- und Ni- bzw. Co-, Cr-, Mn- und Ni-Häufigkeit den beobachteten Verlauf qualitativ reproduzieren kann. Für PG 0909+276 sind die Literaturhäufigkei-

ten dieser Elemente jedoch zu hoch. Das kühlfte Objekt, UVO 1758+36 ( $T_{\text{eff}} = 32\,000\text{ K}$ ,  $\log g = 5.8$ ), zeigt nur eine schwache Einsenkung des beobachteten Flussverlaufs. Für dieses Objekt können die supersolaren Häufigkeiten aus der Literatur nicht widerlegt werden, da sie keine flussabsenkende Wirkung im UV-Bereich zeigen. Insgesamt nehmen die Ni-, Mn- und Fe-Häufigkeiten für die untersuchten Objekte mit der Effektivtemperatur ab. Die Kobalthäufigkeit nimmt mit sinkender Effektivtemperatur erst zu, dann wieder ab.

Die in sdB-Sternen beobachteten Häufigkeiten lassen sich im Allgemeinen durch Diffusion erklären. Die für EC 11481–2303, PG 0909+276 und UVO 1758+36 mit dem *NGRT*-Programm berechneten Diffusionsmodelle bestätigen zwar die supersolaren Häufigkeiten und den Trend abnehmender Häufigkeiten für kühlere  $T_{\text{eff}}$ , können den Flussverlauf jedoch nicht so gut wie das chemisch homogene *TMAP*-Modell repräsentieren.

Eine genauere Betrachtung der Linienprofile von EC 11481–2303 im UV-Bereich zeigt, dass die Unsicherheiten in den Atomdaten zu groß und die spektrale Auflösung der vorhandenen Beobachtungen (selbst die des FUSE-Spektrums) zu gering sind, um eine detaillierte Häufigkeitsbestimmung durchzuführen. Trotzdem stammen bei EC 11481–2303 die meisten Linien mit großer Wahrscheinlichkeit von Eisengruppenelementen, nur wenige haben einen interstellaren Ursprung.

Im optischen Wellenlängenbereich hat die Berücksichtigung der Eisengruppenelemente (mit supersolaren Häufigkeiten) deutliche Auswirkungen auf die Linienprofile der Balmerlinien. Diese Auswirkungen werden schwächer, je kleiner die Häufigkeiten im IUE-Bereich flussabsenkender Eisengruppenelemente ist. Sie nehmen somit für diese Objekte mit der Effektivtemperatur ab, da die Häufigkeiten mit der Effektivtemperatur abnehmen. Da in einigen sdB-Sternen das Balmerlinienproblem beobachtet werden kann, könnten die Eisengruppenelemente für die Modellierung dieser Linien ebenfalls von fundamentaler Bedeutung sein.

Insgesamt wird bestätigt, dass Eisengruppenelemente bei der Analyse von sdB-Sternen eine fundamentale Rolle spielen. Die Genauigkeit der Häufigkeitsbestimmung dieser Elemente wird durch zwei Faktoren limitiert. Zum einen wurden LIN-Atomdaten von Kurucz verwendet, die große Unsicherheiten innehaben. Zum anderen sind die vorhandenen Beobachtungen nicht genügend aufgelöst, um einzelne Linien zuzuordnen. Mit höher aufgelösten Spektren wäre es möglich, eine genauere Bestimmung der Häufigkeiten zu machen und die Unsicherheiten in den Atomdaten zu verkleinern, da Oszillatorenstärken und Wellenlängen anhand der Beobachtung geprüft werden können.

## Absorption im EUV-Bereich

Mit dem neuen *TEUV*-Dienst kann eine Korrektur durch interstellare Absorption an synthetische Spektren im EUV-Bereich angebracht werden. Dafür wurde ein Programm entwickelt, das eine Opazität, verursacht durch gebunden-frei-Absorptionsquerschnitte vom Grundzustandsniveau der Ionen H I, He I–II, C I–III, N I–III, O I–III, basierend auf Opacity Project-Daten an den Modellfluss anbringt. Gesteuert wird das neue Programm über eine speziell entwickelte Webseite mit deren Hilfe der Nutzer zum Teil interaktiv die Parameter anpassen kann. Das resultierende Modell kann direkt heruntergeladen werden.

Als erste Anwendung des *TEUV*-Dienstes wurden HHeCNOCaScTiVCrMnFeCoNi-Modelle für den DO-Weißen Zwerg RE 0503–289 ( $T_{\text{eff}} = 70\,000\text{ K}$ ,  $\log g = 7.5$ ) berechnet und mit Hilfe dieses Dienstes um die interstellare Absorption korrigiert. RE 0503–289 zeigt ein EUV-Spektrum, das bisher nicht konsistent reproduziert werden kann. Aufgrund der Vermutung, dass Eisengruppenelemente für diesen Verlauf verantwortlich sein könnten, wurde der Einfluss aller Eisengruppenelemente auf den Flussverlauf dieses Objekts mit den aktualisierten Atomdaten untersucht. Es zeigt sich, dass die Eisengruppenelemente nicht (allein) für diesen Verlauf verantwortlich sein können. Durch die Berücksichtigung interstellarer Absorption nähert sich das Modell der Beobachtung an, sie reicht jedoch als alleinige Erklärung nicht aus. Die zusätzlich zu den meisten bisherigen EUV-Absorptionsprogrammen berücksichtigten Ionen von C, N und O führen nicht zur Lösung des Problems und haben eine untergeordnete Rolle in der Gesamtabsorption durch das interstellare Medium, was frühere Analysen des interstellaren Mediums im

EUV-Bereich bestätigt. Um den EUV-Flussverlauf von RE 0503–289 zu reproduzieren, fehlen also immer noch Absorber in der Modellatmosphäre. Aufgrund der Identifikation von Elementen schwerer als die Eisengruppe liegt die Vermutung nahe, dass die fehlenden Opazitätsquellen von diesen Elementen stammen. Momentan sind die vorhandenen Atomdaten aber noch zu unvollständig, um dies modellieren zu können.

### 5.3. Ausblick

Diese Arbeit zeigt, dass Spektralanalysen über das *VO* mit den bereits vor dieser Arbeit existierenden Diensten für eine erste Klassifizierung von Sternen oder erste Analysen durchgeführt werden können. Durch die Erweiterungen der existierenden und Erzeugung der neuen Dienste sind nun noch präzisere, detailliertere und realistischere Spektralanalysen über das Internet möglich. Die Berücksichtigung der Eisengruppen-elemente hat sich in den durchgeführten Tests als von großer Bedeutung herausgestellt. Um dem *VO*-Nutzer in Zukunft die Aufnahme dieser Elemente in *TMAW*-Analysen zu ermöglichen, werden diese in einem laufenden Projekt, in dem aktuell *TMAW* um die Elemente Neon und Magnesium erweitert wird, in *TMAW* integriert.

Um mit dem Trend steigender astronomischer Datenmengen mithalten zu können, werden in bereits angelaufenen Tübinger Projekten weitere Dienste entwickelt, die zum Teil auf den existierenden Diensten basieren und die Analysen über Visualisierungen der Modelle zusammen mit der Beobachtung im Rahmen des geplanten *JANALYSIS*-Dienstes erleichtern. Zusätzlich wird ein *TCHI*-Dienst entwickelt, mit dem ein automatischer  $\chi^2$ -Fit von Modellen aus der *TheoSSA*-Datenbank an vom Nutzer hochgeladene Beobachtungen durchgeführt werden kann. Als Reaktion auf die steigende Zahl von Multi-Objekt Beobachtungen wird der Dienst *TMOS* entwickelt, der eine automatische Klassifizierung einer großen Anzahl von beobachteten Sternspektren innerhalb kurzer Zeit durchführen wird.



# Literaturverzeichnis

- Althaus, L. G., Serenelli, A. M., Panei, J. A., Córscico, A. H., García-Berro, E., & Scóccola, C. G. 2005, *Astronomy and Astrophysics* **435**, 631
- Anderson, L. S. 1985, *The Astrophysical Journal* **298**, 848
- Anderson, L. S. 1989, *The Astrophysical Journal* **339**, 558
- Anderson, L. S. 1992, in M. S. Giampapa & J. A. Bookbinder (eds.), *Cool Stars, Stellar Systems, and the Sun*, Vol. 26 of *Astronomical Society of the Pacific Conference Series*, p. 509
- Angulo, R. E., Springel, V., White, S. D. M., Jenkins, A., Baugh, C. M., & Frenk, C. S. 2012, *Monthly Notices of the Royal Astronomical Society* **426**, 2046
- Armstrong, B. 1967, *Journal of Quantitative Spectroscopy & Radiative Transfer* **7**, 61
- Auer, L. H. & Mihalas, D. 1969, *The Astrophysical Journal* **158**, 641
- Barstow, M. A., Dobbie, P. D., Forbes, A. E., & Boyce, D. D. 2007, in R. Napiwotzki & M. R. Burleigh (ed.), *15th European Workshop on White Dwarfs*, Vol. 372 of *Astronomical Society of the Pacific Conference Series*, p. 243
- Barstow, M. A., Dreizler, S., Holberg, J. B., Finley, D. S., Werner, K., Hubeny, I., & Sion, E. M. 2000, *Monthly Notices of the Royal Astronomical Society* **314**, 109
- Barstow, M. A., Holberg, J. B., Werner, K., Buckley, D. A. H., & Stobie, R. S. 1994, *Monthly Notices of the Royal Astronomical Society* **267**, 653
- Barstow, M. A. & Sion, E. M. 1994, *Monthly Notices of the Royal Astronomical Society* **271**, L52
- Barstow, M. A., Wesemael, F., Holberg, J. B., Werner, K., Buckley, D. A. H., Stobie, R. S., Fontaine, G., Rosen, S. R., Demers, S., & Lamontagne, R. 1993, *Advances in Space Research* **13**, 281
- Behara, N. T. & Jeffery, C. S. 2006, *Astronomy and Astrophysics* **451**, 643
- Blanchette, J.-P., Chayer, P., Wesemael, F., Fontaine, G., Fontaine, M., Dupuis, J., Kruk, J. W., & Green, E. M. 2008, *The Astrophysical Journal* **678**, 1329
- Bond, H. E., Grauer, A. D., Green, R. F., & Liebert, J. W. 1984, *The Astrophysical Journal* **279**, 751
- Cannon, C. J. 1973, *The Astrophysical Journal* **185**, 621
- Castellani, M. & Castellani, V. 1993, *The Astrophysical Journal* **407**, 649
- Charpinet, S., Fontaine, G., Brassard, P., Chayer, P., Rogers, F. J., Iglesias, C. A., & Dorman, B. 1997, *Astrophysical Journal Letters* **483**, L123
- Chayer, P., Fontaine, G., & Wesemael, F. 1995, *Astrophysical Journal Supplement Series* **99**, 189
- Chayer, P., Vennes, S., Dupuis, J., & Kruk, J. W. 2005, *The Astrophysical Journal* **630**, L169

- Cruddace, R., Paresce, F., Bowyer, S., & Lampton, M. 1974, *The Astrophysical Journal* **187**, 497
- Curtis, A. & Osborne, M. R. 1966, *The Computer Journal* **9**, 286
- D’Cruz, N. L., Dorman, B., Rood, R. T., & O’Connell, R. W. 1996, *The Astrophysical Journal* **466**, 359
- D’Cruz, N. L., O’Connell, R. W., Rood, R. T., Whitney, J. H., Dorman, B., Landsman, W. B., Hill, R. S., Stecher, T. P., & Bohlin, R. C. 2000, *The Astrophysical Journal* **530**, 352
- Deetjen, J. L. 1999, *Diplomarbeit, Universität Tübingen*
- Dreizler, S. 1999, *Astronomy and Astrophysics* **352**, 632
- Dreizler, S. & Heber, U. 1998, *Astronomy and Astrophysics* **334**, 618
- Dreizler, S. & Werner, K. 1992, in U. Heber & C. S. Jeffery (ed.), *The Atmospheres of Early-Type Stars*, Vol. 401 of *Lecture Notes in Physics*, Berlin Springer Verlag, p. 436
- Dreizler, S. & Werner, K. 1993, *Astronomy and Astrophysics* **278**, 199
- Dreizler, S. & Werner, K. 1996, *Astronomy and Astrophysics* **314**, 217
- Drilling, J. S. 1983, *The Astrophysical Journal, Part2 - Letters to the Editor* **270**, L13
- Edelmann, H. 2003, *Ph.D. thesis*, Universität Erlangen-Nürnberg
- Edelmann, H., Heber, U., & Napiwotzki, R. 2001, *Astronomische Nachrichten* **322**, 401
- Ekberg, J. O. 1993, *Astronomy and Astrophysics Supplement* **101**, 1
- Fitzpatrick, E. L. 1999, *The Publications of the Astronomical Society of the Pacific* **111**, 63
- Fontaine, G. & Chayer, P. 1997, in A. G. D. Philip, J. Liebert, R. Saffer, & D. S. Hayes (eds.), *The Third Conference on Faint Blue Stars*, p. 169
- Garcia, J., Kelly, R., & Voran, T. 2009, *Proceedings of the 10th ICI international conference on high-performance clustered computing* **1**, 1
- Geier, S., Heber, U., Edelmann, H., Morales-Rueda, L., & Napiwotzki, R. 2010, *Astrophysics and Space Science* **329**, 127
- Grauer, A. D., Liebert, J., & Green, R. 1989, in G. Wegner (ed.), *IAU Colloq. 114: White Dwarfs*, Vol. 328 of *Lecture Notes in Physics*, Berlin Springer Verlag, pp 119–121
- Green, R. F., Schmidt, M., & Liebert, J. 1986, *Astrophysical Journal Supplement Series* **61**, 305
- Haas, S. 1997, *Ph.D. thesis*, Universität Erlangen-Nürnberg
- Haas, S., Dreizler, S., Heber, U., Jeffery, S., & Werner, K. 1996, *Astronomy and Astrophysics* **311**, 669
- Han, Z., Podsiadlowski, P., Maxted, P. F. L., & Marsh, T. R. 2003, *Monthly Notice of the Royal Astronomical Society* **341**, 669
- Han, Z., Podsiadlowski, P., Maxted, P. F. L., Marsh, T. R., & Ivanova, N. 2002, *Monthly Notice of the Royal Astronomical Society* **336**, 449
- Heber, U. 2009, *Annual Review of Astronomy and Astrophysics* **47**, 211



- Hellwege, K. 1965, *Landolt-Börnstein Zahlenwerte und Funktionen aus Naturwissenschaft und Technik, Neue Serie, Gruppe VI: Astronomie Astrophysik und Weltraumforschung, Band 1, Hrsg. v. H. H. Voigt*, Springer Verlag Berlin - Heidelberg - New York
- Hellwege, K. 1982, *Landolt-Börnstein Zahlenwerte und Funktionen aus Naturwissenschaft und Technik, Neue Serie, Gruppe VI: Astronomie Astrophysik und Weltraumforschung, Band 2, Hrsg. v. K. Schaifers und H. H. Voigt*, Springer Verlag Berlin - Heidelberg - New York
- Herald, J. E., Bianchi, L., & Hillier, D. J. 2005, *The Astrophysical Journal* **627**, 424
- Herschel, W. 1800, *Royal Society of London Philosophical Transactions Series I* **90**, 255
- Hertzprung, E. 1911, *Publikationen des Astrophysikalischen Observatoriums zu Potsdam* 63
- Hirata, R. & Horaguchi, T. 1995, *Atomic Spectral Line List, Department of Astronomy (Kyoto: University and National Science Museum)*
- Hirsch, H. 2009, *Ph.D. thesis*, Universität Erlangen-Nürnberg
- Hoffmann, A. 2005, *Diplomarbeit, Universität Tübingen*
- Hubeny, I. 1988, *Computer Physics Communications* **52**, 103
- Hubeny, I. & Lanz, T. 1995, *The Astrophysical Journal* **439**, 875
- Hubeny, I. & Lanz, T. 2011, *Astrophysics Source Code Library* p. 9021
- Hui, A. K., Armstrong, B. H., & Wray, A. A. 1978, *Journal of Quantitative Spectroscopy & Radiative Transfer* **19**, 509
- Humlíček, J. 1982, *Journal of Quantitative Spectroscopy & Radiative Transfer* **27**, 437
- Humlíček, J. 1979, *Journal of Quantitative Spectroscopy & Radiative Transfer* **21**, 309
- Husfeld, D., Butler, K., Heber, U., & Drilling, J. S. 1989, *Astronomy and Astrophysics* **222**, 150
- Iben, Jr., I., Kaler, J. B., Truran, J. W., & Renzini, A. 1983, *The Astrophysical Journal* **264**, 605
- Jeans, J. H. 1902, *Philosophical Transactions of the Royal Society A* **199**, 1
- Justham, S., Podsiadlowski, P., & Han, Z. 2011, *Monthly Notices of the Royal Astronomical Society* **410**, 984
- Kilkenny, D., O'Donoghue, D., Koen, C., Stobie, R. S., & Chen, A. 1997, *Monthly Notices of the Royal Astronomical Society* **287**, 867
- Kippenhahn, R. & Weigert, A. 1990, *Stellar Structure and Evolution*, Springer-Verlag
- Kirchhoff, G. 1859, *Monatsbericht der Königlichen Preussischen Akademie der Wissenschaften zu Berlin* p. 662
- Klepp, S. & Rauch, T. 2011, *Astronomy and Astrophysics* **531**, L7
- Koester, D., Liebert, J., & Saffer, R. A. 1994, *The Astrophysical Journal* **422**, 783
- Koesterke, L., Dreizler, S., & Rauch, T. 1998, *Astronomy and Astrophysics* **330**, 1041
- Koesterke, L. & Werner, K. 1998, *Astrophysical Journal Letters* **500**, L55

- Kurucz, R. L. 1992, in B. Barbuy & A. Renzini (eds.), *The Stellar Populations of Galaxies*, Vol. 149 of *IAU Symposium*, p. 225
- Kurucz, R. L. 2009, in I. Hubeny, J. M. Stone, K. MacGregor, & K. Werner (ed.), *American Institute of Physics Conference Series*, Vol. 1171 of *American Institute of Physics Conference Series*, p. 43
- Kurucz, R. L. 2011, *Canadian Journal of Physics* **89**, 417
- Langer, N. 1995, *Leben und Sterben der Sterne*, Beck, C.H., München
- Lanz, T., Barstow, M. A., Hubeny, I., & Holberg, J. B. 1996, *The Astrophysical Journal* **473**, 1089
- Lanz, T., Brown, T. M., Sweigart, A. V., Hubeny, I., & Landsman, W. B. 2004, *The Astrophysical Journal* **602**, 342
- Lanz, T. & Hubeny, I. 2003, *Astrophysical Journal Supplement Series* **146**, 417
- Lucy, L. B. 1964, *SAO Special Report* **167**, 93
- Maxted, P. f. L., Heber, U., Marsh, T. R., & North, R. C. 2001, *Monthly Notices of the Royal Astronomical Society* **326**, 1391
- Mengel, J. G., Norris, J., & Gross, P. G. 1976, *The Astrophysical Journal* **204**, 488
- Michaud, G., Richer, J., & Richard, O. 2008, *The Astrophysical Journal* **675**, 1223
- Michaud, G., Richer, J., & Richard, O. 2011, *Astronomy and Astrophysics* **529**, A60
- Miller Bertolami, M. M., Althaus, L. G., Unglaub, K., & Weiss, A. 2008, *Astronomy and Astrophysics* **491**, 253
- Moore, G. 1965, *Electronics* **38**, 8
- Napiwotzki, R. 1997, *Astronomy and Astrophysics* **322**, 256
- Napiwotzki, R., Christlieb, N., Drechsel, H., Hagen, H.-J., Heber, U., Homeier, D., Karl, C., Koester, D., Leibundgut, B., Marsh, T. R., Moehler, S., Nelemans, G., Pauli, E.-M., Reimers, D., Renzini, A., & Yungelson, L. 2003, *The Messenger* **112**, 25
- Napiwotzki, R. & Rauch, T. 1994, *Astronomy and Astrophysics* **285**, 603
- Newton, I. 1671, *Philosophical Transactions of the Royal Society of London* **6 no. 69-80**, 3075
- O'Toole, S. J. & Heber, U. 2006, *Astronomy and Astrophysics* **452**, 579
- Peytremann, E. 1974, *Astronomy and Astrophysics* **33**, 230
- Polonski, E. F., Vennes, S., & Chayer, P. 1995, in *American Astronomical Society Meeting Abstracts*, Vol. 27 of *Bulletin of the American Astronomical Society*, p. 1311
- Rauch, T. 1997, *Astronomy and Astrophysics* **320**, 237
- Rauch, T. 2003, *Astronomy and Astrophysics* **403**, 709
- Rauch, T. 2008, in M. Guainazzi & P. Osuna (eds.), *Astronomical Spectroscopy and Virtual Observatory*, p. 183
- Rauch, T. & Deetjen, J. L. 2003, in I. Hubeny, D. Mihalas, & K. Werner (ed.), *Stellar Atmosphere Modeling*, Vol. 288 of *Astronomical Society of the Pacific Conference Series*, p. 103

- Rauch, T. & Werner, K. 1988, *Astronomy and Astrophysics* **202**, 159
- Rauch, T., Werner, K., Biémont, É., Quinet, P., & Kruk, J. W. 2012, *Astronomy and Astrophysics* **546**, A55
- Rauch, T., Werner, K., & Kruk, J. W. 2010, *Astrophysics and Space Science* **329**, 133
- Reiff, E., Jahn, D., Rauch, T., Werner, K., Herwig, F., & Kruk, J. W. 2006a, in M. J. Barlow & R. H. Méndez (eds.), *Planetary Nebulae in our Galaxy and Beyond*, Vol. 234 of *IAU Symposium*, p. 489
- Reiff, E., Jahn, D., Rauch, T., Werner, K., Kruk, J. W., & Herwig, F. 2006b, in C. Aerts & C. Sterken (eds.), *Astrophysics of Variable Stars*, Vol. 349 of *Astronomical Society of the Pacific Conference Series*, p. 323
- Reiff, E., Jahn, D., Rauch, T., Werner, K., Kruk, J. W., & Herwig, F. 2007, in R. Napiwotzki & M. R. Burleigh (eds.), *15th European Workshop on White Dwarfs*, Vol. 372 of *Astronomical Society of the Pacific Conference Series*, p. 237
- Reiff, E., Rauch, T., Werner, K., & Kruk, J. W. 2005, in D. Koester & S. Moehler (eds.), *14th European Workshop on White Dwarfs*, Vol. 334 of *Astronomical Society of the Pacific Conference Series*, p. 225
- Reiff, E., Rauch, T., Werner, K., Kruk, J. W., & Koesterke, L. 2008, in A. Werner & T. Rauch (eds.), *Hydrogen-Deficient Stars*, Vol. 391 of *Astronomical Society of the Pacific Conference Series*, p. 121
- Ringat, E. 2010, *Diplomarbeit, Universität Tübingen*
- Ritter, J. W. 1801, *Annalen der Physik* **7**, 527
- Rumph, T., Bowyer, S., & Vennes, S. 1994, *The Astronomical Journal* **107**, 2108
- Russell, H. N. 1913, *The Observatory* **36**, 324
- Saio, H. & Jeffery, C. S. 2000, *Monthly Notices of the Royal Astronomical Society* **313**, 671
- Scharmer, G. B. 1981, *The Astrophysical Journal* **249**, 720
- Schönberner, D. & Drilling, J. S. 1984, *The Astrophysical Journal* **278**, 702
- Schreier, F. 1992, *Journal of Quantitative Spectroscopy & Radiative Transfer* **48**, 743
- Seaton, M. J. 1962, in *Atomic and Molecular Processes*, ed. D. R. Bates (New York: Academic Press)
- Seaton, M. J., Yan, Y., Mihalas, D., & Pradhan, A. K. 1994, *Monthly Notices of the Royal Astronomical Society* **266**, 805
- Sion, E. M. 2011, *Hot White Dwarfs*, pp 1–24, in: *White Dwarf Atmospheres and Circumstellar Environments*, (ed D. W. Hoard), Wiley-VCH Verlag GmbH & Co. KGaA, Weinheim, Germany
- Stroer, A., Heber, U., Lisker, T., Napiwotzki, R., Dreizler, S., Christlieb, N., & Reimers, D. 2007, *Astronomy and Astrophysics* **462**, 269
- Stys, D., Slevinsky, R., Sion, E. M., Saffer, R., Holberg, J. B., O'Donoghue, D., Kilkenny, D., Stobie, R. S., & Koen, C. 2000, *The Publications of the Astronomical Society of the Pacific* **112**, 354
- Unglaub, K. & Bues, I. 2000, *Astronomy and Astrophysics* **359**, 1042
- Unglaub, K. & Bues, I. 2001, *Astronomy and Astrophysics* **374**, 570

- van Regemorter, H. 1962, *The Astrophysical Journal* **136**, 906
- Vennes, S., Dupuis, J., Chayer, P., Polomski, E. P., Dixon, W. V. D., & Hurwitz, M. 1998, *The Astrophysical Journal* **500**, L41
- Vennes, S., Jelinsky, P., Wiercigroch, A., Bowyer, S., Malina, R. F., Fontaine, G., & Wesemael, F. 1992, in *American Astronomical Society Meeting Abstracts*, Vol. 24 of *Bulletin of the American Astronomical Society*, p. 1204
- Vennes, S., Pradhan, A. K., Fontaine, G., Dupuis, J., & Wesemael, F. 1994, in *American Astronomical Society Meeting Abstracts #184*, Vol. 26 of *Bulletin of the American Astronomical Society*, p. 868
- von Fraunhofer, J. 1817, *Annalen der Physik* **56**, 264
- Wassermann, D., Werner, K., Rauch, T., & Kruk, J. W. 2010, *Astronomy and Astrophysics* **524**, A9
- Werner, K. 1986, *Astronomy and Astrophysics* **161**, 177
- Werner, K. 1988, *Astronomy and Astrophysics* **204**, 159
- Werner, K. 1996, *Astronomy and Astrophysics* **309**, 861
- Werner, K., Deetjen, J. L., Dreizler, S., Nagel, T., Rauch, T., & Schuh, S. L. 2003, in I. Hubeny, D. Mihas, & K. Werner (ed.), *Stellar Atmosphere Modeling*, Vol. 288 of *Astronomical Society of the Pacific Conference Series*, p. 31
- Werner, K., Deetjen, J. L., Rauch, T., & Wolff, B. 2001, in J. L. Provencal, H. L. Shipman, J. MacDonald, & S. Goodchild (ed.), *12th European Workshop on White Dwarfs*, Vol. 226 of *Astronomical Society of the Pacific Conference Series*, p. 55
- Werner, K., Dreizler, S., Heber, U., Kappelman, N., Kruk, J., Rauch, T., & Wolff, B. 1997, in R. E. Schielicke (ed.), *Reviews in Modern Astronomy*, Vol. 10 of *Reviews in Modern Astronomy*, p. 219
- Werner, K., Heber, U., & Hunger, K. 1991, *Astronomy and Astrophysics* **244**, 437
- Werner, K. & Herwig, F. 2006, *The Publications of the Astronomical Society of the Pacific* **118**, 183
- Werner, K. & Husfeld, D. 1985, *Astronomy and Astrophysics* **148**, 417
- Werner, K., Rauch, T., & Kruk, J. W. 2005, *Astronomy and Astrophysics* **433**, 641
- Werner, K., Rauch, T., & Kruk, J. W. 2007, *Astronomy and Astrophysics* **466**, 317
- Werner, K., Rauch, T., & Kruk, J. W. 2010, in K. Werner & T. Rauch (eds.), *American Institute of Physics Conference Series*, Vol. 1273 of *American Institute of Physics Conference Series*, p. 75
- Werner, K., Rauch, T., Kruk, J. W., & Kurucz, R. L. 2011, *Astronomy and Astrophysics* **531**, A146
- Werner, K., Rauch, T., Reiff, E., & Kruk, J. W. 2008, in A. Werner & T. Rauch (eds.), *Hydrogen-Deficient Stars*, Vol. 391 of *Astronomical Society of the Pacific Conference Series*, p. 109
- Werner, K., Rauch, T., Reiff, E., Kruk, J. W., & Napiwotzki, R. 2004, *Astronomy and Astrophysics* **427**, 685
- Werner, K., Rauch, T., Ringat, E., & Kruk, J. W. 2012a, *The Astrophysical Journal Letters* **753**, L7
- Werner, K., Rauch, T., Ringat, E., & Kruk, J. W. 2012b, *ArXiv e-prints*, arXiv:1209.4977

Wesemael, F., Veilleux, S., Lamontagne, R., Fontaine, G., & Holberg, J. B. 1985, *The Astrophysical Journal* **298**, 859

Zhang, X. & Jeffery, C. S. 2012, *Monthly Notices of the Royal Astronomical Society* **419**, 452



## A. Anhang

Die erweiterten und neu erstellten Dienste werden mit Hilfe von Unix-Shell-Skripten und FORTRAN-Programmen gesteuert. Die Dienste *TIRO*, *TMAW* und *TEUV* werden im Rahmen dieser Arbeit vollständig dokumentiert. Die zugehörigen Prozeduren sind deshalb im Folgenden gelistet.

### A. *TIRO*-Prozeduren

Für die Bearbeitung der *TIRO*-Requests führt ein Cron-Job regelmäßig die Prozedur `checkrun.bat` aus, mit der die freie Rechenkapazität sowie die aktuelle bearbeiteten Requests abgefragt werden. Diese Prozedur startet anschließend die Prozedur `run_ions.bat` bzw. `run_inter.sh`, die die Berechnung der Querschnittsdateien und Atomdaten durchführt. `run_ions.bat` erzeugt die nötigen Dateien, die die *IrOnIc*-Parameter enthalten und die die Programmdatei ausführen können. Dann wird die Temperatur berechnet, bei der die jeweilige Ionisationsstufe dominant ist. Die Ergebnisse werden in die neuen Parameter-Dateien aufgenommen und die Querschnitte werden berechnet. Für den Start der Berechnungen wird die Prozedur `startjob.bat` verwendet. Die resultierenden Atomdaten werden mit `ironic_combine_web.job` verarbeitet und in eine Datei zusammengefügt. Anschließend wird alles in ein komprimiertes tar-Archiv gepackt und der Nutzer über dessen URL informiert. Die Prozedur `run_inter.sh` prüft zuerst, ob eine Interpolation zwischen Werten aus der *TIRO*-Datenbank möglich ist und führt diese mit Hilfe von `interpol.sh` durch. Falls nicht, werden die Querschnitte und Atomdaten analog zu `run_ions.bat` erzeugt. Diese Prozeduren wurden im Rahmen dieser Arbeit neu geschrieben, lediglich für `ironic_combine_web.job` wurde eine bestehenden Version modifiziert. Zusätzlich wird die bereits bestehende Prozedur `ChangeIrOnIcHeaders` verwendet. Der Stand der Prozeduren am 25.01.2013 wird im Folgenden gelistet. Außerdem wird ein Beispiel der oben erwähnten Steuerdateien `ironic_CaSc.in` und `ironic_CaSc.job` gegeben.

Listing A.1: Die checkrun.bat-Prozedur ist für die Überprüfung der freien Rechenkapazität des GAVO @IAAT-Servers und den Start der Prozedur run\_ions.bat zuständig.

```
#!/bin/bash
#####
# Check if a request is waiting for calculation , the load
# is below a given limit and no other request is running
#
# LAST_MODIFICATION="24-Jan-2013 18:01:17"
#
# Copyright 2012 – 2013 by E. Mueller–Ringat
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D–72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#
#####
# CHANGELOG:
# 01.06.2012: created
# 14.12.2012: change location , the new user is TIRO instead of ironic
#
#####
#
# script will only start if there is space on aitp20
#
DATE='date +%Y-%m-%d_%H:%M:%S'
loadmax=24
# check the current load
#
echo `w | head -1` > load.dat
load=~rauch/tools/readload.Linux_X86 `
echo `w | head -1` > load.dat
load2=~rauch/tools/readload.Linux_X86 `
#
if [ ${load} -ge ${loadmax} ]
then
    exit
fi
#
# check if there are other TIRO jobs running
#
RunCount=$(ls /home/TIRO/requests/running/iro_201* | wc -l)
if [ $RunCount -gt 0 ]
then
    exit
fi
#
# if a request is waiting for calculation , start run_ions.bat
#
ReqDir=/home/TIRO/requests
cd ${ReqDir}
#
# now check if there are requests and create a directory for the selected request
#
REQCOUNT=$(find -L ${ReqDir} -mindepth 1 -maxdepth 1 -name 'iro_201*' | wc -l)
if [ ${REQCOUNT} -gt 0 ]
then
    nice -19 /home/TIRO/scripts/run_ions.bat 2> /home/TIRO/requests/ERROR_${DATE} > OUTPUT_${DATE}
fi
fi
```



Listing A.2: Die run\_ions.bat-Prozedur ist für die Erzeugung der Querschnitte und Modellatome sowie die dafür nötigen Parameter-Dateien zuständig.

```
#!/bin/bash
#####
# Read a request file and create the *.in file (old format),
# perform the first run (TSAR), create *.in and *.job files
# for the third run, perform it, combine all model atom parts to one
# model atom, run ChangelrOnlcHeaders, put everything into a tarball
# and write an email to the requestor
#
# LAST_MODIFICATION="24-Jan-2013 17:52:15"
#
# Copyright 2012 – 2013 by E. Mueller-Ringat
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#
#####
# CHANGELOG:
# 30.10.2011: created
#
# 30.10.2011: Changed from run.bat to run single
#             ionization stages in parallel
# 12.03.2012: Create directories with a date, create all file in
#             this directory, run them and afterwards make a tarball
#             and copy that to a directory accessible via wget
#             then write an email
#             Send email before start
# 20.04.2012: Email to the administrator if no RBB files were created
#             or if "Error" is written in the output file
# 01.06.2012 The check of the load and if other requests are running is
#             transfered to checkrun.bat
# 14.12.2012: change location, the new user is TIRO instead of ironic
# 17.12.2012: change email text to user
# 17.12.2012: minor style changes
#
#####
#
ReqDir=/home/TIRO/requests
cd ${ReqDir}
#
#
ReqName=$(ls -t1 ~TIRO/requests/iro_201* | head -1)
cp ${ReqName} /home/TIRO/requests/running/
REGFILE='echo ${ReqName}|cut -c21-48'
#
# change name to avoid ":"
#
ReqFile='echo ${REGFILE} | sed -e 's\:\_\g''
#
ReqDate='echo ${ReqFile} | cut -c5-50'
mkdir ${ReqDate}
mkdir ${ReqDate}/data
mv ${ReqName} /home/TIRO/requests/${ReqDate}/
cd /home/TIRO/requests/${ReqDate}
#
# read the request file
#
i=0
while read line
do i=$((i+1))
  readarr[$i]=$line
```

```

echo "$i : ${readarr[$i]}"
done < ${REGFILE}
#
#   REGFILE=${readarr[1]}
  firstname=${readarr[2]}
  lastname=${readarr[3]}
  addy=${readarr[4]}
  linet=${readarr[5]}
  ilow=${readarr[6]}
  iup=${readarr[7]}
  CAinfo=${readarr[8]}
  SCinfo=${readarr[9]}
  TIinfo=${readarr[10]}
  Vinfo=${readarr[11]}
  CRinfo=${readarr[12]}
  MNinfo=${readarr[13]}
  FEinfo=${readarr[14]}
  COinfo=${readarr[15]}
  NIinfo=${readarr[16]}
#
# send email to notify that the request calculation starts now
#
  cat > MAIL <<EOF
Dear ${firstname} ${lastname},

the request sent at ${ReqDate} starts calculating now.
You will be informed when it finished and you can retrieve
the results.

If you have further questions,
please do not hesitate to contact ironic@listserv.uni-tuebingen.de.

When publishing research making use of this service, please acknowledge:
"The TIRO service (http://astro-uni-tuebingen.de/~TIRO) used to calculate
opacities for this paper was constructed as part of the
activities of the German Astrophysical Virtual Observatory."

Yours
TIRO Team
EOF
#
  /usr/bin/mail -r "TIRO" -s "TIRO: Calculation started" ${addy} < MAIL
  /usr/bin/mail -r "TIRO" -s "TIRO: Calculation started" \
  ironic@listserv.uni-tuebingen.de < MAIL
rm MAIL
#
  REQFILE=${ReqDir}/${ReqDate}/${REGFILE}
/home/TIRO/FORTRAN/create_infile.Linux_x64 << EOF
${REQFILE}
1
EOF
#
##### now old input files for first run are created
#
cd /home/TIRO/requests/${ReqDate}/
ls
#
# now perform the first run for all elements
#
if test -s ironic_CA_TSTAR.in
then
  cp /home/TIRO/requests/${ReqDate}/ironic_CA_TSTAR.in ironic_input.dat
  cp /atmosarray/rauch/bin/ironic/linux/ironic_tstar.Linux_n64 jobrun
  nice -19 ./jobrun >& STDOUT
  egrep -i 'error|warning' STDOUT
  rm jobrun
  rm ironic_input.dat
  # now create the in file for the third run with the correct TSTAR values
  cp /home/TIRO/requests/${ReqDate}/TSTAR_CA.dat TSTAR_firstrun
/home/TIRO/FORTRAN/create_infile_single.Linux_x64 << EOF
${REQFILE}

```

```

3
1
EOF
#
# create the jobfile for the third run
/home/TIRO/FORTRAN/create_jobfile_single.Linux_x64 << EOF
${ilow}
${iup}
ironic_CA
${ReqDate}
EOF
chmod 700 ironic*.job
rm TSTAR_firstrun
#
fi
if test -s ironic_SC_TSTAR.in
then
cp /home/TIRO/requests/${ReqDate}/ironic_SC_TSTAR.in ironic_input.dat
cp /atmosarray/rauch/bin/ironic/linux/ironic_tstar.Linux_n64 jobrun
nice -19 ./jobrun >& STDOUT
egrep -i 'error|warning' STDOUT
rm jobrun
rm ironic_input.dat
# now create the in file for the third run with the correct TSTAR values
cp ~/requests/${ReqDate}/TSTAR_SC.dat TSTAR_firstrun
/home/TIRO/FORTRAN/create_infile_single.Linux_x64 << EOF
${REQFILE}
3
2
EOF
#
# create the jobfile for the third run
/home/TIRO/FORTRAN/create_jobfile_single.Linux_x64 << EOF
${ilow}
${iup}
ironic_SC
${ReqDate}
EOF
chmod 700 ironic*.job
rm TSTAR_firstrun
#
fi
if test -s ironic_TI_TSTAR.in
then
cp /home/TIRO/requests/${ReqDate}/ironic_TI_TSTAR.in ironic_input.dat
cp /atmosarray/rauch/bin/ironic/linux/ironic_tstar.Linux_n64 jobrun
nice -19 ./jobrun >& STDOUT
egrep -i 'error|warning' STDOUT
rm jobrun
rm ironic_input.dat
# now create the in file for the third run with the correct TSTAR values
cp ~/requests/${ReqDate}/TSTAR_TI.dat TSTAR_firstrun
/home/TIRO/FORTRAN/create_infile_single.Linux_x64 << EOF
${REQFILE}
3
3
EOF
#
# create the jobfile for the third run
/home/TIRO/FORTRAN/create_jobfile_single.Linux_x64 << EOF
${ilow}
${iup}
ironic_TI
${ReqDate}
EOF
chmod 700 ironic*.job
rm TSTAR_firstrun
#
fi
if test -s ironic_V__TSTAR.in
then

```

```

cp /home/TIRO/requests/${ReqDate}/ironic_V__TSTAR.in ironic_input.dat
cp /atmosarray/rauch/bin/ironic/linux/ironic_tstar.Linux_n64 jobrun
nice -19 ./jobrun >& STDOUT
egrep -i 'error|warning' STDOUT
rm jobrun
rm ironic_input.dat
# now create the in file for the third run with the correct TSTAR values
cp ~/requests/${ReqDate}/TSTAR_V_.dat TSTAR_firstrun
/home/TIRO/FORTRAN/create_infile_single.Linux_x64 << EOF
${REQFILE}
3
4
EOF
#
# create the jobfile for the third run
/home/TIRO/FORTRAN/create_jobfile_single.Linux_x64 << EOF
${ilow}
${iup}
ironic_V_
${ReqDate}
EOF
chmod 700 ironic*.job
rm TSTAR_firstrun
#
fi
if test -s ironic_CR_TSTAR.in
then
cp /home/TIRO/requests/${ReqDate}/ironic_CR_TSTAR.in ironic_input.dat
cp /atmosarray/rauch/bin/ironic/linux/ironic_tstar.Linux_n64 jobrun
nice -19 ./jobrun >& STDOUT
egrep -i 'error|warning' STDOUT
rm jobrun
rm ironic_input.dat
# now create the in file for the third run with the correct TSTAR values
cp ~/requests/${ReqDate}/TSTAR_CR.dat TSTAR_firstrun
/home/TIRO/FORTRAN/create_infile_single.Linux_x64 << EOF
${REQFILE}
3
5
EOF
#
# create the jobfile for the third run
/home/TIRO/FORTRAN/create_jobfile_single.Linux_x64 << EOF
${ilow}
${iup}
ironic_CR
${ReqDate}
EOF
chmod 700 ironic*.job
rm TSTAR_firstrun
#
fi
if test -s ironic_MN_TSTAR.in
then
cp /home/TIRO/requests/${ReqDate}/ironic_MN_TSTAR.in ironic_input.dat
cp /atmosarray/rauch/bin/ironic/linux/ironic_tstar.Linux_n64 jobrun
nice -19 ./jobrun >& STDOUT
egrep -i 'error|warning' STDOUT
rm jobrun
rm ironic_input.dat
# now create the in file for the third run with the correct TSTAR values
cp ~/requests/${ReqDate}/TSTAR_MN.dat TSTAR_firstrun
/home/TIRO/FORTRAN/create_infile_single.Linux_x64 << EOF
${REQFILE}
3
6
EOF
#
# create the jobfile for the third run
/home/TIRO/FORTRAN/create_jobfile_single.Linux_x64 << EOF
${ilow}

```

```

    ${iup}
    ironic_MN
    ${ReqDate}
EOF
    chmod 700 ironic*.job
    rm TSTAR_firstrun
    #
fi
if test -s ironic_FE_TSTAR.in
then
    cp /home/TIRO/requests/${ReqDate}/ironic_FE_TSTAR.in ironic_input.dat
    cp /atmosarray/rauch/bin/ironic/linux/ironic_tstar.Linux_n64 jobrun
    nice -19 ./jobrun >& STDOUT
    egrep -i 'error|warning' STDOUT
    rm jobrun
    rm ironic_input.dat
    # now create the in file for the third run with the correct TSTAR values
    cp ~/requests/${ReqDate}/TSTAR_FE.dat TSTAR_firstrun
    /home/TIRO/FORTRAN/create_infile_single.Linux_x64 << EOF
    ${REQFILE}
    3
    7
EOF
    #
    # create the jobfile for the third run
    /home/TIRO/FORTRAN/create_jobfile_single.Linux_x64 << EOF
    ${ilow}
    ${iup}
    ironic_FE
    ${ReqDate}
EOF
    chmod 700 ironic*.job
    rm TSTAR_firstrun
    #
fi
if test -s ironic_CO_TSTAR.in
then
    cp /home/TIRO/requests/${ReqDate}/ironic_CO_TSTAR.in ironic_input.dat
    cp /atmosarray/rauch/bin/ironic/linux/ironic_tstar.Linux_n64 jobrun
    nice -19 ./jobrun >& STDOUT
    egrep -i 'error|warning' STDOUT
    rm jobrun
    rm ironic_input.dat
    # now create the in file for the third run with the correct TSTAR values
    cp ~/requests/${ReqDate}/TSTAR_CO.dat TSTAR_firstrun
    /home/TIRO/FORTRAN/create_infile_single.Linux_x64 << EOF
    ${REQFILE}
    3
    8
EOF
    #
    # create the jobfile for the third run
    /home/TIRO/FORTRAN/create_jobfile_single.Linux_x64 << EOF
    ${ilow}
    ${iup}
    ironic_CO
    ${ReqDate}
EOF
    chmod 700 ironic*.job
    rm TSTAR_firstrun
    #
fi
if test -s ironic_NI_TSTAR.in
then
    cp /home/TIRO/requests/${ReqDate}/ironic_NI_TSTAR.in ironic_input.dat
    cp /atmosarray/rauch/bin/ironic/linux/ironic_tstar.Linux_n64 jobrun
    nice -19 ./jobrun >& STDOUT
    egrep -i 'error|warning' STDOUT
    rm jobrun
    rm ironic_input.dat
    # now create the in file for the third run with the correct TSTAR values

```

```

cp ~/requests/${ReqDate}/TSTAR_NI.dat TSTAR_firstrun
/home/TIRO/FORTRAN/create_infile_single.Linux_x64 << EOF
${REQFILE}
3
9
EOF
#
# create the jobfile for the third run
/home/TIRO/FORTRAN/create_jobfile_single.Linux_x64 << EOF
${ilow}
${iup}
ironic_NI
${ReqDate}
EOF
chmod 700 ironic*.job
rm TSTAR_firstrun
#
fi
#
# check if jobfiles are in the directory
#
ReqDir2=/home/TIRO/requests/${ReqDate}
cd ${ReqDir2}
ReqCount=$(find -L ${ReqDir2}/ -mindepth 1 -maxdepth 1 -name 'ironic*.job' | wc -l)
#
# start the third runs
#
if [ $ReqCount -gt 0 ]
then
# start the Ni jobs at first
#
nicount='ls /home/TIRO/requests/${ReqDate}/ironic*Ni*.job 2>/dev/null | head -1'
if test ! -z $nicount
then
cat > Re << eos
while [ \ $# -gt 0 ]
do
/home/TIRO/scripts/startjob.bat \ $1
#
#
shift
done
eos
#
chmod u=u+x Re
./Re `find /home/TIRO/requests/${ReqDate}/ -mindepth 1 -maxdepth 1 -name "ironic*Ni*.job" \
-print | cut -c21-66`
rm ./Re
fi
#
# then start Fe
#
fcount='ls /home/TIRO/requests/${ReqDate}/ironic*FE*.job 2>/dev/null | head -1'
if test ! -z ${fcount}
then
cat > Re << eos
while [ \ $# -gt 0 ]
do
/home/TIRO/scripts/startjob.bat \ $1
#
#
shift
done
eos
#
chmod u=u+x Re
./Re `find /home/TIRO/requests/${ReqDate}/ -mindepth 1 -maxdepth 1 -name "ironic*FE*.job" \
-print | cut -c21-66`
rm ./Re
fi
#

```

```

#
cat > Re << eos
while [ \${#} -gt 0 ]
do
/home/TIRO/scripts/startjob.bat \${1}
#
#
shift
done
eos
#
chmod u=u+x Re
./Re 'find /home/TIRO/requests/${ReqDate}/ -mindepth 1 -maxdepth 1 -name "ironic*.job" \
-print | cut -c21-66'
rm ./Re
#
sleep 100
#
# check if there is a process still running
#
runjobs=' ps -aefl | grep '0 R TIRO' | grep lrOnlc | wc | cut -c1-10'
while [ ${runjobs} -gt 0 ]
do
    sleep 60
    runjobs=' ps -aefl | grep '0 R TIRO' | grep lrOnlc | wc | cut -c1-10'
done
#
# ok no job is running, then create the atomic-data files
cd ~TIRO/requests/${ReqDate}/data/
pwd
~/scripts/ironic_combine_web.job CA ${ReqDate}
~/scripts/ironic_combine_web.job SC ${ReqDate}
~/scripts/ironic_combine_web.job TI ${ReqDate}
~/scripts/ironic_combine_web.job V_ ${ReqDate}
~/scripts/ironic_combine_web.job CR ${ReqDate}
~/scripts/ironic_combine_web.job MN ${ReqDate}
~/scripts/ironic_combine_web.job FE ${ReqDate}
~/scripts/ironic_combine_web.job CO ${ReqDate}
~/scripts/ironic_combine_web.job NI ${ReqDate}
#
#
#
cat ATOM_ION_CA.dat ATOM_ION_SC.dat ATOM_ION_TI.dat ATOM_ION_V_.dat ATOM_ION_CR.dat \
ATOM_ION_MN.dat ATOM_ION_FE.dat ATOM_ION_CO.dat ATOM_ION_NI.dat > ATOMIC_DATA.dat
#
/home/rauch/lrOnlc/ChangeIrrOnlcHeaders
#
# now create a directory accessible via wget and put all results into a tarball
#
mkdir /home/TIRO/public_html/TIRO/results/${ReqDate}
cd /home/TIRO/requests/${ReqDate}/
rm ASC.tgz
tar -czvf ${ReqDate}.tgz data/R*A
cp ${ReqDate}.tgz /home/TIRO/public_html/TIRO/results/${ReqDate}
chmod 777 /home/TIRO/public_html/TIRO/results/${ReqDate}
chmod 644 /home/TIRO/public_html/TIRO/results/${ReqDate}/*tgz
#
# do RBB files exist? if not, there was a problem in the calculation
count1='ls ~TIRO/requests/${ReqDate}/data/RBB* | wc'
count='echo ${count1} | cut -d " " -f1'
if [ ${count} -gt 1 ]
then
pwd
else
# mail to the Administrator
cat > MAIL <<EOF
Dear TIRO Administrator ,

the TIRO run did not produce RBB files for the request from ${ReqDate}.
Check if there is an error in the command files.
EOF

```

```

#
/usr/bin/mail -r "TIRO" -s "TIRO: Error" ironic@listserv.uni-tuebingen.de < MAIL
rm MAIL
#
exit
fi
# now check if there is an error or warning message in the TIRO output file
count2='grep -i error ~TIRO/requests/${ReqDate}/*out | wc | cut -d ' ' -f 7'
if [ ${count2} -ge 1 ]
then
# mail to the Administrator
    cat > MAIL <<EOF
Dear TIRO Administrator ,

there was an error message in the TIRO Output.
Please check this.
EOF
#
/usr/bin/mail -r "TIRO" -s "TIRO: Error" ironic@listserv.uni-tuebingen.de < MAIL
rm MAIL
#
exit
fi
#
# send a mail to notify that the request calculation finished
#
    cat > MAIL <<EOF
Dear ${firstname} ${lastname} ,

the request sent at ${ReqDate} has successfully calculated.
The resulting model atom as well as the RBB and RBF files
were put into a tarball and are accessible via

    wget http://astro.uni-tuebingen.de/~TIRO/TIRO/results/${ReqDate}/${ReqDate}.tgz

If you have further questions ,
please do not hesitate to contact ironic@listserv.uni-tuebingen.de.

When publishing research making use of this service , please acknowledge :
"The TIRO service (http://astro-uni-tuebingen.de/~TIRO) used to calculate
opacities for this paper was constructed as part of the
activities of the German Astrophysical Virtual Observatory."

Yours
TIRO Team
EOF
#
/usr/bin/mail -r "TIRO" -s "TIRO: Results" ${addy} < MAIL
/usr/bin/mail -r "TIRO" -s "TIRO: Results" ironic@listserv.uni-tuebingen.de < MAIL
rm MAIL
#
fi
mv /home/TIRO/requests/running/${REGFILE} /home/TIRO/requests/finished
#####

```



Listing A.3: Die startjob.bat-Prozedur wird aufgerufen, um die einzelnen Querschnittsberechnungen für die Ionen zu starten.

```
#!/bin/bash
#####
# Start the job files generated in run_ions.bat
#
# LAST_MODIFICATION="24-Jan-2013 18:12:27"
#
# Copyright 2012 - 2013 by E. Mueller-Ringat
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#
#####
# CHANGELOG:
# 12.03.2012: created
#
# 25.05.2012: comments inserted
# 12.06.2012: wait until load is small enough before jobs are started
# 14.12.2012: change location, the new user is TIRO instead of ironic
#
#####
# first read the name of the jobfile and the date of the request
#
jobname='echo ${1} | cut -c21-70'
ReqDate='echo ${1} | cut -c1-20'
#
# go to the directory where the jobfile is located
#
cd /home/TIRO/requests/${ReqDate}/
#
# start the jobfile and wait for 2 seconds
# start only if there is enough space
# else wait ten minutes and try again
#
runjobs=' ps -aefl | grep '0 R TIRO' | grep TIRO | wc | cut -c1-10'
while [ ${runjobs} -gt 24 ]
do
    sleep 600
    runjobs=' ps -aefl | grep '0 R TIRO' | grep TIRO | wc | cut -c1-10'
done
echo 'w | head -1' > load.dat
load=~rauch/tools/readload.Linux_X86 | tail -1'
while [ ${load} -gt 24 ]
do
    sleep 600
    echo 'w | head -1' > load.dat
    load=~rauch/tools/readload.Linux_X86 | tail -1'
done
./${jobname} &
sleep 2
```

Listing A.4: Die `ironic_combine_web.job`-Prozedur wird verwendet, um die Atomdaten der einzelnen Ionen zusammenzufügen. Diese Prozedur wurde von der existierenden `ironic_combine.bat`-Prozedur abgewandelt.

```
#!/bin/sh
#####
# This script is used by the run-files to combine the IrOnIc output
# files to one atomic data file (ATOM_ION_elt.dat) for pro2. It is
# a modified version of ironic_combine
#
# LAST_MODIFICATION="24-Jan-2013 18:18:43"
#
# Copyright 2012 - 2013 by E. Mueller-Ringat
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
# CHANGELOG:
# 24.10.2011: created
#
# 12.03.2012: The first line with ATOM and ion in the ATOM_ION_elt.dat
#             file was wrong (choose one ATOM_MAIN file)
#             Only create an output when an ATOM_MAIN file exists
# 14.12.2012: change location, the new user is TIRO instead of ironic
#####
#
# stop if the script is not used correctly
# else read the name of the element and the date of the request
#
case ${1} in
  '') echo "usage: ironic_combine_web.job < element >"; exit;;
  *) atom=${1}; DATE=${2};;
esac
# go to the directory where the atomic data is located
cd /home/TIRO/requests/${DATE}/data/
rm Reneu
rm ATOM_ION_${atom}.dat
#
filename='ls ATOM_MAIN_${atom}*dat | head -1'
#
# put all atomic data information into one file for the element requested
if [ "${filename}" -ne "" ]
then
cat ${filename} > ATOM_ION_${atom}.dat
fi
#
cat > Reneu << eos
while [ \ $# -gt 0 ]
do
  cat ATOM_LEVEL_${atom}\$1*.dat >> ATOM_ION_${atom}.dat
  cat ATOM_RBF_${atom}\$1*.dat >> ATOM_ION_${atom}.dat
  cat ATOM_RLU_${atom}\$1*.dat >> ATOM_ION_${atom}.dat
  cat ATOM_RLL_${atom}\$1*.dat >> ATOM_ION_${atom}.dat
  cat ATOM_CBB_${atom}\$1*.dat >> ATOM_ION_${atom}.dat
  cat ATOM_CBF_${atom}\$1*.dat >> ATOM_ION_${atom}.dat
  cat ATOM_RFF_${atom}\$1*.dat >> ATOM_ION_${atom}.dat
#
shift
done
eos
#
chmod u+wx Reneu
```

```

./Reneu `find /home/TIRO/requests/${DATE}/data -mindepth 1 -maxdepth 1 -name "ATOM_LEVEL_${atom}*" | sort | cut -c61-62 `

rm ./Reneu
#
# write the last ion into the atomic data file
#
lasthelp=`ls /home/TIRO/requests/${DATE}/data/ATOM_LEVEL_${atom}* | sort -r | cut -c61-62`
last=`echo ${lasthelp} | cut -c1-2`
last=`expr ${last} + 1`
#
count=`ls ATOM_MAIN_${atom}*.dat | wc -l`
if [ ${count} -gt 0 ]
then
cat >> ATOM_ION_${atom}.dat <<EOF
L
${atom}0${last}01A  NONE          0.000000000000E+00      1.00000
0
EOF
else
touch ATOM_ION_${atom}.dat
fi

```

Listing A.5: Die `ChangeIrOnIcHeaders`-Prozedur wird verwendet, um die Querschnitte in das von *TMAP* verwendete Format umzuwandeln. Sie existierte bereits vor dieser Arbeit und wurde nicht verändert.

```
#!/bin/sh
#
# procedure to change headers of IrOnIc files to the standard TMAP input format
#
#
# save original IrOnIc data
#
tar -czf ASC.tgz *.asc
#
# convert RBB headers
#
if test -s csc_convert_rbb.in
then
  rm csc_convert_rbb.in
fi
ls -l RBB*.asc | wc -l > csc_convert_rbb.in
echo "true"      >> csc_convert_rbb.in
echo "asc"       >> csc_convert_rbb.in
echo "asc"       >> csc_convert_rbb.in
ls -l RBB*.asc >> csc_convert_rbb.in
/atmosarray/rauch/bin/ironic/linux/csc_convert_rbb
#
# convert RBF headers
#
if test -s csc_convert_rbf.in
then
  rm csc_convert_rbf.in
fi
ls -l RBF*.asc | wc -l > csc_convert_rbf.in
echo "true"      >> csc_convert_rbf.in
echo "asc"       >> csc_convert_rbf.in
echo "asc"       >> csc_convert_rbf.in
ls -l RBF*.asc >> csc_convert_rbf.in
/atmosarray/rauch/bin/ironic/linux/csc_convert_rbf
#
# eliminate extensions
#
/atmosarray/rauch/IrOnIc/asc_without_ext
chmod 640 R*A
chgrp atmos R*A
```

Listing A.6: Die `run_inter.sh`-Prozedur prüft mit Hilfe von `interpol.sh`, ob eine Interpolation der Querschnitte aus Datenbankwerten möglich ist. Ansonsten werden die Querschnitte wie mit `run_ions.bat` berechnet.

```
#!/bin/sh
#####
# Read a request file and check if interpolation between database
# values is possible. Else create the *.in file (old format),
# perform the first run (TSAR), create *.in and *.job files
# for the third run, perform it, combine all model atom parts to one
# model atom, run ChangelrOnlcHeaders, put everything into a tarball
# and write an email to the requestor
#
# LAST_MODIFICATION="24-Jan-2013 18:22:15"
#
# Copyright 2012 - 2013 by E. Mueller-Ringat
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
# CHANGELOG:
# 15.10.2012: changed run_ions.sh to first interpolate between database
#           values and using the user frequency grid
# 14.12.2012: change location, the new user is TIRO instead of ironic
# 17.12.2012: change email text to user
#
#####
#
#
ReqDir=/home/TIRO/requests
cd ${ReqDir}
#
#
ReqName=$(ls -t1 ~TIRO/requests/tes_201* | tail -1)
cp ${ReqName} /home/TIRO/requests/running/
REGFILE='echo ${ReqName}|cut -c21-48'
#
# change name to avoid ":"
#
ReqFile='echo ${REGFILE} | sed -e 's\:\_\g''
#
ReqDate='echo ${ReqFile} | cut -c5-50'
mkdir ${ReqDate}
mkdir ${ReqDate}/data
mv ${ReqName} /home/TIRO/requests/${ReqDate}/
cd /home/TIRO/requests/${ReqDate}
#
# read the request file
#
i=0
while read line
do i=$((i+1))
  readarr[$i]=${line}
  echo "$i : ${readarr[$i]}"
done < $REGFILE
#
#   REGFILE=${readarr[1]}
#   echo $REGFILE
#   firstname=${readarr[2]}
#   lastname=${readarr[3]}
#   addy=${readarr[4]}
#   linet=${readarr[5]}
#   ilow=${readarr[6]}
#   iup=${readarr[7]}
```

```

CAinfo=${readarr[8]}
SCinfo=${readarr[9]}
Tlinfo=${readarr[10]}
Vinfo=${readarr[11]}
CRinfo=${readarr[12]}
MNinfo=${readarr[13]}
FEinfo=${readarr[14]}
COinfo=${readarr[15]}
Nlinfo=${readarr[16]}
FGRID=${readarr[17]}
# first check or create frequency grid
if [ "${FGRID}" == "syn" ]
then
lambdamin=${readarr[18]}
lambdamax=${readarr[19]}
delta=${readarr[20]}
/home/rauch/bimod/prep_conts_man${sys} > F_BASE << eos
${lambdamin} ${lambdamax} ${delta}
eos
${BI}/setf2${sys} 2>/dev/null << eos
${linet}
PRINT ASCII
1.30E+16
-1
30 10
eos
if test -s FGRID
then
cp FGRID /home/TIRO/requests/${ReqDate}/${ReqDate}.setf2
else
/usr/bin/mail -r "TIRO" -s "TIRO: Frequency grid problem,${ReqDate}" \
ironic@listserv.uni-tuebingen.de
fi
cd /home/TIRO/requests/${ReqDate}/
fi
if [ "${FGRID}" == "mod" ]
then
lambdamin1=${readarr[18]}
lambdamax1=${readarr[19]}
points=${readarr[20]}
echo ${lambdamin1}
echo ${lambdamax1}
echo ${points}
/home/TIRO/FORTRAN/fgrid_user.Linux_x64 <<EOF
${lambdamin1} ${lambdamax1} ${points}
EOF
mv output_user.seft2 ${ReqDate}.setf2
fi
if [ "${FGRID}" == "ind" ]
then
frequgrid=${readarr[18]}
echo ${frequgrid}
mv /home/TIRO/requests/upload/${frequgrid} /home/TIRO/requests/${ReqDate}/${ReqDate}.setf2
fi
#
# check if interpolation is possible, interpolate and exit
#
/home/TIRO/scripts/interpol.sh ${CAinfo} ${SCinfo} ${Tlinfo} ${Vinfo} ${CRinfo} ${MNinfo} ${\
FEinfo} ${COinfo} ${Nlinfo} \
${linet} ${ilow} ${iup} ${ReqDate} ${addy} ${REGFILE} ${firstname} ${lastname} 2> ${ReqDate}\
_err > ${ReqDate}_out
exit_code=$?
#
count3='grep -i PGF ~TIRO/requests/${ReqDate}/${ReqDate}_err | wc | cut -d ' ' -f 7'
if [ ${count3} -ge 1 ]
then
# mail to the Administrator
cat > MAIL <<EOF
Dear TIRO Administrator ,

there was an error message in the FORTRAN output.

```

```

Please check this.
EOF
#
/usr/bin/mail -r "TIRO" -s "TIRO: Error" ironic@listserv.uni-tuebingen.de < MAIL
rm MAIL
#
exit
fi
#
# if interpolation is not possible, start calculation on users FGRID
#
if [ ${exit_code} == 3 ]
then
#
# send email to notify that the request calculation starts now
#
    cat > MAIL <<EOF
Dear ${firstname} ${lastname},

the request sent at ${ReqDate} starts calculating now.
You will be informed when it finished and you can
retrieve the results.

If you have further questions,
please do not hesitate to contact ironic@listserv.uni-tuebingen.de.

When publishing research making use of this service, please acknowledge:
"The TIRO service (http://astro.uni-tuebingen.de/~TIRO) used to calculate
opacities for this paper was constructed as part of the
activities of the German Astrophysical Virtual Observatory."

Yours
TIRO Team
EOF
#
/usr/bin/mail -r "TIRO" -s "TIRO: Calculation started" ${addy} < MAIL
/usr/bin/mail -r "TIRO" -s "TIRO: Calculation started" \
ironic@listserv.uni-tuebingen.de < MAIL
rm MAIL
#
#
REQFILE=${ReqDir}/${ReqDate}/${REGFILE}
/home/TIRO/FORTRAN/create_infile.Linux << EOF
${REQFILE}
1
EOF
#
#### now old input files for first run are created
#
cd /home/TIRO/requests/${ReqDate}/
ls
#
# now perform the first run for all elements
#
if test -s ironic_CA_TSTAR.in
then
    cp /home/TIRO/requests/${ReqDate}/ironic_CA_TSTAR.in ironic_input.dat
    cp /atmosarray/rauch/bin/ironic/linux/ironic_tstar.Linux_n64 jobrun
    nice -19 ./jobrun >& STDOUT
    egrep -i 'error|warning' STDOUT
    rm jobrun
    rm ironic_input.dat
    # now create the in file for the third run with the correct TSTAR values
    cp /home/TIRO/requests/${ReqDate}/TSTAR_CA.dat TSTAR_firstrun
    /home/TIRO/FORTRAN/create_infile_single_fgrid.Linux << EOF
    ${REQFILE}
    3
    1
    ${ReqDate}
EOF
#

```

```

# create the jobfile for the third run
/home/TIRO/FORTRAN/create_jobfile_single.Linux << EOF
${ilow}
${iup}
ironic_CA
${ReqDate}
EOF
chmod 700 ironic*.job
rm TSTAR_firstrun
#
fi
if test -s ironic_SC_TSTAR.in
then
cp /home/TIRO/requests/${ReqDate}/ironic_SC_TSTAR.in ironic_input.dat
cp /atmosarray/rauch/bin/ironic/linux/ironic_tstar.Linux_n64 jobrun
nice -19 ./jobrun >& STDOUT
egrep -i 'error|warning' STDOUT
rm jobrun
rm ironic_input.dat
# now create the in file for the third run with the correct TSTAR values
cp ~/requests/${ReqDate}/TSTAR_SC.dat TSTAR_firstrun
/home/TIRO/FORTRAN/create_infile_single_fgrid.Linux << EOF
${REQFILE}
3
2
${ReqDate}
EOF
#
# create the jobfile for the third run
/home/TIRO/FORTRAN/create_jobfile_single.Linux << EOF
${ilow}
${iup}
ironic_SC
${ReqDate}
EOF
chmod 700 ironic*.job
rm TSTAR_firstrun
#
fi
if test -s ironic_TI_TSTAR.in
then
cp /home/TIRO/requests/${ReqDate}/ironic_TI_TSTAR.in ironic_input.dat
cp /atmosarray/rauch/bin/ironic/linux/ironic_tstar.Linux_n64 jobrun
nice -19 ./jobrun >& STDOUT
egrep -i 'error|warning' STDOUT
rm jobrun
rm ironic_input.dat
# now create the in file for the third run with the correct TSTAR values
cp ~/requests/${ReqDate}/TSTAR_TI.dat TSTAR_firstrun
/home/TIRO/FORTRAN/create_infile_single_fgrid.Linux << EOF
${REQFILE}
3
3
${ReqDate}
EOF
#
# create the jobfile for the third run
/home/TIRO/FORTRAN/create_jobfile_single.Linux << EOF
${ilow}
${iup}
ironic_TI
${ReqDate}
EOF
chmod 700 ironic*.job
rm TSTAR_firstrun
#
fi
if test -s ironic_V__TSTAR.in
then
cp /home/TIRO/requests/${ReqDate}/ironic_V__TSTAR.in ironic_input.dat
cp /atmosarray/rauch/bin/ironic/linux/ironic_tstar.Linux_n64 jobrun

```



```

nice -19 ./jobrun >& STDOUT
egrep -i 'error|warning' STDOUT
rm jobrun
rm ironic_input.dat
# now create the in file for the third run with the correct TSTAR values
cp ~/requests/${ReqDate}/TSTAR_V.dat TSTAR_firstrun
/home/TIRO/FORTRAN/create_infile_single_fgrid.Linux << EOF
${REQFILE}
3
4
${ReqDate}
EOF
#
# create the jobfile for the third run
/home/TIRO/FORTRAN/create_jobfile_single.Linux << EOF
${ilow}
${iup}
ironic_V_
${ReqDate}
EOF
chmod 700 ironic*.job
rm TSTAR_firstrun
#
fi
if test -s ironic_CR_TSTAR.in
then
cp /home/TIRO/requests/${ReqDate}/ironic_CR_TSTAR.in ironic_input.dat
cp /atmosarray/rauch/bin/ironic/linux/ironic_tstar.Linux_n64 jobrun
nice -19 ./jobrun >& STDOUT
egrep -i 'error|warning' STDOUT
rm jobrun
rm ironic_input.dat
# now create the in file for the third run with the correct TSTAR values
cp ~/requests/${ReqDate}/TSTAR_CR.dat TSTAR_firstrun
/home/TIRO/FORTRAN/create_infile_single_fgrid.Linux << EOF
${REQFILE}
3
5
${ReqDate}
EOF
#
# create the jobfile for the third run
/home/TIRO/FORTRAN/create_jobfile_single.Linux << EOF
${ilow}
${iup}
ironic_CR
${ReqDate}
EOF
chmod 700 ironic*.job
rm TSTAR_firstrun
#
fi
if test -s ironic_MN_TSTAR.in
then
cp /home/TIRO/requests/${ReqDate}/ironic_MN_TSTAR.in ironic_input.dat
cp /atmosarray/rauch/bin/ironic/linux/ironic_tstar.Linux_n64 jobrun
nice -19 ./jobrun >& STDOUT
egrep -i 'error|warning' STDOUT
rm jobrun
rm ironic_input.dat
# now create the in file for the third run with the correct TSTAR values
cp ~/requests/${ReqDate}/TSTAR_MN.dat TSTAR_firstrun
/home/TIRO/FORTRAN/create_infile_single_fgrid.Linux << EOF
${REQFILE}
3
6
${ReqDate}
EOF
#
# create the jobfile for the third run
/home/TIRO/FORTRAN/create_jobfile_single.Linux << EOF

```

```

    ${ilow}
    ${iup}
    ironic_MN
    ${ReqDate}
EOF
    chmod 700 ironic*.job
    rm TSTAR_firstrun
    #
fi
if test -s ironic_FE_TSTAR.in
then
    cp /home/TIRO/requests/${ReqDate}/ironic_FE_TSTAR.in ironic_input.dat
    cp /atmosarray/rauch/bin/ironic/linux/ironic_tstar.Linux_n64 jobrun
    nice -19 ./jobrun >& STDOUT
    egrep -i 'error|warning' STDOUT
    rm jobrun
    rm ironic_input.dat
    # now create the in file for the third run with the correct TSTAR values
    cp ~/requests/${ReqDate}/TSTAR_FE.dat TSTAR_firstrun
    /home/TIRO/FORTRAN/create_infile_single_fgrid.Linux << EOF
    ${REQFILE}
    3
    7
${ReqDate}
EOF
    #
    # create the jobfile for the third run
    /home/TIRO/FORTRAN/create_jobfile_single.Linux << EOF
    ${ilow}
    ${iup}
    ironic_FE
    ${ReqDate}
EOF
    chmod 700 ironic*.job
    rm TSTAR_firstrun
    #
fi
if test -s ironic_CO_TSTAR.in
then
    cp /home/TIRO/requests/${ReqDate}/ironic_CO_TSTAR.in ironic_input.dat
    cp /atmosarray/rauch/bin/ironic/linux/ironic_tstar.Linux_n64 jobrun
    nice -19 ./jobrun >& STDOUT
    egrep -i 'error|warning' STDOUT
    rm jobrun
    rm ironic_input.dat
    # now create the in file for the third run with the correct TSTAR values
    cp ~/requests/${ReqDate}/TSTAR_CO.dat TSTAR_firstrun
    /home/TIRO/FORTRAN/create_infile_single_fgrid.Linux << EOF
    ${REQFILE}
    3
    8
${ReqDate}
EOF
    #
    # create the jobfile for the third run
    /home/TIRO/FORTRAN/create_jobfile_single.Linux << EOF
    ${ilow}
    ${iup}
    ironic_CO
    ${ReqDate}
EOF
    chmod 700 ironic*.job
    rm TSTAR_firstrun
    #
fi
if test -s ironic_NI_TSTAR.in
then
    cp /home/TIRO/requests/${ReqDate}/ironic_NI_TSTAR.in ironic_input.dat
    cp /atmosarray/rauch/bin/ironic/linux/ironic_tstar.Linux_n64 jobrun
    nice -19 ./jobrun >& STDOUT
    egrep -i 'error|warning' STDOUT

```

```

rm jobrun
rm ironic_input.dat
# now create the in file for the third run with the correct TSTAR values
cp ~/requests/${ReqDate}/TSTAR_NI.dat TSTAR_firstrun
/home/TIRO/FORTRAN/create_infile_single_fgrid.Linux << EOF
${REQFILE}
3
9
${ReqDate}
EOF
#
# create the jobfile for the third run
/home/TIRO/FORTRAN/create_jobfile_single.Linux << EOF
${ilow}
${iup}
ironic_NI
${ReqDate}
EOF
chmod 700 ironic*.job
rm TSTAR_firstrun
#
fi
#
# check if jobfiles are in the directory
#
ReqDir2=/home/TIRO/requests/${ReqDate}
echo ${ReqDir2}
cd ${ReqDir2}
ReqCount=$(find -L ${ReqDir2}/ -mindepth 1 -maxdepth 1 -name 'ironic*.job' | wc -l)
echo ${ReqCount}
#
# start the third runs
#
if [ $ReqCount -gt 0 ]
then
# start the Ni jobs at first
#
nicount='ls /home/TIRO/requests/${ReqDate}/ironic*NI*.job 2>/dev/null | head -1'
if test ! -z $nicount
then
cat > Re << eos
while [ \ $# -gt 0 ]
do
/home/TIRO/scripts/startjob.bat \ $1
shift
done
eos
#
chmod u=u+x Re
./Re 'find /home/TIRO/requests/${ReqDate}/ -mindepth 1 -maxdepth 1 -name "ironic*NI*.job" \
-print | cut -c21-66'
rm ./Re
fi
#
# then start Fe
#
fecount='ls /home/TIRO/requests/${ReqDate}/ironic*FE*.job 2>/dev/null | head -1'
if test ! -z $fecount
then
cat > Re << eos
while [ \ $# -gt 0 ]
do
/home/TIRO/scripts/startjob.bat \ $1
shift
done
eos
#
chmod u=u+x Re
./Re 'find /home/TIRO/requests/${ReqDate}/ -mindepth 1 -maxdepth 1 -name "ironic*FE*.job" \
-print | cut -c21-66'
rm ./Re

```

```

fi
#
#
cat > Re << eos
while [ \${#} -gt 0 ]
do
/home/TIRO/scripts/startjob.bat \${1}
shift
done
eos
#
chmod u=u+x Re
./Re `find /home/TIRO/requests/${ReqDate}/ -mindepth 1 -maxdepth 1 -name "ironic*.job" \
-print | cut -c21-66`
rm ./Re
#
sleep 10
#
# check if there is a process still running:
#
runjobs=` ps -aefl | grep '0 R TIRO' | grep lrOnIc | wc | cut -c1-10`
while [ \${runjobs} -gt 0 ]
do
sleep 60
runjobs=` ps -aefl | grep '0 R TIRO' | grep lrOnIc | wc | cut -c1-10`
done
#
# ok no job is running, then create the atomic-data files
cd ~TIRO/requests/${ReqDate}/data/
pwd
~/scripts/ironic_combine_web.job CA ${ReqDate}
~/scripts/ironic_combine_web.job SC ${ReqDate}
~/scripts/ironic_combine_web.job TI ${ReqDate}
~/scripts/ironic_combine_web.job V_ ${ReqDate}
~/scripts/ironic_combine_web.job CR ${ReqDate}
~/scripts/ironic_combine_web.job MN ${ReqDate}
~/scripts/ironic_combine_web.job FE ${ReqDate}
~/scripts/ironic_combine_web.job CO ${ReqDate}
~/scripts/ironic_combine_web.job NI ${ReqDate}
#
#
cat ATOM_ION_CA.dat ATOM_ION_SC.dat ATOM_ION_TI.dat ATOM_ION_V_.dat ATOM_ION_CR.dat \
ATOM_ION_MN.dat ATOM_ION_FE.dat ATOM_ION_CO.dat ATOM_ION_NI.dat > ATOMIC_DATA.dat
#
/home/rauch/lrOnIc/ChangelrOnIcHeaders
#
# now create a directory accessible via wget and put all results into a tarball
#
mkdir /home/TIRO/public_html/TIRO/results/${ReqDate}
cd /home/TIRO/requests/${ReqDate}/
rm ASC.tgz
tar -czvf ${ReqDate}.tgz data/R*A
cp ${ReqDate}.tgz /home/TIRO/public_html/TIRO/results/${ReqDate}
chmod 777 /home/TIRO/public_html/TIRO/results/${ReqDate}
chmod 644 /home/TIRO/public_html/TIRO/results/${ReqDate}/*tgz
#
# do RBB files exist? if not, there was a problem in the calculation
count1=`ls ~TIRO/requests/${ReqDate}/data/RBB* | wc`
count=`echo $count1 | cut -d " " -f1`
if [ $count -gt 1 ]
then
pwd
else
# mail to the Administrator
cat > MAIL <<EOF
Dear TIRO Administrator ,

the TIRO run did not produce RBB files for the request from ${ReqDate}.
Check if there is an error in the command files.
EOF

```

```

#
/usr/bin/mail -r "TIRO" -s "TIRO: Error" ironic@listserv.uni-tuebingen.de < MAIL
rm MAIL
#
exit
#
fi
# now check if there is an error or warning message in the TIRO output file
count2=`grep -i error ~TIRO/requests/${ReqDate}/*out | wc | cut -d ' ' -f 7`
if [ $count2 -ge 1 ]
then
# mail to the Administrator
cat > MAIL <<EOF
Dear TIRO Administrator ,

there was an error message in the TIRO Output.
Please check this.
EOF
#
/usr/bin/mail -r "TIRO" -s "TIRO: Error" ironic@listserv.uni-tuebingen.de < MAIL
rm MAIL
#
exit
#
fi
#
# send a mail to notify that the request calculation finished
#
cat > MAIL <<EOF
Dear ${firstname} ${lastname} ,

the request sent at ${ReqDate} has successfully calculated.
The resulting model atom as well as the RBB and RBF files
were put into a tarball and are accessible via

wget http://astro.uni-tuebingen.de/~TIRO/TIRO/results/${ReqDate}/${ReqDate}.tgz

If you have further questions ,
please do not hesitate to contact ironic@listserv.uni-tuebingen.de.

When publishing research making use of this service , please acknowledge:
"The TIRO service (http://astro-uni-tuebingen.de/~TIRO) used to calculate
opacities for this paper was constructed as part of the
activities of the German Astrophysical Virtual Observatory."

Yours
TIRO Team
EOF
#
/usr/bin/mail -r "TIRO" -s "TIRO: Results" ${addy} < MAIL
/usr/bin/mail -r "TIRO" -s "TIRO: Results" ironic@listserv.uni-tuebingen.de < MAIL
rm MAIL
#
fi
#
mv /home/TIRO/requests/running/${REGFILE} /home/TIRO/requests/finished
fi
#####

```

Listing A.7: Die `interpol.sh`-Prozedur prüft, ob eine Interpolation der Querschnitte aus der Datenbank auf die vom Nutzer gewünschten Werte möglich ist und führt diese gegebenenfalls durch.

```
#!/bin/sh
#####
# This procedure checks if interpolation between database
# values is possible and starts the interpolation
#
# LAST_MODIFICATION="24-Jan-2013 18:24:18"
#
# Copyright 2012 - 2013 by E. Mueller-Ringat
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
# CHANGELOG:
# 15.11.2012: created
# 14.12.2012: change location, the new user is TIRO instead of ironic
# 17.12.2012: change email text to user
#####
#
abunCA=${2}
typeCA=${3}
kindCA=${4}
abunSC=${6}
typeSC=${7}
kindSC=${8}
abunTI=${10}
typeTI=${11}
kindTI=${12}
abunV_=${14}
typeV_=${15}
kindV_=${16}
abunCR=${18}
typeCR=${19}
kindCR=${20}
abunMN=${22}
typeMN=${23}
kindMN=${24}
abunFE=${26}
typeFE=${27}
kindFE=${28}
abunCO=${30}
typeCO=${31}
kindCO=${32}
abunNI=${34}
typeNI=${35}
kindNI=${36}
Tline=${37}
lflow=${38}
lup=${39}
FGRID=${40}
addy=${41}
REGFILE=${42}
firstname=${43}
lastname=${44}
#
inter="no"
Tup="0"
Tlow="0"
if [ "${Tline}" -ge "20000" ] && [ "${Tline}" -lt "40000" ]
then
    Tup=40000
```

```

Tlow=20000
fi
if [ "${Tline}" -ge "40000" ] && [ "${Tline}" -lt "60000" ]
then
Tup=60000
Tlow=40000
fi
if [ "${Tline}" -ge "60000" ] && [ "${Tline}" -lt "80000" ]
then
Tup=80000
Tlow=60000
fi
if [ "${Tline}" -ge "80000" ] && [ "${Tline}" -lt "100000" ]
then
Tup=100000
Tlow=80000
fi
if [ "${Tline}" -ge "100000" ] && [ "${Tline}" -lt "120000" ]
then
Tup=120000
Tlow=100000
fi
if [ "${Tline}" -ge "120000" ] && [ "${Tline}" -lt "140000" ]
then
Tup=140000
Tlow=120000
fi
if [ "${Tline}" -ge "140000" ] && [ "${Tline}" -lt "160000" ]
then
Tup=160000
Tlow=140000
fi
if [ "${Tline}" -ge "160000" ] && [ "${Tline}" -lt "180000" ]
then
Tup=180000
Tlow=160000
fi
if [ "${Tline}" -ge "180000" ] && [ "${Tline}" -lt "200000" ]
then
Tup=200000
Tlow=180000
fi
# stop if Tline is larger than 200000 or less than 20000
#
if [ "${Tup}" == "0" ]
then
exit 3
fi
if [ "${Tlow}" == "0" ]
then
exit 3
fi
#
if [ "${kindCA}" == "${kindSC}" ] && [ "${kindSC}" == "${kindTI}" ] && \
[ "${kindTI}" == "${kindV_}" ] && [ "${kindV_}" == "${kindCR}" ] && \
[ "${kindCR}" == "${kindMN}" ] && [ "${kindMN}" == "${kindFE}" ] && \
[ "${kindFE}" == "${kindCO}" ] && [ "${kindCO}" == "${kindNI}" ] && [ "${kindNI}" == "\
individual" ]
then
if [ "${typeCA}" == "${typeSC}" ] && [ "${typeSC}" == "${typeTI}" ] && \
[ "${typeTI}" == "${typeV_}" ] && [ "${typeV_}" == "${typeCR}" ] && \
[ "${typeCR}" == "${typeMN}" ] && [ "${typeMN}" == "${typeFE}" ] && \
[ "${typeFE}" == "${typeCO}" ] && [ "${typeCO}" == "${typeNI}" ] && [ "${typeNI}" == "LIN" ]
then
inter="individual"
fi
fi
#
if [ "${kindCA}" == "${kindSC}" ] && [ "${kindSC}" == "${kindTI}" ] && \
[ "${kindTI}" == "${kindV_}" ] && [ "${kindV_}" == "${kindCR}" ] && \
[ "${kindCR}" == "${kindMN}" ] && [ "${kindMN}" == "${kindFE}" ] && \

```

```

[ "${kindFE}" == "${kindCO}" ] && [ "${kindCO}" == "${kindNI}" ] && [ "${kindNI}" == "generic" \
]
then
if [ "${typeCA}" == "${typeSC}" ] && [ "${typeSC}" == "${typeTI}" ] && \
[ "${typeTI}" == "${typeV_}" ] && [ "${typeV_}" == "${typeCR}" ] && \
[ "${typeCR}" == "${typeMN}" ] && [ "${typeMN}" == "${typeFE}" ] && \
[ "${typeFE}" == "${typeCO}" ] && [ "${typeCO}" == "${typeNI}" ] && [ "${typeNI}" == "LIN" ]
then
abundcheck="no"
abundcheck='/home/TIRO/FORTRAN/check.Linux_x64 <<EOF
1
$abunCA
$abunSC
$abunTI
$abunV_
$abunCR
$abunMN
$abunFE
$abunCO
$abunNI
EOF
'
if [ "$abundcheck" == " yes" ]
then
inter="generic"
fi
fi
fi
#
if [ "${kindCA}" == "${kindSC}" ] && [ "${kindSC}" == "${kindTI}" ] && \
[ "${kindTI}" == "${kindV_}" ] && [ "${kindV_}" == "${kindCR}" ] && \
[ "${kindCR}" == "${kindMN}" ] && [ "${kindMN}" == "${kindCO}" ] && \
[ "${kindFE}" != "${kindCO}" ] && [ "${kindCO}" != "${kindNI}" ] && [ "${kindCO}" == "generic" \
]
then
if [ "${typeCA}" == "${typeSC}" ] && [ "${typeSC}" == "${typeTI}" ] && \
[ "${typeTI}" == "${typeV_}" ] && [ "${typeV_}" == "${typeCR}" ] && \
[ "${typeCR}" == "${typeMN}" ] && [ "${typeMN}" == "${typeCO}" ] && \
[ "${typeCO}" == "LIN" ]
then
abundcheck="no"
abundcheck='/home/TIRO/FORTRAN/check.Linux_x64 <<EOF
2
${abunCA}
${abunSC}
${abunTI}
${abunV_}
${abunCR}
${abunMN}
${abunCO}
EOF
'
if [ "${abundcheck}" == " yes" ]
then
inter="genpart"
fi
fi
fi
#
if [ "${inter}" == "no" ]
then
exit 3
fi
#
if [ "${inter}" == "generic" ]
then
# read RBB and RBF and new FGRID and interpolate , move new RBB and RBF to directory
/home/TIRO/FORTRAN/Interpol.Linux_x64 <<EOF
1
${llow}
${lup}

```



```

${Tline}
${Tlow}
${Tup}
${FGRID}
EOF
ReqDate=${FGRID}
cp /ironicdata/database/solar/generic/${Tlow}/ATOM*_CA*.dat \
~TIRO/requests/${ReqDate}/data/
fi
#
if [ "${inter}" == "genpart" ]
then
# read RBB and RBF and new FGRID and interpolate , move new RBB and RBF to directory
/home/TIRO/FORTRAN/Interpol.Linux_x64 <<EOF
2
${llow}
${lup}
${Tline}
${Tlow}
${Tup}
${FGRID}
EOF
# now for Fe and Ni
/home/TIRO/FORTRAN/Interpol.Linux_x64 <<EOF
4
${llow}
${lup}
${Tline}
${Tlow}
${Tup}
${FGRID}
EOF
ReqDate=${FGRID}
cp /ironicdata/database/solar/genpart/${Tlow}/ATOM*_CA*.dat \
~TIRO/requests/${ReqDate}/data/
cp /ironicdata/database/solar/individual/${Tlow}/ATOM*_FE*.dat \
~TIRO/requests/${ReqDate}/data/
cp /ironicdata/database/solar/individual/${Tlow}/ATOM*_NI*.dat \
~TIRO/requests/${ReqDate}/data/
fi
#
if [ "${inter}" == "individual" ]
then
# all elements: read RBB and RBF and new FGRID and interpolate ,
# move new RBB and RBF to directory
/home/TIRO/FORTRAN/Interpol.Linux_x64 <<EOF
3
${llow}
${lup}
${Tline}
${Tlow}
${Tup}
${FGRID}
EOF
cp /ironicdata/database/solar/individual/Ca/${Tlow}/ATOM*_CA*.dat \
~TIRO/requests/${ReqDate}/data/
cp /ironicdata/database/solar/individual/Sc/${Tlow}/ATOM*_SC*.dat \
~TIRO/requests/${ReqDate}/data/
cp /ironicdata/database/solar/individual/Ti/${Tlow}/ATOM*_TI*.dat \
~TIRO/requests/${ReqDate}/data/
cp /ironicdata/database/solar/individual/V/${Tlow}/ATOM*_V*.dat \
~TIRO/requests/${ReqDate}/data/
cp /ironicdata/database/solar/individual/Cr/${Tlow}/ATOM*_CR*.dat \
~TIRO/requests/${ReqDate}/data/
cp /ironicdata/database/solar/individual/Mn/${Tlow}/ATOM*_MN*.dat \
~TIRO/requests/${ReqDate}/data/
cp /ironicdata/database/solar/individual/Fe/${Tlow}/ATOM*_FE*.dat \
~TIRO/requests/${ReqDate}/data/
cp /ironicdata/database/solar/individual/Co/${Tlow}/ATOM*_CO*.dat \
~TIRO/requests/${ReqDate}/data/
cp /ironicdata/database/solar/individual/Ni/${Tlow}/ATOM*_NI*.dat \

```

```

~TIRO/requests/${ReqDate}/data/
fi
#
#
if test -s ~TIRO/requests/${FGRID}/${FGRID}.setf2wrong
then
# mail to the Administrator
cat > MAIL <<EOF
Dear TIRO Administrator ,

there is something wrong with the input frequency grid. Check if its lower limit is 10^12 and \
its upper limit 10^18 and if the points are monotonically increasing.
EOF
#
/usr/bin/mail -r "TIRO" -s "TIRO: Error" ironic@listserv.uni-tuebingen.de < MAIL
rm MAIL
#
exit
fi

# ok RBB and RBF are created, then create the atomic-data files
ReqDate=${FGRID}
cd ~TIRO/requests/${ReqDate}/data/
#
if [ "${inter}" == "generic" ]
then
~/scripts/ironic_combine_web.job CA ${ReqDate}
fi
if [ "${inter}" == "genpart" ]
then
~/scripts/ironic_combine_web.job CA ${ReqDate}
~/scripts/ironic_combine_web.job FE ${ReqDate}
~/scripts/ironic_combine_web.job NI ${ReqDate}
fi
if [ "${inter}" == "individual" ]
then
~/scripts/ironic_combine_web.job CA ${ReqDate}
~/scripts/ironic_combine_web.job SC ${ReqDate}
~/scripts/ironic_combine_web.job TI ${ReqDate}
~/scripts/ironic_combine_web.job V_ ${ReqDate}
~/scripts/ironic_combine_web.job CR ${ReqDate}
~/scripts/ironic_combine_web.job MN ${ReqDate}
~/scripts/ironic_combine_web.job FE ${ReqDate}
~/scripts/ironic_combine_web.job CO ${ReqDate}
~/scripts/ironic_combine_web.job NI ${ReqDate}
fi
#
#
#
cat ATOM_ION_CA.dat ATOM_ION_SC.dat ATOM_ION_TI.dat ATOM_ION_V_.dat ATOM_ION_CR.dat \
ATOM_ION_MN.dat ATOM_ION_FE.dat ATOM_ION_CO.dat ATOM_ION_NI.dat > ATOMIC_DATA.dat
#
#
# now create a directory accessible via wget and put all results into a tarball
#
mkdir /home/TIRO/public_html/TIRO/results/${ReqDate}
cd /home/TIRO/requests/${ReqDate}/
rm ASC.tgz
tar -czvf ${ReqDate}.tgz data/R*A
cp ${ReqDate}.tgz /home/TIRO/public_html/TIRO/results/${ReqDate}
chmod 777 /home/TIRO/public_html/TIRO/results/${ReqDate}
chmod 644 /home/TIRO/public_html/TIRO/results/${ReqDate}/*tgz
#
# do RBB files exist? if not, there was a problem in creation
count1='ls ~TIRO/requests/${ReqDate}/data/RBB* | wc'
count='echo $count1 | cut -d " " -f1'
if [ ${count} -gt 1 ]
then
pwd
else
# mail to the Administrator

```

```

    cat > MAIL <<EOF
Dear TIRO Administrator ,

the TIRO run did not produce RBB files for the request from ${ReqDate}.
Check if there is an error in the command files.
EOF
#
/usr/bin/mail -r "TIRO" -s "TIRO: Error" ironic@listserv.uni-tuebingen.de < MAIL
rm MAIL
#
exit
#
fi
# send a mail to notify that the request calculation finished
#

```

```

    cat > MAIL <<EOF
Dear ${firstname} ${lastname},

the request sent at ${ReqDate} has successfully calculated.
The resulting model atom as well as the RBB and RBF files
were put into a tarball and are accessible via

wget http://astro.uni-tuebingen.de/~TIRO/TIRO/results/${ReqDate}/${ReqDate}.tgz

```

If you have further questions,  
please **do** not hesitate to contact [ironic@listserv.uni-tuebingen.de](mailto:ironic@listserv.uni-tuebingen.de).

When publishing research making use of this service, please acknowledge:  
"The TIRO service (<http://astro.uni-tuebingen.de/~TIRO>) used to calculate  
opacities for this paper was constructed as part of the  
activities of the German Astrophysical Virtual Observatory."

Yours  
TIRO Team  
EOF

```

#
/usr/bin/mail -r "TIRO" -s "TIRO: Results" ${addy} < MAIL
/usr/bin/mail -r "TIRO" -s "TIRO: Results" ironic@listserv.uni-tuebingen.de < MAIL
rm MAIL
#
#
mv /home/TIRO/requests/running/${REGFILE} /home/TIRO/requests/finished
#####

```

Listing A.8: Die Steuerdatei `ironic_CaSc.in`.

```
#####
#
FREE DESCRIPTION OF THE (GENERIC) ATOM:
Ca+Sc, ionization stages II – V (=01–04)
#
NUMBER OF (GENERIC) IONIZATION STAGES:
4
#
# choose 3/4 of Teff
LINE TEMPERATURE:
52500
#
ELECTRON DENSITY 1:
0.0
ELECTRON DENSITY 2:
1.0E16
#
FORMAT OF THE CROSS SECTION FILES (asc/bin):
asc
#
CALCULATE VOIGT PROFILE FOR  $v = (nue - nue0)/delta\_dop \leq v\_max$  []:
1000000000
#
CREATE THE ATOM_LEVEL FILES ONLY (true/false):
false
#
PATH TO THE FREQUENCY GRID FILE:
/home/ringat/IrOnIc/generic_ionisationsstufentest/HCaFe_generic.setf2
#####
IDENTIFIER OF THIS (GENERIC) IONIZATION STAGE:
CA 01 Tstar only
CHARGE:
1
ELEMENT:
CA SC
Z:
20 21
MASS:
40.08 44.96
RATIO:
1 1
T_STAR:
35501.00
SPLIT NIVEAUS BY PARITY (true/false):
false
#
FILE SELECTION INDICATOR (kur_lin/opa_rbb):
kur_lin kur_lin
#
PATH TO THE LEVEL-FILE:
/atom/kuruczData/GF2001.GAM /atom/kuruczData/GF2101.GAM
#
PATH TO THE LINE-FILE:
/atom/kuruczData/GF2001.LIN /atom/kuruczData/GF2101.LIN
#
PATH TO THE OPACITY RBF FILE:
none none
#
BORDER ALL OR EVE:
95752
BORDER EMPTY OR ODD:
95752
#####
IDENTIFIER OF THIS (GENERIC) IONIZATION STAGE:
CA 02
CHARGE:
2
ELEMENT:
CA SC
Z:
```

```

20 21
MASS:
40.08 44.96
RATIO:
1 1
T_STAR:
20936.7
SPLIT NIVEAUS BY PARITY (true/false):
false
#
FILE SELECTION INDICATOR (kur_lin/opa_rbb):
kur_lin kur_lin
#
PATH TO THE LEVEL-FILE:
/atom/kuruczData/GF2002.GAM /atom/kuruczData/GF2102.GAM
#
PATH TO THE LINE-FILE:
/atom/kuruczData/GF2002.LIN /atom/kuruczData/GF2102.LIN
#
PATH TO THE OPACITY RBF FILE:
none none
#
#
BORDER ALL OR EVE:
50000 100000 150000 200000 275000 350000 410642
BORDER EMPTY OR ODD:
50000 100000 150000 200000 275000 350000 410642
#####
IDENTIFIER OF THIS (GENERIC) IONIZATION STAGE:
CA 03
CHARGE:
3
ELEMENT:
CA SC
Z:
20 21
MASS:
40.08 44.96
RATIO:
1 1
T_STAR:
38214.2
SPLIT NIVEAUS BY PARITY (true/false):
false
#
FILE SELECTION INDICATOR (kur_lin/opa_rbb):
kur_lin kur_lin
#
PATH TO THE LEVEL-FILE:
/atom/kuruczData/GF2003.GAM /atom/kuruczData/GF2103.GAM
#
PATH TO THE LINE-FILE:
/atom/kuruczData/GF2003.LIN /atom/kuruczData/GF2103.LIN
#
PATH TO THE OPACITY RBF FILE:
none none
#
#
BORDER ALL OR EVE:
30000 160000 220000 270000 360000 450000 542600
BORDER EMPTY OR ODD:
30000 160000 220000 270000 360000 450000 542600
#####
IDENTIFIER OF THIS (GENERIC) IONIZATION STAGE:
CA 04
CHARGE:
4
ELEMENT:
CA SC
Z:
20 21
MASS:

```

40.08 44.96  
RATIO:  
1 1  
T\_STAR:  
48329.5  
SPLIT NIVEAUS BY PARITY (**true/false**):  
**false**  
#  
FILE SELECTION INDICATOR (kur\_lin/opa\_rbb):  
kur\_lin kur\_lin  
#  
PATH TO THE LEVEL-FILE:  
/atom/kuruczData/GF2004.GAM /atom/kuruczData/GF2104.GAM  
#  
PATH TO THE LINE-FILE:  
/atom/kuruczData/GF2004.LIN /atom/kuruczData/GF2104.LIN  
#  
PATH TO THE OPACITY RBF FILE:  
none none  
#  
BORDER ALL OR EVE:  
97371 194742 292113 389484 486855 584226 681600  
BORDER EMPTY OR ODD:  
97371 194742 292113 389484 486855 584226 681600

Listing A.9: Die Steuerdatei `ironic_CaSc.job`.

```
#!/bin/sh
set +x;. ${HOME}/.jobstart
# do not edit the beginning of this file
#####
## own job following ##
#####
#
# IrOnIc job file
# Version 21-Jan-2011 09:12:13
#
#----- Input Paths & File Definitions: Edit this part only! -----
#
# IRONIC_OPT: ironic_high shifts output ionization stages
#             by one towards higher stages
# MODEL: IrOnIc subdirectory
# ATOM: identification of IrOnIc job and input files
# MAIN_PROGRAM: ironic_tstar for 1st IrOnIc run
#               ironic      for 2nd and 3rd IrOnIc run
# DATA: individual data directory
IRONIC_OPT=ironic
IRONIC_BIN=/atmosarray/TMAD/bin/${IRONIC_OPT}/linux
IRONIC_DIR=/atmosarray/${LOGNAME}/IrOnIc

MODEL=generic_ionisationsstufentest
ATOM=CaSc_generic

MAIN_PROGRAM=${IRONIC_BIN}/ironic_tstar${sys}
#MAIN_PROGRAM=${IRONIC_BIN}/ironic${sys}

RUN_NAME=IrOnIc_${ATOM}

IRONIC_PATH=${IRONIC_DIR}/${MODEL}
DATA=${IRONIC_PATH}/data_CaSc_generic

INPUT=${IRONIC_PATH}/ironic_${ATOM}.in
OUTPUT=${IRONIC_PATH}/ironic_${ATOM}.out

#----- Make Output-Data Directory -----
if test ! -s ${DATA}
then
mkdir -p ${DATA}
fi

#----- Data Allocation -----
cp ${INPUT}          ironic_input.dat
cp ${MAIN_PROGRAM}  ${RUN_NAME}

#----- Main Program Execution -----
clear
echo "'date': executing IrOnIc@${HOSTNAME} (${TMPDIR})"

nice -19 ./${RUN_NAME} >& STDOUT

egrep -i 'error|warning' STDOUT

echo "'date': IrOnIc finished, copying data and cleaning up now"

#----- Data Storing & Clean Up -----
rm ${RUN_NAME}
rm ironic_input.dat
mv STDOUT ${OUTPUT}
mv *      ${DATA}

# do not edit the rest of this file
#####
set +x; ${HOME}/.jobend ${TMPDIR}
```

## B. TMAW-Prozeduren

### B.1. Die Hauptprozeduren

Wie bereits in Kapitel 3.4 beschrieben, werden die Parameter, die über die Webseite eingegeben werden, in Request-Dateien gespeichert. Die Prozedur `checkreq.sh` wird regelmäßig automatisch auf allen Vierkernrechnern des IAAT, sowie auf vier Zweikernrechnern und dem *GAVO* @IAAT-Server ausgeführt. Es überprüft, ob auf dem jeweiligen Rechner genug Rechenkapazität frei ist und garantiert, dass maximal eine bestimmte Anzahl an Requests pro Rechner gestartet wird. Für den Start der Berechnung der einzelnen Request-Dateien ruft es die Prozedur `TMAW_iaat_run.sh` auf. `TMAW_iaat_run.sh` übernimmt die Email-Kommunikation mit dem Nutzer und Administrator und startet die Prozedur `TMAW_iaat_prepare.sh` zur Vorbereitung der Atomdaten und Frequenzgitter sowie zur Überprüfung der Parameter und Erstellung der Programmdatei. Außerdem startet `TMAW_iaat_run.sh` die `TMAW_iaat_jobfile.sh`-Prozedur, die die verschiedenen Rechenschritte zur Erzeugung des Modells und synthetischen Spektrums durchführt. Diese Prozeduren waren schon zu Beginn der Arbeit vorhanden und wurden modifiziert. Da gleichzeitig mehrere *GAVO*-Mitarbeiter an dieser Prozeduren gearbeitet haben, sind die Änderungen von mehreren Personen durchgeführt worden. Sie führen einzelne Programme von *TMAP* aus, die im *TMAP* User's Guide<sup>1</sup> beschrieben werden. Der Stand der Prozeduren am 25.01.2013 (sofern nicht anders angegeben) wird im Folgenden gelistet.

---

<sup>1</sup><http://astro.uni-tuebingen.de/~TMAP/UserGuide/UserGuide.pdf>



Listing A.10: Die `checkreq.sh`-Prozedur zur Überprüfung freier Rechenkapazität und Vermeidung von Mehrfachberechnungen auf dem selben Rechner. Diese Prozedur war bereits zu Beginn dieser Arbeit vorhanden und wurde leicht modifiziert.

```
#!/bin/sh
#####
# This script is used every few minutes in order to check whether
# a TMAW request is waiting
#
# LAST_MODIFICATION="24-Jan-2013 18:27:05"
#
# Copyright 2003-2013 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
# CHANGELOG:
#      30.10.2006 creation of the script
#      26.10.2007 multi-machine usage
#                  one job/machine according to IAAT laws
#      07.03.2008 start mail
#      22.11.2008 introduced grid-job handling
#      24.11.2008 iaat/grid compute-resource selection
#      27.01.2009 name convention changed
#      18.09.2009 bug - mv of running directory - eliminated
#      08.12.2009 ssh problems, request directory changed
#      10.12.2009 "sleeps" for different machines to avoid
#                  same request fetches by different machines
#      28.01.2010 use all BW PC II-type machines (4 cores)
#                  on the IAAT cluster
#                  (cron jobs start checkreq.sh every 5 min),
#                  one job/machine, two jobs each on weekends
#      22.10.2010 change RUNCOUNT to look for *req_* instead
#                  of *req_iaat* because of postGRID-jobs
#      16.12.2010 check if load is < 3 before jobstart
#      24.05.2011 for grid requests write an email for
#                  the first and last processed job only
#      09.06.2011 use only one executable for a grid of models
#                  check if it is there before start
#      22.08.2011 max. 12 jobs can be calculated on aitp20
#      23.08.2011 load is set with ~rauch/tools/readload.Linux_X86
#      15.10.2011 loadmax/runmax error corrected
#      30.11.2011 start only if no previous model is still compiling
#                  (if no bimod/TMAW_preparing exists)
#      07.12.2011 create a TMAW_preparing file after start
#      28.12.2011 if grid requests are present, process them first
#      06.03.2012 TMAW_preparing is generated only when request starts
#      15.03.2012 change head to tail for the ls command that takes
#                  the latestrequest -> take the eldest first
#      17.03.2012 added four u-076 machines
#      21.03.2012 MAINTENANCE break introduced
#                  (use "maintenance" to start,
#                  "maintenance end" to stop )
#      03.05.2012 added ait320
#      08.05.2012 load-control problem solved
#      07.01.2013 error starting grid jobs solved
#      11.01.2013 check TMAW_preparing after a hostspecific waiting time
#####
#
# MAINTENANCE break
#
if test -e /home/TMAW/MAINTENANCE/inprogress
then
```

```

    exit
fi
#
HOSTNAME='hostname'
#
#
# wait some seconds (different waiting times for different machines)
# before checking the available resources to avoid parallel starts
#
case ${HOSTNAME} in
    ait320) sleep 2;;
    ait490) sleep 4;;
    ait780f) sleep 6;;
    aithp1) sleep 8;;
    aithp5) sleep 10;;
    aithp6) sleep 12;;
    aithx2) sleep 14;;
    aitmvx) sleep 16;;
    aitp11) sleep 18;;
    aitp12) sleep 20;;
    aitp22) sleep 22;;
    aitp23) sleep 24;;
    aitp28) sleep 26;;
    aitp29) sleep 28;;
    aitpc1) sleep 30;;
    aitpc8) sleep 32;;
    aitsv6) sleep 34;;
    u-076-c107) sleep 36;;
    u-076-c108) sleep 38;;
    u-076-c109) sleep 40;;
    u-076-c110) sleep 42;;
esac
#
#####
#
#
# in case that a request is compiling,
# do not activate any further request
#
if test -s /home/TMAW/bimod/TMAW_preparing
then
    exit
fi
#
#####
#
# define the directories were the requests files are located
# and the scripts to run the requests
#
    REQDIR=${HOME}/requests
    RUNDIR=${REQDIR}/running
    GRIDDIR=${REQDIR}/GRID
IAAT_RUNSCRIPT=${HOME}/scripts/TMAW_iaat_run.sh
GRID_RUNSCRIPT=${HOME}/scripts/TMAW_grid_run.sh
#
#
# first check if grid requests are present
#
GRID_REQCOUNT=$(find -L ${REQDIR} -mindepth 1 -maxdepth 1 -name 'req_grid_*' | wc -l)
#
#
if [ $GRID_REQCOUNT -lt 1 ]
then
#
#
#####
#
# IAAT calculations
# _____
#
#
#

```

```

# define maximum loads on specific machines
#
day='date | cut -c1-3'
hour='date | cut -c12-13'
#
case ${day} in
  Fri) if [ ${hour} -gt 18 ]
      then
        loadmax=3
      else
        loadmax=2
      fi ;;
  Sat) loadmax=3 ;;
  Sun) if [ ${hour} -lt 12 ]
      then
        loadmax=3
      else
        loadmax=3
      fi ;;
  *) loadmax=2 ;;
esac
case ${HOSTNAME} in
  ait780f) loadmax=2;;
  ait320) loadmax=6;;
  aitp20) loadmax=24;;
esac
#
#
# check actual load on request machine
#
echo 'w | head -1' > load.dat
load=~rauch/tools/readload.Linux_X86 | tail -1
#
case ${load} in
  *) exit ;;
esac
if [ ${load} -ge ${loadmax} ]
then
  exit
fi
#
#
# get number of TMAW requests waiting for calculation on IAAT resources
#
IAAT_REQCOUNT=$(find -L ${REQDIR} -mindepth 1 -maxdepth 1 -name 'req_*' | wc -l)
echo ${IAAT_REQCOUNT}
#
if [ ${IAAT_REQCOUNT} -gt 0 ]
then
  #
  #
  # there are TMAW requests for calculation on IAAT resources
  #
  # get number of running TMAW requests on this machine
  #
  RUNCOUNT=$(ls ${RUNDIR}/*req_*${HOSTNAME} | wc -l)
  #
  #
  # maximum number of TMAW jobs/machine
  #
  day='date | cut -c1-3'
  hour='date | cut -c12-13'
  #
  case ${day} in
    Fri) if [ ${hour} -gt 18 ]
        then
          RUNMAX=3
        else
          RUNMAX=2
        fi ;;
    Sat) RUNMAX=3 ;;
  
```

```

Sun) if [ ${hour} -lt 12 ]
    then
        RUNMAX=3
    else
        RUNMAX=2
    fi
    ;;
*) RUNMAX=2 ;;
esac
case ${HOSTNAME} in
    ait320) RUNMAX=6;;
    aitp20) RUNMAX=24;;
    u-076-c107) RUNMAX=1;;
    u-076-c108) RUNMAX=1;;
    u-076-c109) RUNMAX=1;;
    u-076-c110) RUNMAX=1;;
esac
if [ ${RUNCOUNT} -ge ${RUNMAX} ]
then
    #
    # there are already one/two requests running on this machine
    exit
fi
#
#
# check if the executable is there
#
REQFILE=$(ls -t1 ~/requests/req_* | tail -1)
helpfile='echo $REQFILE | cut -c36-55'
#
#
# start oldest request
#
cd ${REQDIR}
REQFILE=$(ls -t1 req_* | tail -1)
mv ${REQFILE} ${RUNDIR}/${REQFILE}_${HOSTNAME}
REQFILE=${REQFILE}_${HOSTNAME}
#
RUNFILE=${RUNDIR}/${REQFILE}
echo "${REQFILE}" >> ${RUNFILE}
cp ${RUNFILE} ${GRIDDIR}/${REQFILE}_named
#
#
namedlines='wc -l ${GRIDDIR}/${REQFILE}_named | cut -c1-2'
if [ ${namedlines} -ne 18 ]
then
    echo "error in request file ${GRIDDIR}/${REQFILE}_named"
    exit
fi
#
# now check again if another requests started meanwhile and is compiling
ssh aitmac ls -l ${BI}/TMAW_preparing > TMAW_preparing.check
if test -s TMAW_preparing.check
then
    exit
else
    echo '${HOSTNAME}' > /home/TMAW/bimod/TMAW_preparing
    echo '${HOSTNAME}' > /home/TMAW/bimod/TMAW_preparing@${HOSTNAME}
fi
#
# nice -19 ${IAAT_RUNSCRIPT} < ${GRIDDIR}/${REQFILE}_named
nice -19 ${IAAT_RUNSCRIPT} < ${GRIDDIR}/${REQFILE}_named
> OUTPUT_CHECK1_${HOSTNAME}
rm ${GRIDDIR}/${REQFILE}_named
rm ${RUNDIR}/${REQFILE}_${HOSTNAME}
#
#
# end of IAAT part
fi
#
#
#

```

```

#####
#
# AstroGrid-D calculations
# -----
#
# get number of TMAW requests waiting for calculation on AstroGrid-D resources
#
else
  GRID_REQCOUNT=$(find -L ${REQDIR} -mindepth 1 -maxdepth 1 -name 'req_grid_*' | wc -l)
#
if [ ${GRID_REQCOUNT} -gt 0 ]
then
  switch=0
  # there are TMAW requests for calculation on AstroGrid-D resources
  #
  #
  RUNCOUNT=$(ls ${RUNDIR}/*req_*${HOSTNAME} | wc -l)
  #
  #
  # define the maximum number of TMAW jobs/machine
  #
  day='date | cut -c1-3'
  hour='date | cut -c12-13'
  #
  case ${day} in
    Fri) if [ ${hour} -gt 18 ]
          then
            RUNMAX=3
          else
            RUNMAX=2
          fi ;;
    Sat) RUNMAX=4 ;;
    Sun) if [ ${hour} -lt 12 ]
          then
            RUNMAX=4
          else
            RUNMAX=3
          fi ;;
    *) RUNMAX=2 ;;
  esac
  case ${HOSTNAME} in
    aitmvx) RUNMAX=4;;
    aitp20) RUNMAX=24;;
  esac
  if [ ${RUNCOUNT} -ge ${RUNMAX} ]
  then
    #
    # there are already one/two requests running on this machine
    exit
  fi
  #
  #
  # start oldest request
  #
  cd ${REQDIR}
  REQFILE=$(ls -t1 req_grid_* | tail -1)
  REQFILE2=$(ls -t1 req_grid_* | tail -1)
  #
  mainname='echo ${REQFILE} | cut -c16-36'
  #
  if test -s ${mainname}.test
  then
    cat ${mainname} > ${mainname}.test
    switch=1
  fi
  if [ ${GRID_REQCOUNT} -eq 2 ]
  then
    switch=2
    rm ${mainname}.test
  fi
  cp ${REQFILE} ${RUNDIR}/${REQFILE}

```

```

mv ${REQFILE} ${GRIDDIR}/${REQFILE}_${HOSTNAME}
#
RUNFILE=${GRIDDIR}/${REQFILE}_${HOSTNAME}
cp ${RUNFILE} ${GRIDDIR}/${REQFILE}_named
echo "${REQFILE}" >> ${GRIDDIR}/${REQFILE}_named
#
#
# inform the user about the submission to AstroGrid-D resources
#
if [ $switch -eq 1 ]
then
echo "The first job request ${REQFILE} was submitted to AstroGrid-D." > MAIL
/usr/bin/mail -r "TMAW" -s "TMAW INFO: Job submitted to AstroGrid-D" -a ${RUNFILE} \
astro-tmaw@listserv.uni-tuebingen.de < MAIL
rm MAIL
fi
if [ $switch -gt 1 ]
then
echo "The last job request ${REQFILE} was submitted to AstroGrid-D." > MAIL
/usr/bin/mail -r "TMAW" -s "TMAW INFO: Job submitted to AstroGrid-D" -a ${RUNFILE} \
astro-tmaw@listserv.uni-tuebingen.de < MAIL
rm MAIL
fi
#
nice -19 ${GRID_RUNSCRIPT} < ${GRIDDIR}/${REQFILE}_named
> ${HOME}/requests/OUTPUT_CHECK
rm ${GRIDDIR}/${REQFILE}_named
mv ${GRIDDIR}/${REQFILE}_${HOSTNAME} ${GRIDDIR}/${REQFILE}
#
#
# end of GRID part
fi
fi

```

Listing A.11: TMAW\_iaat\_run.sh, die eigentliche Hauptprozedur. Auch sie war bereits vor dieser Arbeit vorhanden und wurde modifiziert. Stand: November 2012.

```
#!/bin/bash
set +x;. ${HOME}/scripts/runstart.sh
#####
#
# Procedure is started by a TMAW cronjob. It uses TMAW_iaat_prepare.sh
# to produce model atoms, frequency grid files, and executables. Then it
# starts TMAW_iaat_jobfile.sh to calculate the model and SEDs.
# It also communicates via email (job started, deviations of Teff,
# error happened while model was calculated), creates plot files and
# archives results into a tarball
#
#
# LAST_MODIFICATION="30-Aug-2012 09:03:20"
#
# Copyright 2006-2012 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
# CHANGELOG:
# 26.10.2006 first version of the script
# 31.10.2006 new runstart.sh for second tmpdir
# tar and mail results
# 16.11.2006 added wavelength range input by user
# 19.12.2006 changed $name to global variable
# 08.01.2007 added plotting of T-structure
# 09.01.2007 added CNO abundances
# 21.02.2007 general revision
# 10.03.2007 test phase inside firewall
# 29.10.2007 four-machine tests
# 02.11.2007 output is compressed and then sent to user
# 25.11.2008 user resolution
# 10.02.2009 test to use the whole IAAT cluster
# 04.01.2010 default 64-bit (x64) compilation
# 27.01.2010 astro-tmaw members excluded from info mails
# 28.01.2010 use all BW PC II-type machines (4 cores)
# on the IAAT cluster
# (cron jobs start checkreq.sh every 5 min),
# one job/machine, two jobs each on weekends
# 01.02.2010 save only converged models (.dat .meta)
# and submit new entry to TheoSSA
# 04.02.2010 copy request file in working directory
# 13.10.2010 make Hemassfraction a global variable
# 06.09.2011 check if the request is above the Eddington limit
# 22.11.2011 request started email after Eddington limit check
# instead in checkreq.sh
# 22.11.2011 the meta data (${VODIR}/${name}*.meta) is
# added to the tgz file
# 23.11.2011 Job Finished emails are not sent any more
# 24.11.2011 T-structure.ps, TMAW_out, and flux file get the DATE
# information additionally
# 28.11.2011 tar and send is shifted to the end of the file
# tgz files in the public_html directory get timestamps
# $DATE is written before "WFP" instead of afterwards
# the request file is put into the public_html dir too
# 01.12.2011 job started emails are only sent for the first request
# then a file 'calculation started' exists
# The final results remove this file again
# 08.12.2011 a parameter overview is listed in the started email
# if 'Segmentation' or 'killed' appears in the ERROR file an email
# notification is sent
```

```

#      12.12.2011 The last calculated Teff is compared to the requested
#                  value and sent by mail
#      14.12.2011 Email notification if 'PARA' appears in the output file
#      20.12.2011 Email notification if the last iteration is not number 15
#                  The Email notification for PARA is changed to
#                  'Error in PARAMETER set'
#      12.01.2012 meta files are not any longer in the tar file
#                  the tar part was executed in the wrong directory
#      10.04.2012 if a result file already exists, the TMAW_preparing file is deleted
#      14.05.2012 if there is no WFP file, set job_exit_code=3
#                  the name of the request file is given to the jobfile too
#      21.05.2012 additional error check (PARAMETER too low)
#      25.05.2012 additional comments inserted
#      18.06.2012 changing -cvf all_models.tar to -cvPf
#      28.08.2012 if the public_html results/DATE directory does not exist,
#                  create it
#                  use a dummy.plot file if Display_request_${DATE}.plot
#                  does not exist
#      30.08.2012 failed emails go to the user again
#####
#
# define the scripts to create model atoms, frequency grids and executable
# and to perform the single calculation steps
#
  jobfile=${HOME}/scripts/TMAW_iaat_jobfile.sh
  preparejob=${HOME}/scripts/TMAW_iaat_prepare.sh
#
# define the directories were the request file can be located
# and were the resulting files will be located
#
  REQDIR=${HOME}/requests
  FAILDIR=${REQDIR}/failed
  FINDIR=${REQDIR}/finished
  RESDIR=${HOME}/jobs/TMAP_WEB
  RESDIR2=${HOME}/results
  VODIR=${HOME}/public_html/VO/fluxtables/H+He+C+N+O
  WFPDIR=${HOME}/WFP
#
export RESDIR2
export RUNTMPDIR=${RUNTMPDIR}
#
# read the request file
i=0
while read line
do i=$((i+1))
    readarr[$i]=$line
    echo "$i : ${readarr[$i]}"
done
#
    logg=${readarr[1]}
    teff=${readarr[2]}
    addy=${readarr[3]}
    NAME_first=${readarr[4]}
    NAME_last=${readarr[5]}
    DATE=${readarr[6]}
    ABUNDHin=${readarr[7]}
    ABUNDHEin=${readarr[8]}
    ABUNDCin=${readarr[9]}
    ABUNDNin=${readarr[10]}
    ABUNDOin=${readarr[11]}
    WAVBEG=${readarr[12]}
    WAVEND=${readarr[13]}
    WAVSTP=${readarr[14]}
    FLUXINTERVAL=${readarr[15]}
    REGFILE=${readarr[16]}
##
  REQFILE=${REQDIR}/running/${REGFILE}
  cp ${REQFILE} .
#
#
echo ${teff} > TEFF_l7.in

```



```

TEFF= '/home/rauch/bimod/TEFF_I7${sys} < TEFF_I7.in 2>/dev/null 1>&1'
echo ${logg} > LOGG.in
LOGG= '/home/rauch/bimod/LOGG${sys} < LOGG.in 2>/dev/null 1>&1'
rm TEFF_I7.in LOGG.in
#
# check if requested parameters are located above Eddington limit
Eddi='${HOME}/bin/checkeddington.Linux_x64 << EOF
${TEFF} ${LOGG}
EOF
'
#
# check if this is a single request or a grid of requests
REGFILEPART='echo ${REGFILE} | cut -c15-34'
Reqnr1=$(ls ~/requests/req_*${REGFILEPART} | wc -l)
Reqnr2=$(ls ~/requests/running/req_*${REGFILEPART}* | wc -l)
Reqnr='expr $Reqnr1 + $Reqnr2'
#
if [ ${Eddi} -le 0 ]
then
# mv ${REQFILE} ~/requests/failed_eddi/
cat > MAIL << EOF
Dear ${NAME_first} ${NAME_last},

a star with the requested parameters is located above the Eddington limit.
The model-atmosphere calculation with a static code like TMAP
is numerically instable and not yet implemented via TMAW.
Thus, your request had to be rejected.

Yours
TMAW Team
EOF
#
if [ ${Reqnr} -le 1 ]
then
echo $addy
/usr/bin/mail -r "TMAW" -s "TMAW INFO: Eddington limit exceeded, TMAW request rejected" \
-a ${REQFILE} ${addy} < MAIL
case ${addy} in
rauch@astro.uni-tuebingen.de) /usr/bin/mail -r "TMAW" -s "TMAW INFO: \
Eddington limit exceeded, TMAW request rejected" -a ${REQFILE} \
astro-tmaw@listserv.uni-tuebingen.de < MAIL ;;
ringat@astro.uni-tuebingen.de) /usr/bin/mail -r "TMAW" -s "TMAW INFO: \
Eddington limit exceeded, TMAW request rejected" -a ${REQFILE} \
astro-tmaw@listserv.uni-tuebingen.de < MAIL ;;
*) /usr/bin/mail -r "TMAW" -s "TMAW INFO: \
Eddington limit exceeded, TMAW request rejected" -a ${REQFILE} \
astro-tmaw@listserv.uni-tuebingen.de < MAIL ;;
esac
rm MAIL
fi
mv ${REQFILE} ~/requests/failed_eddi/
exit
fi
# check if the /home/TMAW/public_html/TMAW/results/${DATE} directory exists
if test ! -s /home/TMAW/public_html/TMAW/results/${DATE}
then
mkdir
/home/TMAW/public_html/TMAW/results/${DATE}/
fi
# requests start only if they are below the Eddington limit
# and start only if the started file does not exist -> only one email for a grid of models
#
i=0
while read line
do
variable[$i]=0
variable[$i]='echo $line'
echo ${variable[$i]}
i='expr $i + 1'
done < /home/TMAW/public_html/TMAW/results/${DATE}/request_overview
echo $variable

```

```

if test ! -s /home/TMAW/public_html/TMAW/results/${DATE}/calculation_started
then
  echo 'The calculation started for the request from ${DATE}' > \
/home/TMAW/public_html/TMAW/results/${DATE}/calculation_started
  echo "The job request ${REGFILE} has started at IAAT." > MAIL
  echo "The request was sent at ${DATE}" >> MAIL
  echo "log g = ${variable[0]} - ${variable[1]} " >> MAIL
  echo "in steps of ${variable[2]} " >> MAIL
  echo "Effective temperature = ${variable[3]} - ${variable[4]} " >> MAIL
  echo "in steps of ${variable[5]} " >> MAIL
  echo "H=${ABUNDHin} " >> MAIL
  echo "HE=${ABUNDHEin} " >> MAIL
  echo "C=${ABUNDCin} " >> MAIL
  echo "N=${ABUNDNin} " >> MAIL
  echo "O=${ABUNDOin} " >> MAIL

  /usr/bin/mail -r "TMAW" -s "TMAW INFO: Job started (${HOSTNAME})" -a \
${REQFILE} ${addy} < MAIL
  /usr/bin/mail -r "TMAW" -s "TMAW INFO: Job started (${HOSTNAME})" -a \
${REQFILE} astro-tmaw@listserv.uni-tuebingen.de < MAIL
  rm MAIL
fi
#
#
#
# normalize abundances (for model name)
#
echo "5 1 ${ABUNDHin} ${ABUNDHEin} ${ABUNDCin} ${ABUNDNin} ${ABUNDOin}" > normH
echo "5 2 ${ABUNDHin} ${ABUNDHEin} ${ABUNDCin} ${ABUNDNin} ${ABUNDOin}" > normHE
echo "5 3 ${ABUNDHin} ${ABUNDHEin} ${ABUNDCin} ${ABUNDNin} ${ABUNDOin}" > normC
echo "5 4 ${ABUNDHin} ${ABUNDHEin} ${ABUNDCin} ${ABUNDNin} ${ABUNDOin}" > normN
echo "5 5 ${ABUNDHin} ${ABUNDHEin} ${ABUNDCin} ${ABUNDNin} ${ABUNDOin}" > normO
#
ABUNDH='/home/rauch/bimod/TMAW_norm${sys} < normH 2>/dev/null 1>&1'
ABUNDHE='/home/rauch/bimod/TMAW_norm${sys} < normHE 2>/dev/null 1>&1'
ABUNDC='/home/rauch/bimod/TMAW_norm${sys} < normC 2>/dev/null 1>&1'
ABUNDN='/home/rauch/bimod/TMAW_norm${sys} < normN 2>/dev/null 1>&1'
ABUNDO='/home/rauch/bimod/TMAW_norm${sys} < normO 2>/dev/null 1>&1'
rm normH normHE normC normN normO
#
export HEmassfraction=${ABUNDHE}
name=${TEFF}_${LOGG}_${H}_${ABUNDH}_HE_${ABUNDHE}_C_${ABUNDC}_N_${ABUNDN}_O_${ABUNDO}
export ${name}
export namepart=__H_${ABUNDH}_HE_${ABUNDHE}_C_${ABUNDC}_N_${ABUNDN}_O_${ABUNDO}
#
#
# check if a model with the same parameters exists
# if yes, copy the results to the user accessible directory,
# create an overview plot, and inform via email
#
# user choice
Wa=${WAVBEG}
We=${WAVEND}
#
if [ ${Wa} -lt 10000 ]
then
  if [ ${Wa} -lt 1000 ]
  then
    if [ ${Wa} -lt 100 ]
    then
      if [ ${Wa} -lt 10 ]
      then
        Wanf="0000${Wa}"
      else
        Wanf="000${Wa}"
      fi
    else
      Wanf="00${Wa}"
    fi
  else
    Wanf="0${Wa}"
  fi

```

```

    fi
else
    Wanf="${Wa} "
fi
#
if [ ${We} -lt 10000 ]
then
    if [ ${We} -lt 1000 ]
    then
        if [ ${We} -lt 100 ]
        then
            if [ ${We} -lt 10 ]
            then
                Wend="0000${We} "
            else
                Wend="000${We} "
            fi
        else
            Wend="00${We} "
        fi
    else
        Wend="0${We} "
    fi
fi
else
    Wend="${We} "
fi
#
USERINTERVAL=${Wanf}-${Wend}
#
#
# individual choice of wavelength ranges
#
if [ ${Reqnr} -le 1 ]
then
if test -s ${RESDIR2}/${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}*.tgz
then
    cat > MAIL << EOF
Dear ${NAME_first} ${NAME_last},

your TMAW request (${DATE}) was successfully completed.
The result files can be retrieved via
wget http://astro.uni-tuebingen.de/~TMAW/TMAW/results/${DATE}/all_models.tar .

If you have further questions,
please do not hesitate to contact astro-tmaw@listserv.uni-tuebingen.de.

When publishing research making use of this service, please acknowledge:
"The TMAW service (http://astro.uni-tuebingen.de/~TMAW) used to calculate
theoretical spectra for this paper was constructed as part of the
activities of the German Astrophysical Virtual Observatory."

Yours
TMAW Team
EOF
#
rm ~/bimod/TMAW_preparing*
chmod 777 ~/public_html/TMAW/results/${DATE}/
##
cp ${RESDIR2}/${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}.tgz \
~/public_html/TMAW/results/${DATE}/${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}_${DATE}.tgz
cp ${REQFILE} ~/public_html/TMAW/results/${DATE}/
#
chmod 644 ~/public_html/TMAW/results/${DATE}/re*
chmod 644 ~/public_html/TMAW/results/${DATE}/${name}*.tgz
#
#
ls ~/public_html/TMAW/results/${DATE}/*${namepart}*${FLUXINTERVAL}_${USERINTERVAL}.tgz \
| cut -c21-32 > /home/TMAW/public_html/TMAW/results/${DATE}/results
#
sed -e 's\_ \ \g' /home/TMAW/public_html/TMAW/results/${DATE}/results > \

```

```

/home/TMAW/public_html/TMAW/results/${DATE}/results2
#
if test ! -s /home/TMAW/public_html/TMAW/plots/Display_request_${DATE}.plot
then
cp /home/TMAW/public_html/TMAW/plots/Display_request_dummy.plot Display_result.plot
else
cp /home/TMAW/public_html/TMAW/plots/Display_request_${DATE}.plot Display_result.plot
fi
#
sed -e 's\END\ \g' Display_result.plot > Display_results.plot
echo ' N=? PLOTSYMBOL=2 PEN=3 XYTABLE SELECT 1 2 SYMBOLSIZE=-.3 COLOR=1' >> \
  Display_results.plot
echo 'COMMAND XLOG' >> Display_results.plot
echo 'COMMAND INCLUDE results2' >> Display_results.plot
echo 'FINISH' >> Display_results.plot
echo 'END' >> Display_results.plot
#
cp /atmosarray/TMAW/models/Edli_H.dat .
cp /home/TMAW/public_html/TMAW/results/${DATE}/results2 .
/home/rauch/wrplot.dir/proc.dir/wrps.com Display_results 2>/dev/null
#
ps2pdf Display_results.ps Display_results.pdf
mv Display_results.pdf /home/TMAW/public_html/TMAW/results/${DATE}/
chmod 664 /home/TMAW/public_html/TMAW/results/${DATE}/Display_results.pdf
#
tar -cvPf all_models.tar /home/TMAW/public_html/TMAW/results/${DATE}/*tgz
mv all_models.tar /home/TMAW/public_html/TMAW/results/${DATE}/
chmod 644 /home/TMAW/public_html/TMAW/results/${DATE}/all_models.tar
#
rm /home/TMAW/public_html/TMAW/results/${DATE}/calculation_started
#
#
/usr/bin/mail -r "TMAW" -s "TMAW: Final Results" -a \
/home/TMAW/public_html/TMAW/results/${DATE}/Display_results.pdf ${addy} < MAIL
case ${addy} in
    rauch@astro.uni-tuebingen.de) ;;
    ringat@astro.uni-tuebingen.de) ;;
    *) /usr/bin/mail -r "TMAW" -s "TMAW INFO: \
Final Results (${HOSTNAME})" -a \
/home/TMAW/public_html/TMAW/results/${DATE}/Display_results.pdf \
astro-tmaw@listserv.uni-tuebingen.de < MAIL ;;
esac
mv ${REQFILE} ${FINDIR}
rm MAIL
exit
fi
else
if test -s ${RESDIR2}/${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}.tgz
then
cp ${RESDIR2}/${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}.tgz \
~/public_html/TMAW/results/${DATE}/${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}_${DATE}.tgz
cp ${REQFILE} ~/public_html/TMAW/results/${DATE}/
chmod 777 ~/public_html/TMAW/results/${DATE}/
#
chmod 644 ~/public_html/TMAW/results/${DATE}/re*
chmod 644 ~/public_html/TMAW/results/${DATE}/${name}_LF_*${DATE}.tgz
rm ~/bimod/TMAW_preparing
#
cat > MAIL <<EOF
Dear ${NAME_first} ${NAME_last},

the model ${name}_LF from your TMAW request ${DATE} was successfully calculated.
The result file can be retrieved via

wget \
http://astro.uni-tuebingen.de/~TMAW/TMAW/results/${DATE}/ \
${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}_${DATE}.tgz

You will be notified at the end of your request calculations to download all models at once.
If you have further questions,
please do not hesitate to contact astro-tmaw@listserv.uni-tuebingen.de.

```



```

#
# After reading the parameters start jobfile
#
${jobfile} ${teff} ${logg} ${WAVBEG} ${WAVEND} ${WAVSTP} ${DATE} ${name} \
${REGFILE} 2> ERROR_${DATE} > OUTPUT_${DATE}
#
#
job_exit_code=$?
#
# check if the calculation stopped before the usual 15 iterations have been performed
#
if test ! -s /home/TMAW/jobs/TMAP_WEB/${name}_LF_${DATE}.WFP_03000-55000
then
job_exit_code=3
fi
#
# check the effective temperature of the model as test for the quality of the calculation
#
Teffout='grep integr OUTPUT_${DATE} | tail -n 1 | cut -c82-88'
deviation='bc -l <<< "scale=4; 1-${Teffout}/${teff}"'
cat > MAIL <<EOF
The calculation finished.
If no further email arrives that the job stopped before iteration 15
or that parameter or LINE1_PROF problems occurred, everything went well.

The deviation of the effective temperature of the model
(model Teff / nominal Teff)-1 is ${deviation}.
EOF
/usr/bin/mail -r "TMAW" -s "TMAW QC: Calculation finished, deviation" -a \
OUTPUT_${DATE} ringat@astro.uni-tuebingen.de < MAIL
rm MAIL
#
# check if an error with the parameters occurred in any of the calculation steps
#
count='grep -c 'PARAMETER too low' ERROR_${DATE}'
if [ $count -gt 0 ]
then
cat > MAIL << EOF
Something went wrong because the parameters are too small.
EOF
/usr/bin/mail -r "TMAW" -s "TMAW Problem: PARAMETER wrong" \
astro-tmaw@listserv.uni-tuebingen.de < MAIL
rm MAIL
fi
count='grep -c 'Error in PARAMETER set' OUTPUT_${DATE}'
if [ $count -gt 0 ]
then
cat > MAIL << EOF
Something went wrong because the parameters are too small.
EOF
/usr/bin/mail -r "TMAW" -s "TMAW Problem: PARAMETER wrong" \
astro-tmaw@listserv.uni-tuebingen.de < MAIL
rm MAIL
fi
count='grep -c Segmentation ERROR_${DATE}'
if [ $count -gt 0 ]
then
cat > MAIL << EOF
Something went wrong because the parameters are too small.
EOF
/usr/bin/mail -r "TMAW" -s "TMAW Problem: PARAMETER wrong" \
astro-tmaw@listserv.uni-tuebingen.de < MAIL
rm MAIL
fi
count='grep -c Killed ERROR_${DATE}'
if [ $count -gt 0 ]
then
cat > MAIL << EOF
Something went wrong because the parameters are too big.
EOF
/usr/bin/mail -r "TMAW" -s "TMAW Problem: PARAMETER wrong" \

```

```
astro-tmaw@listserv.uni-tuebingen.de < MAIL
rm MAIL
fi
```

```
# test if job ran ok
#
if [ $job_exit_code == 3 ]
then
#
echo "Job failed"
#
cat > MAIL <<EOF
```

Dear \${NAME\_first} \${NAME\_last},  
your job request send to us via TMAW at \${DATE} failed.  
It is most likely that the combination of requested  
photospheric parameters did not produce a stable model.  
If you have further questions  
please **do** not hesitate to contact astro-tmaw@listserv.uni-tuebingen.de.  
Yours  
TMAW Team  
EOF

```
/usr/bin/mail -r "TMAW" -s "TMAW INFO: request failed (${HOSTNAME})" -a \
${REQFILE} astro-tmaw@listserv.uni-tuebingen.de < MAIL
/usr/bin/mail -r "TMAW" -s "TMAW INFO: request failed (${HOSTNAME})" -a \
${REQFILE} ${addy} < MAIL
```

```
rm MAIL
mv ${REQFILE} ${FAILDIR}
#
exit
#
```

```
else
#
#
echo "Job finished successfully"
#
```

```
if test -s DATABASE
then
DATABASE=YES
else
DATABASE=NO
fi
cp OUTPUT_${DATE} OUTPUT
tar -czf OUTPUT.tgz OUTPUT
```

```
#
#
#
#####
```

```
# #####
# # # ##### ## #####
# # # # # # # # #
# ##### # # # # # # # #
# # # # # # # # # # #
# # # # # # # # # # #
# # ##### ##### # ## # ##### #####
#
```

```
#####
# Making plotfile and plotting flux
#
```

```
cd ${RESDIR}
#
WINT='expr ${WAVEND} - ${WAVBEG}'
WTCK='expr ${WINT} / 100'
if [ ${WTCK} -lt 500 ]
then
if [ ${WTCK} -lt 100 ]
then
if [ ${WTCK} -lt 10 ]
then
WTCK=10
WLAB=100
else
```

```

        WTCK=100
        WLAB=1000
    fi
    else
        WTCK=200
        WLAB=2000
    fi
    else
        WTCK=500
        WLAB=5000
    fi
#
#
    cat > ${name}_LF_${DATE}_flux.plot <<EOF
PAPERFORMAT A4Q
PLOT :
\OFS 2.5 2.5
\FONT=HELVET
\LETTERSIZ=0.2
\NOCOPYRIGHT
\INBOX
\LATEBOX
\PAUSE
\COLOR=1
\DATE
\FILE
HEADER :\CENTER\ TMAW: ${name}_LF_${DATE} flux
X-ACHSE:\CENTER\ wavelength / \A
Y-ACHSE:\CENTER\ log &RF&N&T#l#&M / erg cm&H-2&M sec&H-1&M cm&H-1&M
      MASSTAB  MINIMUM  MAXIMUM  TEILUNGEN  BESCHRIFT.  DARUNTER
X: 25CM      ${WAVBEG}  ${WAVEND}    ${WTCK}    ${WLAB}    0.
Y: AUTO 16CM      14.4      19.2      .1      1.      0.
N= ? PLOTSYMBOL 5 PEN 2 XYTABLE SELECT 1 2 COLOR=2
COMMAND Y-CUT 1.0E-33 1.E38
COMMAND YLOG
COMMAND INCLUDE "${name}_LF_${DATE}.WFP"
FINISH
\PEN=3
END
EOF
#
/home/rauch/wrplot.dir/proc.dir/wrps.com ${name}_LF_${DATE}_flux
#
# Making plotfile and plotting temperature structure
#
    cat > ${name}_LF_${DATE}_T-structure.plot <<EOF
PAPERFORMAT A4Q
PLOT :
\OFS 2.5 2.5
\FONT=HELVET
\LETTERSIZ=0.2
\NOCOPYRIGHT
\INBOX
\LATEBOX
\PAUSE
\DATE
\PEN=3
\COLOR=1
HEADER :\CENTER\ TMAW: ${name}_LF T structure
X-ACHSE:\CENTER\ log &Rm&N / g cm&H-2&M
Y-ACHSE:\CENTER\ &RT&N / kK
      MASSTAB  MINIMUM  MAXIMUM  TEILUNGEN  BESCHRIFT.  DARUNTER
X: 25CM      -7.9      4.3      1.      1.      0.
Y: 16CM      0.      999.      100.      200.      0.
N= ? PLOTSYMBOL 5 PEN 3 XYTABLE SELECT 1 2 COLOR=2
COMMAND INCLUDE "${name}_LF.T-structure"
FINISH
\PEN=3
END
EOF
#

```



```

/home/rauch/wrplot.dir/proc.dir/wrps.com ${name}_LF_${DATE}_T-structure
#
#
#
#####
#
#
# # # #####
# ## ## ##### ## # # ## ## ##
# # # # # # # # # # # # # # #
# # # # # # # # # # # # # # #
# # # # # # # # # # # # # # #
# # # # # # # # # # # # # # #
# # # # # # # # # # # # # # #
#####
#
# create metadata and send it to GAVO archive @ ARI
#
cd ${RUNTMPDIR}
cp ~/jobs/TMAP_WEB/${name}_LF_${DATE}.WFP_????-???? ${WFPDIR}
cp ${WFPDIR}/${name}_LF_${DATE}.WFP_????-???? .
ls -ltr ${WFPDIR}/${name}_LF_${DATE}.WFP_????-????
#
# 5 - 2000 A (binned to 0.1 A)
#
mname=${name}_LF_${DATE}.WFP_00005-02000
#
ln -sf ${mname} in
${HOME}/bin/split.Linux_X86
rm in
if test ! -s wf.tab
then
echo "something wrong with ${name}_LF (split)!"
exit
fi
${HOME}/bin/fcloudy.Linux_X86
rm wf.tab
if test -s wf.bin_0.1
then
cat wf.header wf.bin_0.1 > ${mname}.bin_0.1
rm wf.header wf.bin_0.1
else
echo "something wrong with ${mname} (fcloudy) !"
rm wf.header
exit
fi
echo "${mname}" > tmawname
${HOME}/bin/devide.Linux_X86
rm tmawname
#
if [ ${DATABASE} = "YES" ]
then
cp ${mname}.dat ${mname}.meta ${VODIR}
chmod 644 ${VODIR}/${mname}.dat ${VODIR}/${mname}.meta
python ${HOME}/scripts/uploadspec.py -u -l rauch -p <password> ${mname}.meta
fi
#
# 2000 - 3000 A
#
mname=${name}_LF_${DATE}.WFP_02000-03000
ln -sf ${mname} ${mname}.bin_0.1
echo "${mname}" > tmawname
${HOME}/bin/devide.Linux_X86
rm tmawname
#
if [ ${DATABASE} = "YES" ]
then
cp ${mname}.dat ${mname}.meta ${VODIR}
chmod 644 ${VODIR}/${mname}.dat ${VODIR}/${mname}.meta
python ${HOME}/scripts/uploadspec.py -u -l rauch -p <password> ${mname}.meta

```

```

fi
#
# 3000 - 55000 A
#
mname=${name}_LF_${DATE}.WFP_03000-55000
ln -sf ${mname} ${mname}.bin_0.1
echo "${mname}" > tmawname
${HOME}/bin/devide.Linux_X86
rm tmawname
if [ ${DATABASE} = "YES" ]
then
cp ${mname}.dat ${mname}.meta ${VODIR}
chmod 644 ${VODIR}/${mname}.dat ${VODIR}/${mname}.meta
python ${HOME}/scripts/uploadspec.py -u -l rauch -p <password> ${mname}.meta
fi
#
tar -czf meta.tgz *.meta
if test -s meta.tgz
then
if test -s ~/models/TMAW/${name}_LF-WWW.converged
then
cat > MAIL << EOF
Dear ${NAME_first} ${NAME_last} ,
metadata files for standard flux tables according to your request parameters
were sent to the GAVO database.
The respective flux tables are available now within
http://dc.g-vo.org/theossa
If you have further questions, please do not hesitate to contact
astro-tmaw@listserv.uni-tuebingen.de.
Yours
TMAW Team
EOF
/usr/bin/mail -r "TMAW" -s "TMAW INFO: MetaData (${HOSTNAME})" -a \
meta.tgz astro-tmaw@listserv.uni-tuebingen.de < MAIL
rm MAIL
exit
fi
fi
fi
#
#
#####
#
# ##### ##
# # # # # # # # # # # # # # # #
# # # # # # # # # # # # # # # #
# # ##### ##### # # # # # # #
# # # # # # # # # # # # # # # #
# # # # # # # # # # # # # # # #
# # # # # # # # # # # # # # # #
#####
#
# saving flux tables, targzipping results and writing final mail
#
cd ${RESDIR}
cp ${name}_LF_${DATE}.WFP_????-???? ${WFPDIR} | chmod 644 \
${WFPDIR}/${name}_LF_${DATE}.WFP_????-????
#
if [ ${FLUXINTERVAL} == "UV" ]
then
cp ${name}_LF.flux ${name}_LF_${DATE}.flux
cp ${name}_LF.TMAW_out ${name}_LF_${DATE}.TMAW_out
tar czf ${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}.tgz README \
${name}_LF_${DATE}.TMAW_out ${name}_LF_${DATE}.flux ${name}_LF_${DATE}.IDENT \
${name}_LF_${DATE}.WFP ${name}_LF_${DATE}.WFP_0005-02000 ${name}_LF_*.ps && \
mv ${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}.tgz ${RESDIR2} && rm ${name}_LF*
fi
if [ ${FLUXINTERVAL} == "NUV" ]
then

```

```

cp ${name}_LF.flux ${name}_LF_${DATE}.flux
cp ${name}_LF.TMAW_out ${name}_LF_${DATE}.TMAW_out

tar czf ${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}.tgz README \
${name}_LF_${DATE}.TMAW_out ${name}_LF_${DATE}.flux ${name}_LF_${DATE}.IDENT \
${name}_LF_${DATE}.WFP ${name}_LF_${DATE}.WFP_02000-03000 ${name}_LF_*.ps && \
mv ${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}.tgz ${RESDIR2} && rm ${name}_LF*
fi
if [ ${FLUXINTERVAL} == "OPTIR" ]
then
cp ${name}_LF.flux ${name}_LF_${DATE}.flux
cp ${name}_LF.TMAW_out ${name}_LF_${DATE}.TMAW_out

#
tar czf ${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}.tgz README \
${name}_LF_${DATE}.TMAW_out ${name}_LF_${DATE}.flux ${name}_LF_${DATE}.IDENT \
${name}_LF_${DATE}.WFP ${name}_LF_${DATE}.WFP_03000-55000 ${name}_LF_*.ps && \
mv ${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}.tgz ${RESDIR2} && rm ${name}_LF*
fi
#
cp ${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}.tgz ${RESDIR2}
cp ${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}.tgz \
~/public_html/TMAW/results/${DATE}/${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}_${DATE}.tgz
chmod 644 ~/public_html/TMAW/results/${DATE}/*tgz
cd ${RUNTMPDIR}
#
if [ ${Reqnr} -le 1 ]
then
cat > MAIL << EOF
Dear ${NAME_first} ${NAME_last},

your TMAW request (${DATE}) was successfully completed.
The result files can be retrieved via
wget http://astro.uni-tuebingen.de/~TMAW/TMAW/results/${DATE}/all_models.tar .

If you have further questions,
please do not hesitate to contact astro-tmaw@listserv.uni-tuebingen.de.

When publishing research making use of this service, please acknowledge:
"The TMAW service (http://astro.uni-tuebingen.de/~TMAW) used to calculate
theoretical spectra for this paper was constructed as part of the
activities of the German Astrophysical Virtual Observatory."

Yours
TMAW Team
EOF
#
cp ${RESDIR2}/${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}.tgz \
~/public_html/TMAW/results/${DATE}/${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}_${DATE}.tgz
cp ${REQFILE} ~/public_html/TMAW/results/${DATE}/
#
ls ~/public_html/TMAW/results/${DATE}/*${namepart}*_${USERINTERVAL}_${DATE}.tgz \
| cut -c21-32 > /home/TMAW/public_html/TMAW/results/${DATE}/results
#
sed -e 's\_\ \g' /home/TMAW/public_html/TMAW/results/${DATE}/results > \
/home/TMAW/public_html/TMAW/results/${DATE}/results2
#
cp /home/TMAW/public_html/TMAW/plots/Display_request_${DATE}.plot Display_result.plot
sed -e 's\END\ \g' Display_result.plot > Display_results.plot
echo ' N=? PLOTSYMBOL=2 PEN=3 XYTABLE SELECT 1 2 SYMBOLSIZE=-.3 COLOR=1' >> \
Display_results.plot
echo 'COMMAND XLOG' >> Display_results.plot
echo 'COMMAND INCLUDE results2' >> Display_results.plot
echo 'FINISH' >> Display_results.plot
echo 'END' >> Display_results.plot
#
cp /atmosarray/TMAW/models/Edli_H.dat .
cp /home/TMAW/public_html/TMAW/results/${DATE}/results2 .
/home/rauch/wrplot.dir/proc.dir/wrps.com Display_results 2>/dev/null
ps2pdf Display_results.ps Display_results.pdf
mv Display_results.pdf /home/TMAW/public_html/TMAW/results/${DATE}/

```

```

chmod 664 /home/TMAW/public_html/TMAW/results/${DATE}/Display_results.pdf
#
tar -cvPf all_models.tar /home/TMAW/public_html/TMAW/results/${DATE}/* tgz
mv all_models.tar /home/TMAW/public_html/TMAW/results/${DATE}/
chmod 644 /home/TMAW/public_html/TMAW/results/${DATE}/all_models.tar
#
/home/TMAW/public_html/TMAW/results/${DATE}/calculation_started
#
/usr/bin/mail -r "TMAW" -s "TMAW: Final Results" -a \
/home/TMAW/public_html/TMAW/results/${DATE}/Display_results.pdf ${addy} < MAIL
    case ${addy} in
        rauch@astro.uni-tuebingen.de) ;;
        ringat@astro.uni-tuebingen.de) /usr/bin/mail -r "TMAW" -s "TMAW INFO: \
Final Results (${HOSTNAME})" -a \
/home/TMAW/public_html/TMAW/results/${DATE}/Display_results.pdf \
astro-tmaw@listserv.uni-tuebingen.de < MAIL ;;
        *) /usr/bin/mail -r "TMAW" -s "TMAW INFO: \
Final Results (${HOSTNAME})" -a \
/home/TMAW/public_html/TMAW/results/${DATE}/Display_results.pdf \
astro-tmaw@listserv.uni-tuebingen.de < MAIL ;;
    esac
    rm MAIL
    mv ${REQFILE} ${FINDIR}
#
else
cp ${RESDIR2}/${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}.tgz \
~/public_html/TMAW/results/${DATE}/${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}_${DATE}.tgz
cp ${REQFILE} ~/public_html/TMAW/results/${DATE}/
    cat > MAIL <<EOF
Dear ${NAME_first} ${NAME_last},

the model ${name}_LF from your TMAW request ${DATE} was successfully calculated.
The result file can be retrieved via

    wget http://astro.uni-tuebingen.de/~TMAW/TMAW/results/${DATE}/\
${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}_${DATE}.tgz

You will be notified at the end of your request calculations to download all models at once.
If you have further questions,
please do not hesitate to contact astro-tmaw@listserv.uni-tuebingen.de.

When publishing research making use of this service, please acknowledge:
"The TMAW service (http://astro.uni-tuebingen.de/~TMAW) used to calculate
theoretical spectra for this paper was constructed as part of the
activities of the German Astrophysical Virtual Observatory."

Yours
TMAW Team
EOF
#
/usr/bin/mail -r "TMAW" -s "TMAW: Job finished, results" ${addy} < MAIL
    case ${addy} in
        rauch@astro.uni-tuebingen.de) ;;
        ringat@astro.uni-tuebingen.de) /usr/bin/mail -r "TMAW" -s "TMAW INFO: \
Results (${HOSTNAME})" astro-tmaw@listserv.uni-tuebingen.de < MAIL ;;
        *) /usr/bin/mail -r "TMAW" -s "TMAW INFO: \
Results (${HOSTNAME})" astro-tmaw@listserv.uni-tuebingen.de < MAIL ;;
    esac
    rm MAIL
    mv ${REQFILE} ${FINDIR}
fi
#
# do not edit the rest of this file
#####
#set +x; ${HOME}/.jobend ${RUNTMPDIR}

```

Listing A.12: Die Prozedur TMAW\_iaat\_prepare.sh zur Erzeugung der für die Modellatmosphärenrechnung benötigten Atomdaten, Frequenzgitter und Programmdateien und existierte bereits vor dieser Arbeit. Stand: November 2012.

```
#!/bin/sh
set +x;. ${HOME}/.jobstart
# do not edit the beginning of this file
#####
#
# Procedure to create model atoms, frequency grid file and
# executables from the corresponding parameter values
#
#
# LAST_MODIFICATION="20-Sep-2012 09:07:30"
#
# Copyright 2007-2012 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
# CHANGELOG:
# 10.01.2007 first version of the script, ATOMS2 call
# 11.01.2007 SETF2 call
# 12.01.2007 parameter extraction
# 15.01.2007 added second SETF2 call for NLTE FGRID
# 21.02.2007 general revision
# 05.03.2007 normalize abundances
# formatted parameters coded in model name
# 28.06.2007 Teff-dependent model ions for carbon
# 24.10.2007 individual binaries (machine + request date)
# 13.12.2007 C LF model ions
# 11.01.2008 new paths for model-atom files
# 12.01.2008 Teff-dependent model ions for oxygen
# 28.01.2008 Teff-dependent model ions for nitrogen
# 19.02.2008 O LF model ions
# 23.02.2008 N LF model ions
# 29.02.2008 default PARAMETER for LINE1_PROF increased
# 13.03.2008 default PARAMETER for LINE1_PROF increased
# 24.08.2009 default PARAMETER for LINE1_PROF increased
# 05.01.2010 default PARAMETER for LINE1_PROF increased
# 25.03.2010 default PARAMETER for LINE1_PROF increased
# 04.10.2010 IABUNDXX variable introduced
# 07.10.2010 IABUNDXX bug eliminated
# 25.11.2010 EUV limit improved for 50 kK < Teff < 81 kK
# 07.06.2011 line-formation calculation introduced
# 04.07.2011 default PARAMETER for LINE1_PROF increased
# 08.08.2011 wait 240 sec for existing TMAW_preparing
# 09.08.2011 TMAW_preparing is checked twice before it is
# removed -> additional rm TMAW_preparing inserted
# 17.08.2011 LF files for ATOMS, FGRID, PARAMETER
# 22.08.2011 only one MASTER parameter file is used-> no compiling
# 29.11.2011 the MASTER parameter file is checked and eventually
# created again by checkPARA.sh
# 07.12.2011 TMAW_preparing, generated by checkreq.sh is deleted
# after the compilation part
# 09.01.2012 NRBFMAW increased to 39999999
# wrange for OUTPUT from ATOMS2 was corrected
# 13.01.2012 NRBFMAW increased to 99999999
# 19.03.2012 rm /home/TMAW/bimod/TMAW_preparing*
# NRBBMAD smaller for LINE1_PROF
# 21.03.2012 directory structure in adaten/rawdata changed
# 26.03.2012 AUTO ION HE for LF increased to 20
# 24.04.2012 NRBBMAX is not a default but taken individually
# 16.05.2012 AUTO ION HE changed from 14 to 20
```

```

#          25.05.2012 additional comments inserted
#          01.06.2012 NRBBMAD increased
#          31.08.2012 NRBBMAD decreased
#          20.09.2012 ION_2000-3000, LINIEN_2000-3000
#                    (LINE1-ERR-INPUT, no line requested)
#####
# IMPORTANT:
#
# TMAW_prepare
# needs the following parameters to run
#
# First command line parameter: T_eff
# Second                       : H abundance
# Third                         : He abundance
# Fourth                        : C abundance
# Fifth                         : N abundance
# Sixth                         : O abundance
# Seventh                       : lower limit of wavelength interval
# Eighth                        : upper limit of wavelength interval
# Ninth                         : user resolution
# Tenth                         : request date
#####
# TMAW_prepare job following ##
#####
#
# own definitions
#
ADATADIR=${HOME}/adaten/rawdata
H_mod=H/MODELING
H_syn=H/SYNSPEC
HEmod=HE/MODELING
HEsyn=HE/SYNSPEC
C_mod=C/MODELING
C_syn=C/SYNSPEC
N_mod=N/MODELING
N_syn=N/SYNSPEC
O_mod=O/MODELING
O_syn=O/SYNSPEC
#
BI=/home/rauch/bimod
load=/home/rauch/jobs/load/load
#
runsys=${sys}
#
# check if all parameters are present
#
if [ $# -lt 9 ]
then
    echo "TMAW_prepare: wrong number of arguments"
    exit 3
fi
TEFF=${1}
ABUNDH=${2}
ABUNDHE=${3}
ABUNDC=${4}
ABUNDN=${5}
ABUNDO=${6}
WAVBEG=${7}
WAVEND=${8}
WAVSTP=${9}
DATE=${10}
#
# combination of input model atoms for ATOMS2 according to requested elements
#
# convert abundances into normalized number fractions (for LTE2)
#
echo "5 1 ${ABUNDH} ${ABUNDHE} ${ABUNDC} ${ABUNDN} ${ABUNDO}" > normH
echo "5 2 ${ABUNDH} ${ABUNDHE} ${ABUNDC} ${ABUNDN} ${ABUNDO}" > normHE
echo "5 3 ${ABUNDH} ${ABUNDHE} ${ABUNDC} ${ABUNDN} ${ABUNDO}" > normC
echo "5 4 ${ABUNDH} ${ABUNDHE} ${ABUNDC} ${ABUNDN} ${ABUNDO}" > normN

```

```

echo "5 5 ${ABUNDH} ${ABUNDHE} ${ABUNDC} ${ABUNDN} ${ABUNDO}" > normO
#
#
Hfnum='/home/rauch/bimod/TMAW_mass2num${sys} < normH      2>/dev/null 1>&1'
HEfnum='/home/rauch/bimod/TMAW_mass2num${sys} < normHE     2>/dev/null 1>&1'
Cfnum='/home/rauch/bimod/TMAW_mass2num${sys} < normC       2>/dev/null 1>&1'
Nfnum='/home/rauch/bimod/TMAW_mass2num${sys} < normN       2>/dev/null 1>&1'
Ofnum='/home/rauch/bimod/TMAW_mass2num${sys} < normO       2>/dev/null 1>&1'
#
echo ${Hfnum} > IABUNDHin
echo ${HEfnum} > IABUNDHEin
echo ${Cfnum} > IABUNDCin
echo ${Nfnum} > IABUNDNin
echo ${Ofnum} > IABUNDOin
#
IABUNDH='/home/rauch/bimod/TMAW_gt${sys} < IABUNDHin 2>/dev/null 1>&1'
IABUNDHE='/home/rauch/bimod/TMAW_gt${sys} < IABUNDHEin 2>/dev/null 1>&1'
IABUNDC='/home/rauch/bimod/TMAW_gt${sys} < IABUNDCin 2>/dev/null 1>&1'
IABUNDN='/home/rauch/bimod/TMAW_gt${sys} < IABUNDNin 2>/dev/null 1>&1'
IABUNDO='/home/rauch/bimod/TMAW_gt${sys} < IABUNDOin 2>/dev/null 1>&1'
#
#
# using H as base for model atoms
cat ${ADATADIR}/${H_mod}/H > source
cat ${ADATADIR}/${H_mod}/H > source_LF
echo "ABUNDANCE H ${Hfnum}" > ABUNDIN
#
# model atoms used depend on T_eff
#
# selection of model atoms for atmosphere calculation
#
if [ ${TEFF} -lt 20000 ]
then
HE_model=HE_cool
C_model=C_I-IV
N_model=N_I-IV
O_model=O_I-IV
else
if [ ${TEFF} -le 30000 ]
then
HE_model=HE_cool
C_model=C_I-V
N_model=N_I-V
O_model=O_I-V
else
if [ ${TEFF} -le 40000 ]
then
HE_model=HE_cool
C_model=C_I-V
N_model=N_I-V
O_model=O_II-V
else
if [ ${TEFF} -le 70000 ]
then
HE_model=HE_cool
C_model=C_II-V
N_model=N_II-VI
O_model=O_II-VI
else
if [ ${TEFF} -le 100000 ]
then
HE_model=HE_cool
C_model=C_III-V
N_model=N_III-VI
O_model=O_III-VII
else
if [ ${TEFF} -le 140000 ]
then
HE_model=HE_hot
C_model=C_III-V
N_model=N_III-VI

```

```

O_model=O_III-VII
else
  if [ ${TEFF} -le 150000 ]
  then
    HE_model=HE_hot
    C_model=C_III-V
    N_model=N_IV-VII
    O_model=O_IV-VII
  else
    if [ ${TEFF} -le 180000 ]
    then
      HE_model=HE_hot
      C_model=C_IV-VI
      N_model=N_IV-VII
      O_model=O_IV-VII
    else
      if [ ${TEFF} -le 200000 ]
      then
        HE_model=HE_hot
        C_model=C_IV-VI
        N_model=N_IV-VII
        O_model=O_V-VII
      else
        if [ ${TEFF} -le 300000 ]
        then
          HE_model=HE_hot
          C_model=C_IV-VIII
          N_model=N_V-VIII
          O_model=O_V-VII
        else
          HE_model=HE_hot
          C_model=C_IV-VIII
          N_model=N_V-VIII
          O_model=O_V-VIII
        fi
      fi
    fi
  fi
fi
fi
fi
fi
fi
fi
fi
fi
#
# combination of model atoms
#
PURE_H=1
#
if [ ${IABUNDHE} -gt 0 ]
then
  cat ${ADATADIR}/${HEmod}/${HE_model} >> source
  cat ${ADATADIR}/${HEmod}/${HE_model}_LF >> source_LF
  echo "ABUNDANCE HE ${HEfnum}" >> ABUNDIN
  PURE_H=0
fi
if [ ${IABUNDC} -gt 0 -a ${C_model} != 'none' ]
then
  cat ${ADATADIR}/${C_mod}/${C_model} >> source
  cat ${ADATADIR}/${C_mod}/${C_model}_LF >> source_LF
  echo "ABUNDANCE C ${Cfnum}" >> ABUNDIN
  PURE_H=0
fi
if [ ${IABUNDN} -gt 0 -a ${N_model} != 'none' ]
then
  cat ${ADATADIR}/${N_mod}/${N_model} >> source
  cat ${ADATADIR}/${N_mod}/${N_model}_LF >> source_LF
  echo "ABUNDANCE N ${Nfnum}" >> ABUNDIN
  PURE_H=0
fi
if [ ${IABUNDO} -gt 0 -a ${O_model} != 'none' ]

```



```

then
  cat ${ADATADIR}/${O_mod}/${O_model} >> source
  cat ${ADATADIR}/${O_mod}/${O_model}_LF >> source_LF
  echo "ABUNDANCE O  ${Ofnum}" >> ABUNDIN
  PURE_H=0
fi
#
if [ ${PURE_H} -eq 1 ]
then
  echo "${PURE_H}" > ${RUNTMPDIR}/PURE_H
fi
#
cp ABUNDIN ${RUNTMPDIR}
#
# creation of atomic data files with ATOMS2
# extract parameters for PRO2 and LINE1_PROF
#
#
# LTE2 model atom
#
if [ ${IABUNDHE} -gt 0 ]
then
  HEX=''
else
  HEX='.'
fi
#
cat > opti << EOF
AUTO ION H   I   10  16 -10
${HEX}AUTO ION HE  II   14  32 -14
CBB-AUTO-FILL
CBF-AUTO-FILL
CBX-AUTO-FILL
RBB-AUTO-FILL ( NONE)
RBF-AUTO-FILL ( NOOP)
RDI-AUTO-FILL ( NONE)
END OPTIONS
EOF
cat opti source > SOURCE
${BI}/atoms2${sys} > OUTPUT_ATOMS2_LTE
cp ATOMS ${RUNTMPDIR}/ATOMS_LTE
echo "new LTE model atom created"
ls -l ${RUNTMPDIR}/ATOMS_LTE
#
rm opti ATOMS
#
#
# PRO2 model atom
#
cat > opti << EOF
AUTO ION H   I   10  16 -10
${HEX}AUTO ION HE  II   14  32 -14
CBB-AUTO-FILL
CBF-AUTO-FILL
CBX-AUTO-FILL
RBF-AUTO-FILL
RDI-AUTO-FILL ( NONE)
END OPTIONS
EOF
cat opti source > SOURCE
#
${BI}/atoms2${sys} > OUTPUT_ATOMS2_NLTE
cp ATOMS ${RUNTMPDIR}/ATOMS_NLTE
echo "new NLTE model atom created"
ls -l ${RUNTMPDIR}/ATOMS_NLTE
#
rm opti source ATOMS
#
#
# PRO2 line-formation model atom
#

```

```

cat > opti << EOF
AUTO ION H    I    15  16 -15
${HEX}AUTO ION HE II  20  32 -20
CBB-AUTO-FILL
CBF-AUTO-FILL
CBX-AUTO-FILL
RBF-AUTO-FILL
RDI-AUTO-FILL ( NONE)
END OPTIONS
EOF
cat opti source_LF > SOURCE
#
${BI}/atoms2${sys} > OUTPUT_ATOMS2_NLTE_LF
cp ATOMS ${RUNTMPDIR}/ATOMS_NLTE_LF
echo "new NLTE line-formation model atom created"
ls -l ${RUNTMPDIR}/ATOMS_NLTE_LF
#
rm opti source_LF ATOMS
#
#
# LINE1_PROF model atoms
#
# using H_synspec as base for model atoms
cat ${ADATADIR}/${H_syn}/H_synspec > source_LF
#
# model atoms used depend on T_eff
#
# selection of model atoms for synthetic spectrum
#
if [ ${TEFF} -lt 20000 ]
then
  HE_model=HE_cool_synspec
  C_model=C_I-IV_synspec
  N_model=N_I-IV_synspec
  O_model=O_I-IV_synspec
else
  if [ ${TEFF} -le 30000 ]
  then
    HE_model=HE_cool_synspec
    C_model=C_I-V_synspec
    N_model=N_I-V_synspec
    O_model=O_I-V_synspec
  else
    if [ ${TEFF} -le 40000 ]
    then
      HE_model=HE_cool_synspec
      C_model=C_I-V_synspec
      N_model=N_I-V_synspec
      O_model=O_II-V_synspec
    else
      if [ ${TEFF} -le 70000 ]
      then
        HE_model=HE_cool_synspec
        C_model=C_II-V_synspec
        N_model=N_II-VI_synspec
        O_model=O_II-VI_synspec
      else
        if [ ${TEFF} -le 100000 ]
        then
          HE_model=HE_cool_synspec
          C_model=C_III-V_synspec
          N_model=N_III-VI_synspec
          O_model=O_III-VII_synspec
        else
          if [ ${TEFF} -le 140000 ]
          then
            HE_model=HE_hot_synspec
            C_model=C_III-V_synspec
            N_model=N_III-VI_synspec
            O_model=O_III-VII_synspec
          else

```

```

if [ ${TEFF} -le 150000 ]
then
HE_model=HE_hot_synspec
C_model=C_III-V_synspec
N_model=N_IV-VII_synspec
O_model=O_IV-VII_synspec
else
if [ ${TEFF} -le 180000 ]
then
HE_model=HE_hot_synspec
C_model=C_IV-VI_synspec
N_model=N_IV-VII_synspec
O_model=O_IV-VII_synspec
else
if [ ${TEFF} -le 200000 ]
then
HE_model=HE_hot_synspec
C_model=C_IV-VI_synspec
N_model=N_IV-VII_synspec
O_model=O_V-VII_synspec
else
if [ ${TEFF} -le 300000 ]
then
HE_model=HE_hot_synspec
C_model=C_IV-VII_synspec
N_model=N_V-VIII_synspec
O_model=O_V-VII_synspec
else
HE_model=HE_hot_synspec
C_model=C_IV-VII_synspec
N_model=N_V-VIII_synspec
O_model=O_V-VIII_synspec
fi
fi
fi
fi
fi
fi
fi
fi
#
# combination of model atoms for line-formation
#
if [ ${IABUNDHE} -gt 0 ]
then
cat ${ADATADIR}/${HEsyn}/${HE_model} >> source_LF
fi
if [ ${IABUNDC} -gt 0 -a ${C_model} != 'none' ]
then
cat ${ADATADIR}/${C_syn}/${C_model} >> source_LF
fi
if [ ${IABUNDN} -gt 0 -a ${N_model} != 'none' ]
then
cat ${ADATADIR}/${N_syn}/${N_model} >> source_LF
fi
if [ ${IABUNDO} -gt 0 -a ${O_model} != 'none' ]
then
cat ${ADATADIR}/${O_syn}/${O_model} >> source_LF
fi
#
#####
# user's choice
#
Wa=${WAVBEG}
We=${WAVEND}
#
if [ ${Wa} -lt 10000 ]
then
if [ ${Wa} -lt 1000 ]

```

```

then
  if [ ${Wa} -lt 100 ]
  then
    if [ ${Wa} -lt 10 ]
    then
      Wanf="0000${Wa}"
    else
      Wanf="000${Wa}"
    fi
  else
    Wanf="00${Wa}"
  fi
else
  Wanf="0${Wa}"
fi
else
  Wanf="${Wa}"
fi
#
if [ ${We} -lt 10000 ]
then
  if [ ${We} -lt 1000 ]
  then
    if [ ${We} -lt 100 ]
    then
      if [ ${We} -lt 10 ]
      then
        Wend="0000${We}"
      else
        Wend="000${We}"
      fi
    else
      Wend="00${We}"
    fi
  else
    Wend="0${We}"
  fi
else
  Wend="${We}"
fi
#
#
cat > opti << EOF
AUTO ION H    I    15  16 -15
${HEX}AUTO ION HE  II  20  32 -20
LINEFORMATION - RBB-INTERVALL=[ ${Wanf} , ${Wend} ] , ION= NON
RBF-AUTO-FILL
RDI-AUTO-FILL ( NONE)
END OPTIONS
EOF
cat opti source_LF > SOURCE
${BI}/atoms2${sys} > OUTPUT_ATOMS2_WFP_LF
cp ATOMS ${RUNTMPDIR}/ATOMS_WFP_LF
echo "new WFP model atom created"
ls -l ${RUNTMPDIR}/ATOMS_WFP_LF
#
echo "${TEFF}"      > SDATEN
echo "PRINT ASCII" >> SDATEN
echo "1.30E+16"    >> SDATEN
echo "-1"          >> SDATEN
#
${BI}/setf2${sys} < SDATEN > OUTPUT_SETF2_UsersChoice_LF
cp ION_USER      ${RUNTMPDIR}/ION_USER
cp LINIEN_USER   ${RUNTMPDIR}/LINIEN_USER
#
mv F_BASE F_BASE_UsersChoice
rm opti source_LF ATOMS
#
# using H_synspec as base for model atoms
cat ${ADATADIR}/${H_syn}/H_synspec > source_LF
#

```

```

#
# model atoms used depend on T_eff
#
if [ ${IABUNDHE} -gt 0 ]
then
  cat ${ADATADIR}/${HEsyn}/${HE_model} >> source_LF
fi
if [ ${IABUNDC} -gt 0 -a ${C_model} != 'none' ]
then
  cat ${ADATADIR}/${C_syn}/${C_model} >> source_LF
fi
if [ ${IABUNDN} -gt 0 -a ${N_model} != 'none' ]
then
  cat ${ADATADIR}/${N_syn}/${N_model} >> source_LF
fi
if [ ${IABUNDO} -gt 0 -a ${O_model} != 'none' ]
then
  cat ${ADATADIR}/${O_syn}/${O_model} >> source_LF
fi
#
#####
# standard ionizing flux
#
Wa=5
We=2000
#
if [ ${Wa} -lt 10000 ]
then
  if [ ${Wa} -lt 1000 ]
  then
    if [ ${Wa} -lt 100 ]
    then
      if [ ${Wa} -lt 10 ]
      then
        Wanf="0000${Wa}"
      else
        Wanf="000${Wa}"
      fi
    else
      Wanf="00${Wa}"
    fi
  else
    Wanf="0${Wa}"
  fi
else
  Wanf="${Wa}"
fi
#
if [ ${We} -lt 10000 ]
then
  if [ ${We} -lt 1000 ]
  then
    if [ ${We} -lt 100 ]
    then
      if [ ${We} -lt 10 ]
      then
        Wend="0000${We}"
      else
        Wend="000${We}"
      fi
    else
      Wend="00${We}"
    fi
  else
    Wend="0${We}"
  fi
else
  Wend="${We}"
fi
#
#

```

```

#
cat > opti << EOF
AUTO ION H 1 15 16 -15
${HEX}AUTO ION HE 11 20 32 -20
LINEFORMATION - RBB-INTERVALL=[ ${Wanf}, ${Wend}], ION= NON
RBF-AUTO-FILL
RDI-AUTO-FILL ( NONE)
END OPTIONS
EOF
cat opti source_LF > SOURCE
${BI}/atoms2${sys} > OUTPUT_ATOMS2_WFP_LF_${Wanf}-${Wend}
cp ATOMS ${RUNTMPDIR}/ATOMS_WFP_LF_${Wanf}-${Wend}
echo "new WFP model atom created"
ls -l ${RUNTMPDIR}/ATOMS_WFP_LF_${Wanf}-${Wend}
#
rm opti source_LF ATOMS
#
#
# using H_synspec as base for model atoms
cat ${ADATADIR}/${H_syn}/H_synspec > source_LF
#
#
# model atoms used depend on T_eff
#
if [ ${IABUNDHE} -gt 0 ]
then
cat ${ADATADIR}/${HEsyn}/${HE_model} >> source_LF
fi
if [ ${IABUNDC} -gt 0 -a ${C_model} != 'none' ]
then
cat ${ADATADIR}/${C_syn}/${C_model} >> source_LF
fi
if [ ${IABUNDN} -gt 0 -a ${N_model} != 'none' ]
then
cat ${ADATADIR}/${N_syn}/${N_model} >> source_LF
fi
if [ ${IABUNDO} -gt 0 -a ${O_model} != 'none' ]
then
cat ${ADATADIR}/${O_syn}/${O_model} >> source_LF
fi
#
#####
# standard near ultraviolet
#
Wa=2000
if [ ${PURE_H} -eq 1 ]
then
We=4000
else
We=3000
fi
#
if [ ${Wa} -lt 10000 ]
then
if [ ${Wa} -lt 1000 ]
then
if [ ${Wa} -lt 100 ]
then
if [ ${Wa} -lt 10 ]
then
Wanf="0000${Wa}"
else
Wanf="000${Wa}"
fi
else
Wanf="00${Wa}"
fi
else
Wanf="0${Wa}"
fi
else

```

```

Wanf="${Wa} "
fi
#
if [ ${We} -lt 10000 ]
then
  if [ ${We} -lt 1000 ]
  then
    if [ ${We} -lt 100 ]
    then
      if [ ${We} -lt 10 ]
      then
        Wend="0000${We} "
      else
        Wend="000${We} "
      fi
    else
      Wend="00${We} "
    fi
  else
    Wend="0${We} "
  fi
else
  Wend="${We} "
fi
#
#
#
cat > opti << EOF
AUTO ION H      I      15  16 -15
${HEX}AUTO ION HE  II     20  32 -20
LINEFORMATION - RBB-INTERVALL=[ ${Wanf} , ${Wend} ], ION= NON
RBF-AUTO-FILL
RDI-AUTO-FILL ( NONE)
END OPTIONS
EOF
cat opti source_LF > SOURCE
${Bl}/atoms2${sys} > OUTPUT_ATOMS2_WFP_LF_${Wanf}-${Wend}
cp ATOMS ${RUNTMPDIR}/ATOMS_WFP_LF_${Wanf}-${Wend}
echo "new WFP model atom created"
ls -l ${RUNTMPDIR}/ATOMS_WFP_LF_${Wanf}-${Wend}
#
echo "${TEFF}" > SDATEN
echo "PRINT ASCII" >> SDATEN
echo "1.30E+16" >> SDATEN
echo "-1" >> SDATEN
#
${Bl}/setf2${sys} < SDATEN > OUTPUT_SETF2_UsersChoice_LF
cp ION_USER ${RUNTMPDIR}/ION_${Wanf}-${Wend}
cp LINIEN_USER ${RUNTMPDIR}/LINIEN_${Wanf}-${Wend}
#
#
rm opti source_LF ATOMS
#
#
# using H_synspec as base for model atoms
cat ${ADATADIR}/${H_syn}/H_synspec > source_LF
#
# model atoms used depend on T_eff
#
if [ ${IABUNDHE} -gt 0 ]
then
  cat ${ADATADIR}/${HEsyn}/${HE_model} >> source_LF
fi
if [ ${IABUNDC} -gt 0 -a ${C_model} != 'none' ]
then
  cat ${ADATADIR}/${C_syn}/${C_model} >> source_LF
fi
if [ ${IABUNDN} -gt 0 -a ${N_model} != 'none' ]
then
  cat ${ADATADIR}/${N_syn}/${N_model} >> source_LF

```

```

fi
if [ ${IABUNDO} -gt 0 -a ${O_model} != 'none' ]
then
  cat ${ADATADIR}/${O_syn}/${O_model} >> source_LF
fi
#
#####
# standard optical - infrared
#
Wa=3000
We=55000
#
if [ ${Wa} -lt 10000 ]
then
  if [ ${Wa} -lt 1000 ]
  then
    if [ ${Wa} -lt 100 ]
    then
      if [ ${Wa} -lt 10 ]
      then
        Wanf="0000${Wa}"
      else
        Wanf="000${Wa}"
      fi
    else
      Wanf="00${Wa}"
    fi
  else
    Wanf="0${Wa}"
  fi
else
  Wanf="${Wa}"
fi
#
if [ ${We} -lt 10000 ]
then
  if [ ${We} -lt 1000 ]
  then
    if [ ${We} -lt 100 ]
    then
      if [ ${We} -lt 10 ]
      then
        Wend="0000${We}"
      else
        Wend="000${We}"
      fi
    else
      Wend="00${We}"
    fi
  else
    Wend="0${We}"
  fi
else
  Wend="${We}"
fi
#
#
#
cat > opti << EOF
AUTO ION H 1 15 16 -15
${HEX}AUTO ION HE 11 20 32 -20
LINEFORMATION - RBB-INTERVALL=[ ${Wanf}, ${Wend}], ION= NON
RBF-AUTO-FILL
RDI-AUTO-FILL ( NONE)
END OPTIONS
EOF
cat opti source_LF > SOURCE
${BI}/atoms2${sys} > OUTPUT_ATOMS2_WFP_LF_${Wanf}-${Wend}
cp ATOMS ${RUNTMPDIR}/ATOMS_WFP_LF_${Wanf}-${Wend}
echo "new WFP model atom created"
ls -l ${RUNTMPDIR}/ATOMS_WFP_LF_${Wanf}-${Wend}

```



```

#
rm opti source_LF ATOMS
#
# creation of frequency-grid files with SETF2
# extract parameters for PRO2 and LINE1_PROF
#
#####
# LTE2 frequency grid
#
cp ${RUNTMPDIR}/ATOMS_LTE ATOMS
#
echo "${TEFF}" > SDATEN
echo "PRINT CHECK" >> SDATEN
echo "1.30E+16" >> SDATEN
echo "-1" >> SDATEN
#
if [ ${TEFF} -lt 81000 ]
then
  if [ ${TEFF} -lt 50000 ]
  then
    if [ ${TEFF} -lt 30000 ]
    then
      echo "100 10" >> SDATEN
    else
      echo "100 10" >> SDATEN
    fi
  else
    echo " 30 10" >> SDATEN
  fi
else
  echo " 10 10" >> SDATEN
fi
#
${BI}/setf2${sys} < SDATEN > OUTPUT_SETF2_LTE
#
if test -f FGRID
then
  cp FGRID ${RUNTMPDIR}/FGRID_LTE
  chmod 600 ${RUNTMPDIR}/FGRID_LTE
  echo "new LTE frequency grid created"
  ls -l ${RUNTMPDIR}/FGRID_LTE
  rm section* ATOMS F_BASE FGRID SDATEN
else
  echo "no LTE frequency grid created"
  exit 3
fi
#
#####
# NLTE frequency grid
#
cp ${RUNTMPDIR}/ATOMS_NLTE ATOMS
#
if [ ${TEFF} -ge 50000 ]
then
  if [ ${TEFF} -ge 100000 ]
  then
    echo "1.100D+01 8.000D+02 1.000D+01" > section1
  else
    echo "1.000D+02 8.000D+02 1.000D+01" > section1
  fi
else
  if [ ${TEFF} -lt 30000 ]
  then
    echo "1.810D+02 8.000D+02 1.000D+01" > section1
  else
    echo "1.010D+02 8.000D+02 1.000D+01" > section1
  fi
fi
echo "8.0100D+02 9.0000D+02 1.0000D+00" > section2

```

```

echo "9.0010D+02 1.0000D+03 1.0000D-01" > section3
echo "1.0010D+03 1.3000D+03 5.0000D-01" > section4
echo "1.3020D+03 1.0000D+04 1.0000D+00" > section5
echo "1.0005D+04 5.5000D+04 5.0000D+00" > section6
#
${BI}/prep_conts_man${sys} > F_BASE < section1
${BI}/prep_conts_man${sys} >> F_BASE < section2
${BI}/prep_conts_man${sys} >> F_BASE < section3
${BI}/prep_conts_man${sys} >> F_BASE < section4
${BI}/prep_conts_man${sys} >> F_BASE < section5
${BI}/prep_conts_man${sys} >> F_BASE < section6
#
echo "${TEFF}" > SDATEN
echo "PRINT CHECK" >> SDATEN
echo "1.30E+16" >> SDATEN
echo "1" >> SDATEN
#
if [ ${TEFF} -lt 81000 ]
then
  if [ ${TEFF} -lt 50000 ]
  then
    if [ ${TEFF} -lt 30000 ]
    then
      echo "100 10" >> SDATEN
      else
      echo "100 10" >> SDATEN
      fi
    else
      echo " 30 10" >> SDATEN
    fi
  else
    echo " 10 50" >> SDATEN
  fi
#
${BI}/setf2${sys} < SDATEN > OUTPUT_SETF2_NLTE
#
if test -f FGRID
then
  cp FGRID ${RUNTMPDIR}/FGRID_NLTE
  chmod 600 ${RUNTMPDIR}/FGRID_NLTE
  echo "new NLTE frequency grid created"
  ls -l ${RUNTMPDIR}/FGRID_NLTE
  rm section* ATOMS F_BASE FGRID SDATEN
else
  echo "no NLTE frequency grid created"
  exit 3
fi
#
#####
# NLTE frequency grid for line-formation calculation
#
cp ${RUNTMPDIR}/ATOMS_NLTE_LF ATOMS
#
if [ ${TEFF} -ge 50000 ]
then
  if [ ${TEFF} -ge 100000 ]
  then
    echo "1.100D+01 8.000D+02 1.000D+01" > section1
    else
    echo "1.000D+02 8.000D+02 1.000D+01" > section1
    fi
  else
    if [ ${TEFF} -lt 30000 ]
    then
      echo "1.810D+02 8.000D+02 1.000D+01" > section1
      else
      echo "1.010D+02 8.000D+02 1.000D+01" > section1
      fi
    fi
  echo "8.0100D+02 9.0000D+02 1.0000D+00" > section2
  echo "9.0010D+02 1.0000D+03 1.0000D-01" > section3

```

```

echo "1.0010D+03 1.3000D+03 5.0000D-01" > section4
echo "1.3020D+03 1.0000D+04 1.0000D+00" > section5
echo "1.0005D+04 5.5000D+04 5.0000D+00" > section6
#
${Bl}/prep_conts_man${sys} > F_BASE < section1
${Bl}/prep_conts_man${sys} >> F_BASE < section2
${Bl}/prep_conts_man${sys} >> F_BASE < section3
${Bl}/prep_conts_man${sys} >> F_BASE < section4
${Bl}/prep_conts_man${sys} >> F_BASE < section5
${Bl}/prep_conts_man${sys} >> F_BASE < section6
#
echo "${TEFF}" > SDATEN
echo "PRINT CHECK" >> SDATEN
echo "1.30E+16" >> SDATEN
echo "1" >> SDATEN
#
if [ ${TEFF} -lt 81000 ]
then
  if [ ${TEFF} -lt 50000 ]
  then
    if [ ${TEFF} -lt 30000 ]
    then
      echo "100 10" >> SDATEN
    else
      echo "100 10" >> SDATEN
    fi
  else
    echo " 30 10" >> SDATEN
  fi
else
  echo " 10 50" >> SDATEN
fi
#
${Bl}/setf2${sys} < SDATEN > OUTPUT_SETF2_NLTE_LF
#
if test -f FGRID
then
  cp FGRID ${RUNTMPDIR}/FGRID_NLTE_LF
  chmod 600 ${RUNTMPDIR}/FGRID_NLTE_LF
  echo "new NLTE frequency grid created"
  ls -l ${RUNTMPDIR}/FGRID_NLTE_LF
  rm section* ATOMS F_BASE FGRID SDATEN
else
  echo "no NLTE frequency grid for line-formation created"
  exit 3
fi
#
# now check the outputs and check if the parameter file must be changed
# then re-compile
#
#####
## PARA_MASTER
#####
# Extracting from ATOMS2 output
NATOMMAX='grep NATOMAX OUTPUT_ATOMS2_NLTE | awk '{ print $4 }''
NIONMAX='grep NIONMAX OUTPUT_ATOMS2_NLTE | awk '{ print $4 }''
NLMAX='grep NLMAX OUTPUT_ATOMS2_NLTE | awk '{ print $4 }''
LTEMAX='grep LTEMAX OUTPUT_ATOMS2_NLTE | awk '{ print $4 }''
NRBBMAX='grep NRBBMAX OUTPUT_ATOMS2_NLTE | awk '{ print $4 }''
NRBFMAX='grep NRBFMAX OUTPUT_ATOMS2_NLTE | awk '{ print $4 }''
NRDIMAX='grep NRDIMAX OUTPUT_ATOMS2_NLTE | awk '{ print $4 }''
NRFFMAX='grep NRFFMAX OUTPUT_ATOMS2_NLTE | awk '{ print $4 }''
NRLLMAX='grep NRLLMAX OUTPUT_ATOMS2_NLTE | awk '{ print $4 }''
NRLUMAX='grep NRLUMAX OUTPUT_ATOMS2_NLTE | awk '{ print $4 }''
NCBBMAX='grep NCBBMAX OUTPUT_ATOMS2_NLTE | awk '{ print $4 }''
NCBFMAX='grep NCBFMAX OUTPUT_ATOMS2_NLTE | awk '{ print $4 }''
NCBXMAY='grep NCBXMAY OUTPUT_ATOMS2_NLTE | awk '{ print $4 }''
# Extracting from SETF2 output
NFMAX='grep NFMAX OUTPUT_SETF2_NLTE | awk '{ print $4 }''
NRBBMAX='grep NRBBMAX OUTPUT_SETF2_NLTE | awk '{ print $4 }''
NRBBMAD='grep NRBBMAD OUTPUT_SETF2_NLTE | awk '{ print $4 }''

```

```

NRBFMAW='grep NRBFMAW OUTPUT_SETF2_NLTE | awk '{ $foo=$9 ; sub(/\)/, "", $foo); \
print $foo }''
#
NRBLMAX=${NRBBMAX}
NCBBMAD=6
NCBFMAD=2
NRFFMAD=2
NCBXMAD=8
NRLFMAX=1
NRBFMAD=6
NRLLMAD=1
NRLUMAD=1
if [ ${NRLLMAX} -eq 0 ]
then
NRLLMAX=1
fi
if [ ${NRLUMAX} -eq 0 ]
then
NRLUMAX=1
fi
NL=1
#
~/scripts/checkPARAM.sh pro2 MASTER ${NLMAX} ${NIONMAX} ${NATOMAX} ${NFMAX} ${LTEMAX} \
${NLMAX} ${NRBBMAX} ${NRBFMAX} ${NCBBMAX} ${NCBFMAX} ${NRFFMAX} ${NCBXMAD} ${NRLLMAX} \
${NRLUMAX} ${NRBBMAD} ${NRBFMAD} ${NCBBMAD} ${NCBFMAD} ${NRFFMAD} ${NCBXMAD} ${NL} \
${NL} 2>~/requests/checkPARAM_values
#
#####
## PARAMETER_LF
#####
### Extracting from ATOMS2 output
NATOMAX='grep NATOMAX OUTPUT_ATOMS2_NLTE_LF | awk '{ print $4 }''
NIONMAX='grep NIONMAX OUTPUT_ATOMS2_NLTE_LF | awk '{ print $4 }''
NLMAX='grep NLMAX OUTPUT_ATOMS2_NLTE_LF | awk '{ print $4 }''
LTEMAX='grep LTEMAX OUTPUT_ATOMS2_NLTE_LF | awk '{ print $4 }''
NRBBMAX='grep NRBBMAX OUTPUT_ATOMS2_NLTE_LF | awk '{ print $4 }''
NRBFMAX='grep NRBFMAX OUTPUT_ATOMS2_NLTE_LF | awk '{ print $4 }''
NRDIMAX='grep NRDIMAX OUTPUT_ATOMS2_NLTE_LF | awk '{ print $4 }''
NRFFMAX='grep NRFFMAX OUTPUT_ATOMS2_NLTE_LF | awk '{ print $4 }''
NRLLMAX='grep NRLLMAX OUTPUT_ATOMS2_NLTE_LF | awk '{ print $4 }''
NRLUMAX='grep NRLUMAX OUTPUT_ATOMS2_NLTE_LF | awk '{ print $4 }''
NCBBMAX='grep NCBBMAX OUTPUT_ATOMS2_NLTE_LF | awk '{ print $4 }''
NCBFMAX='grep NCBFMAX OUTPUT_ATOMS2_NLTE_LF | awk '{ print $4 }''
NCBXMAD='grep NCBXMAD OUTPUT_ATOMS2_NLTE_LF | awk '{ print $4 }''
# Extracting from SETF2 output
NFMAX='grep NFMAX OUTPUT_SETF2_NLTE_LF | awk '{ print $4 }''
NRBBMAX='grep NRBBMAX OUTPUT_SETF2_NLTE_LF | awk '{ print $4 }''
NRBBMAD='grep NRBBMAD OUTPUT_SETF2_NLTE_LF | awk '{ print $4 }''
NRBFMAW='grep NRBFMAW OUTPUT_SETF2_NLTE_LF | awk '{ $foo=$9 ; sub(/\)/, "", $foo); \
print $foo }''
#
NRBLMAX=${NRBBMAX}
NCBBMAD=6
NCBFMAD=2
NRFFMAD=2
NCBXMAD=8
NRLFMAX=1
NRBFMAD=6
NRLLMAD=1
NRLUMAD=1
if [ ${NRLLMAX} -eq 0 ]
then
NRLLMAX=1
fi
if [ ${NRLUMAX} -eq 0 ]
then
NRLUMAX=1
fi
NL=1
#

```

```

~/scripts/checkPARAM.sh pro2 MASTER_LF ${NLMAX} ${NIONMAX} ${NATOMAX} ${NFMAX} ${LTEMAX} \
${NLMAX} ${NRBBMAX} ${NRBFMAX} ${NCBBMAX} ${NCBFMAX} ${NRFFMAX} ${NCBMAX} ${NRLLMAX} \
${NRLUMAX} ${NRBBMAD} ${NRBFMAD} ${NCBBMAD} ${NCBFMAD} ${NRFFMAD} ${NCBXMAD} ${NL} ${NL} \
2>~/requests/checkPARAM_values
#
#####
## PARAMETER LINE1_PROF
#####
### 1) user's choice
#
wrange=""
#
# Extracting from ATOMS2 output
NATOMAX='grep NATOMAX OUTPUT_ATOMS2_WFP_LF | awk '{ print $4 }''
NIONMAX='grep NIONMAX OUTPUT_ATOMS2_WFP_LF | awk '{ print $4 }''
NLMAX='grep NLMAX OUTPUT_ATOMS2_WFP_LF | awk '{ print $4 }''
LTEMAX='grep LTEMAX OUTPUT_ATOMS2_WFP_LF | awk '{ print $4 }''
NRBBMAX='grep NRBBMAX OUTPUT_ATOMS2_WFP_LF | awk '{ print $4 }''
NRBFMAX='grep NRBFMAX OUTPUT_ATOMS2_WFP_LF | awk '{ print $4 }''
NRDIMAX='grep NRDIMAX OUTPUT_ATOMS2_WFP_LF | awk '{ print $4 }''
NRFFMAX='grep NRFFMAX OUTPUT_ATOMS2_WFP_LF | awk '{ print $4 }''
NRLLMAX='grep NRLLMAX OUTPUT_ATOMS2_WFP_LF | awk '{ print $4 }''
NRLUMAX='grep NRLUMAX OUTPUT_ATOMS2_WFP_LF | awk '{ print $4 }''
#
NRBLMAX=${NRBBMAX}
NCBBMAD=6
NCBFMAD=2
NRFFMAD=2
NCBXMAD=8
NRLFMAX=1
NRBFMAD=6
NRLLMAD=1
NRLUMAD=1
# default settings
NFMAX=100000
NRBBMAD=29999
NRBFMAX=99999999
NCBBMAX=1
NCBFMAX=1
NCBXMAD=1
#
if [ ${NRLLMAX} -eq 0 ]
then
NRLLMAX=1
fi
if [ ${NRLUMAX} -eq 0 ]
then
NRLUMAX=1
fi
NL=1
#
~/scripts/checkPARAM.sh line1_prof MASTER ${NLMAX} ${NIONMAX} ${NATOMAX} ${NFMAX} ${LTEMAX} \
${NL} ${NRBBMAX} ${NRBFMAX} ${NCBBMAX} ${NRBFMAX} ${NRFFMAX} ${NCBMAX} ${NRLLMAX} \
${NRLUMAX} ${NRBBMAD} ${NRBFMAD} ${NCBBMAD} ${NCBFMAD} ${NRFFMAD} ${NCBXMAD} ${NL} \
${NL} 2>~/requests/checkPARAM_values
##
# 2) standard UV
#
wrange='_00005-02000'
#wrange=''
#
# Extracting from ATOMS2 output
NATOMAX='grep NATOMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NIONMAX='grep NIONMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NLMAX='grep NLMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
LTEMAX='grep LTEMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRBBMAX='grep NRBBMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRBFMAX='grep NRBFMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRDIMAX='grep NRDIMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRFFMAX='grep NRFFMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRLLMAX='grep NRLLMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''

```

```

NRLUMAX='grep NRLUMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
#
NRBLMAX=${NRBBMAX}
NCBBMAD=6
NCBFMAD=2
NRFFMAD=2
NCBXMAD=8
NRLFMAX=1
NRBFMAD=6
NRLLMAD=1
NRLUMAD=1
# default settings
NFMAX=100000
NRBBMAD=29999
NRBFMAW=99999999
NCBBMAX=1
NCBFMAX=1
NCBXMAD=1
#
if [ ${NRLLMAX} -eq 0 ]
then
NRLLMAX=1
fi
if [ ${NRLUMAX} -eq 0 ]
then
NRLUMAX=1
fi
NL=1
#
~/scripts/checkPARAM.sh line1_prof MASTER ${NLMAX} ${NIONMAX} ${NATOMAX} ${NFMAX} ${LTEMAX} \
${NL} ${NRBBMAX} ${NRBFMAX} ${NCBBMAX} ${NCBFMAX} ${NRFFMAX} ${NCBXMAD} ${NRLLMAX} \
${NRLUMAX} ${NRBBMAD} ${NRBFMAD} ${NCBBMAD} ${NCBFMAD} ${NRFFMAD} ${NCBXMAD} ${NL} \
${NL} 2>~/requests/checkPARAM_values
#
## 3) standard near infrared
#
if [ ${PURE_H} -eq 1 ]
then
wrange='_02000-04000'
else
wrange='_02000-03000'
fi
#
# Extracting from ATOMS2 output
NATOMAX='grep NATOMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NIONMAX='grep NIONMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NLMAX='grep NLMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
LTEMAX='grep LTEMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRBBMAX='grep NRBBMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRBFMAX='grep NRBFMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRDIMAX='grep NRDIMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRFFMAX='grep NRFFMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRLLMAX='grep NRLLMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRLUMAX='grep NRLUMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
#
NRBLMAX=${NRBBMAX}
NCBBMAD=6
NCBFMAD=2
NRFFMAD=2
NCBXMAD=8
NRLFMAX=1
NRBFMAD=6
NRLLMAD=1
NRLUMAD=1
# default settings
NFMAX=100000
NRBBMAD=29999
NRBFMAW=99999999
NCBBMAX=1
NCBFMAX=1
NCBXMAD=1

```

```

#
if [ ${NRLLMAX} -eq 0 ]
then
  NRLLMAX=1
fi
if [ ${NRLUMAX} -eq 0 ]
then
  NRLUMAX=1
fi
#
NL=1
~/scripts/checkPARAM.sh line1_prof MASTER ${NLMAX} ${NIONMAX} ${NATOMAX} ${NFMAX} ${LTEMAX} \
${NL} ${NRBBMAX} ${NRBFMAX} ${NCBBMAX} ${NCBFMAX} ${NRFFMAX} ${NCBMAX} ${NRLLMAX} \
${NRLUMAX} ${NRBBMAD} ${NRBFMAD} ${NCBBMAD} ${NCBFMAD} ${NRFFMAD} ${NCBMAD} ${NL} \
${NL} 2>~/requests/checkPARAM_values
#
# 4) standard optical - infrared
#
wrange='_03000-55000'
#
# Extracting from ATOMS2 output
NATOMAX='grep NATOMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NIONMAX='grep NIONMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NLMAX='grep NLMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
LTEMAX='grep LTEMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRBBMAX='grep NRBBMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRBFMAX='grep NRBFMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRDIMAX='grep NRDIMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRFFMAX='grep NRFFMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRLLMAX='grep NRLLMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRLUMAX='grep NRLUMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
#
NRBLMAX=${NRBBMAX}
NCBBMAD=6
NCBFMAD=2
NRFFMAD=2
NCBMAD=8
NRLFMAX=1
NRBFMAD=6
NRLLMAD=1
NRLUMAD=1
# default settings
NFMAX=10000
NRBBMAD=29999
NRBFMAW=99999999
NCBBMAX=1
NCBFMAX=1
NCBMAX=1
#
if [ ${NRLLMAX} -eq 0 ]
then
  NRLLMAX=1
fi
if [ ${NRLUMAX} -eq 0 ]
then
  NRLUMAX=1
fi
NL=1
#
~/scripts/checkPARAM.sh line1_prof MASTER ${NLMAX} ${NIONMAX} ${NATOMAX} ${NFMAX} ${LTEMAX} \
${NL} ${NRBBMAX} ${NRBFMAX} ${NCBBMAX} ${NCBFMAX} ${NRFFMAX} ${NCBMAX} ${NRLLMAX} \
${NRLUMAX} ${NRBBMAD} ${NRBFMAD} ${NCBBMAD} ${NCBFMAD} ${NRFFMAD} ${NCBMAD} ${NL} \
${NL} 2>~/requests/checkPARAM_values
#
rm /home/TMAW/bimod/TMAW_preparing*
#
# do not edit the rest of this file
#####
set +x; ${HOME}/.jobend ${TMPDIR}

```

Listing A.13: Auszug aus der TMAW\_iaat\_jobfile.sh-Prozedur. Diese Prozedur ist für die eigentliche Berechnung der Atmosphäre zuständig und war ebenfalls vor dieser Arbeit bereits vorhanden. Stand: November 2012.

```
#!/bin/sh
set +x;. ${HOME}/.jobstart
# do not edit the beginning of this file
#####
#
# Procedure to calculate a model in 9 (continuum) + 7 (line) + 3 (post) steps
# Afterwards SEDs are produced for all wavelength ranges
#
#
# LAST_MODIFICATION="23-Okt-2012 09:20:15"
#
# Copyright 2006-2012 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#
#
#####
# CHANGELOG:
# 19.10.2006 first version of the script
# 19.12.2006 $name is now set globally in TMAW_iaat_run.sh
# 08.01.2007 added second NLTE job for lines, first NLTE
# run now without lines
# 15.01.2007 changed to TMAW_jobfile
# using ATOMS, FGRID and binaries prepared in
# TMAW_prepare.sh now.
# 21.02.2007 general revision
# 05.03.2007 4-step model calculation
# 10.10.2007 2000-3000A problem
# 24.10.2007 individual binaries (machine + request date)
# 02.11.2007 increased degree of convergence
# 04.11.2007 final_model
# 02.03.2008 occupation probability for all ions
# 03.03.2008 5-step continuum-model calculation
# 06.03.2008 7-step continuum-model calculation
# 07.03.2008 6-step line-model calculation
# 10.03.2008 7-step line-model calculation
# 22.05.2008 9-step continuum-model calculation
# 17.11.2008 DISTA for high g extended
# 21.11.2008 occupation probabilities for all ions
# 25.11.2008 user resolution
# 09.04.2009 refined convergence strategy
# 08.07.2009 DISTA for high g extended (preview)
# 11.08.2009 database transfer only in case of convergence
# 14.01.2010 increase ITMAX and steps for convergence
# 22.08.2011 only one MASTER parameter file is used for all jobs
# 14.11.2011 additional steps with the real temperature correction and
# line-formation are inserted
# 16.11.2011 Email notification if line1_prof did not work
# 28.11.2011 timestamp for WFP file is set in front of "WFP"
# to get similar format
# 14.12.2011 t-correction intervall changed to -3-10
# 12.01.2012 the DISTA and F_BASE parts were modified for
# 5-2000 and 3000-55000
# 06.03.2012 The DISTA and F_BASE modification was changed back
# for the 3000-55000 range
# The atomic data for LF was reduced
# 19.03.2012 T-CORRECTION PROFILE: 1 inserted for post-calculation step 8+9
# T-CORRECTION intervall changed to -5 - +10
# CVEC for post calc step 9 -0.5 instead of 0
```



```

#      22.03.2012 CVEC for step 8+9 of postcalculation increased
#      10.04.2012 Additional step 0 introduced to start with the line
#                  calculation with a step-up of f-values

#      04.05.2012 T-CORRECTION intervall changed to -5 - +4
#      10.05.2012 exit when LINE1_PROF had an error
#      11.05.2012 NLTE step 7 is not repeated 10 times any more
#                  but 3 times with
#                  higher ITMAX (50) and KANTOROVICH SWITCH LIMIT 1E33
#      14.05.2012 The name of the requestfile is used to copy the MASTER
#                  executables to this name -> jobcontrol finds out
#                  which request stopped
#      15.05.2012 FF GAUNT FACTORS: SUTHERLAND cart deleted
#                  DEPTH DEPENDENT LINE PROFILES replaced by
#                  DEPTH DEPENDENT LINE PROFILES, LINEARIZATION:FIRST
#      21.05.2012 additional length check of the WFP files is implemented
#                  DYNAMIC DISTA for 3000-55000 interval
#      25.05.2012 additional comments inserted
#                  DISTA adjusted
#      28.08.2012 DYNAMIC DISTA for 2000-3000 interval
#                  DISTA adjustment for 3000-55000
#                  commented parts were deleted
#      20.09.2012 ION_2000-3000, LINIEN_2000-3000
#                  (LINE1-ERR-INPUT, no line requested)
#      15.10.2012 step 7: ITMAX=100 instead of ITMAX=30
#      15.10.2012 step 0) REDUCE LOG CVEC -2.0, NO TEMPERATURE CORRECTION,
#                  STEP-UP F-VALUES ITMAX=100: instead of ITMAX=20
#      15.10.2012 step 1) REDUCE LOG CVEC -2.0, NO TEMPERATURE CORRECTION,
#                  ITMAX=40: instead of ITMAX=10
#      18.10.2012 step 0: REDUCE LOG CVEC -3.0 instead of STEP-UP F-VALUES
#      19.10.2012 step 0b: REDUCE LOG CVEC -2.5
#      23.10.2012 step 0: STEP-UP F-VALUES mit REDUCE LOG CVEC -3.0
#                  for Teff geq 30000
#####
#
# IMPORTANT:
# JOBFIL FOR TMAW JOBS
#
# Needs parameters to run
# First command line parameter: T_eff
# Second                       : log g
# Third                         : lower limit of wavelength interval
# Fourth                       : upper limit of wavelength interval
# Fifth                        : user resolution
# Sixth                        : request date
# Seventh                      : model name
#####
#
# check if all parameters are present
#
#sys=.Linux_x64
echo $1
echo $2
echo $3
echo $4
echo $5
echo $6
echo $7
echo $8

if [ $# -lt 8 ]
then
    echo "wrong number of arguments"
    exit 3
fi
#
# TMAW job following
#####
#
# own definitions

```

```

#
#
mod=TMAW
#
#
name=${7}
#
final_model=${HOME}/models/${mod}/${name}_WWW
export final_model
#
code=pro2
code2=line1_prof
#
DATE1=${6}
#
REGFILE=${8}
#
# copy the executable to one called like the request to identify it with jobcontrol.sh
cp ~/bimod/pro2_MASTER${sys} ${RUNTMPDIR}/pro2_${REGFILE}${sys}
cp ~/bimod/pro2_MASTER_LF${sys} ${RUNTMPDIR}/pro2_${REGFILE}_LF${sys}
cp ~/bimod/line1_prof_MASTER${sys} ${RUNTMPDIR}/line1_prof_${REGFILE}${sys}
modcode=${REGFILE}
modcode2=${REGFILE}_LF
#
DA=/home/rauch/data
JO=${HOME}/jobs
jobdir=${JO}/TMAP_WEB
export jobdir
#
Bl=${HOME}/bimod
#
#— model parameters
#
#— Teff
TT=${1}
#
#— log g
GG=${2}
#
#— wavelength interval for flux table
WAVBEG=${3}
WAVEND=${4}
WAVSTP=${5}
#
#
#
echo "start of sequence?"
#
if test -s ${final_model}.failed
then
  echo "a previous model calculation for ${name} failed, please wait for instructions"
  exit 3
fi
#
# start of: sequence, if model does not exist yet and is not currently calculating
#
#
if test ! -s ${final_model}.calculating \
  -a ! -s ${final_model}.failed
then
  echo "sequence started for ${name} at 'date +%H:%M:%S' 'date +%d-%m-%Y' at ${HOSTNAME}"
  echo "sequence started for ${name} at 'date +%H:%M:%S' 'date +%d-%m-%Y' at ${HOSTNAME}" > ${\
    final_model}.calculating
  touch ${final_model}.calculating@${HOSTNAME}
#
#
#
#
#####
#

```

```

# # ##### ##### #####
# # # # # #
# # # # #
# # # ##### #####
# # # # #
# # # # #
# ##### # ##### #####
#
#####
# start model
#
if test ! -s ${HOME}/models/lte
then
mkdir -p ${HOME}/models/lte
fi
MO=${HOME}/models/lte/${name}
#
#
# start of: calculate LTE2 model only if not already existing
if test ! -s ${MO}
then
#
echo "5000" > CPUTIME
#
if [ ${TT} -lt 30000 ]
then
ITMAX=1
else
ITMAX=100
fi
#
cat > LDATEN << eos
T EFF ${TT}
LOG G ${GG}
DAMP=0.1
ITMAX=${ITMAX}
EPS=1.0E-6
TAU SCALE 86 90 -2
.TAU SCALE MINIMUM -9
TAU SCALE MAXIMUM 4.0
PRINT INTEGRATED EDDINGTON FLUX,ITERATION:LAST
PRINT TEMPERATURE CORRECTIONS,ITERATION:LAST,ALL DEPTHS
.PRINT MODEL ATOMS (OVERVIEW)
eos
cat ${RUNTMPDIR}/ABUNDIN >> LDATEN
#
cp ${RUNTMPDIR}/ATOMS_LTE ATOMS
cp ${RUNTMPDIR}/FGRID_LTE FGRID
#
#####{BI}/lte2${sys} 2>/dev/null < LDATEN
touch ERRORS
${BI}/lte2${sys}< LDATEN
#
if test -s MODELL
then
cp MODELL ${MO}
chmod 600 ${MO}
echo "new LTE2 model ${MO} created"
ls -l ${MO}
else
cat "no new model ${MO} created" ERRORS > ${MO}.failed
cat "no new model ${MO} created" ERRORS > ${MO}.failed@${HOSTNAME}
echo "no LTE2 model ${MO} created"
cat "no new model ${MO} created" ERRORS > ${final_model}.failed
rm ${final_model}.calculating*
# clean up all binaries compiled for this request
#rm ${BI}/*${modcode}*${sys}
exit 3
fi
#
rm *

```

```

#
# end of: calculate LTE2 model only if not already existing
fi
#
#
#
#####
#
# ##### ##### ##### #####
# # # # # # #
# # # # # # #
# ##### ##### # # #####
# # # # # # #
# # # # # # #
# # # # # ##### #####
#
#####
# NLTE model
#
#
if test ! -s ${HOME}/models/${mod}
then
  mkdir -p ${HOME}/models/${mod}
fi
#
#
# NLTE continuum model
# in ten steps
#
# step 1) REDUCE LOG CVEC -2.0, NO TEMPERATURE CORRECTION, STATISTICAL EQUATIONS ONLY
# step 2) REDUCE LOG CVEC -1.0, NO TEMPERATURE CORRECTION, STATISTICAL EQUATIONS ONLY
# step 3) NO TEMPERATURE CORRECTION, STATISTICAL EQUATIONS ONLY
# step 4) REDUCE LOG CVEC -2.0, NO TEMPERATURE CORRECTION
# step 5) REDUCE LOG CVEC -1.0, NO TEMPERATURE CORRECTION
# step 6) NO TEMPERATURE CORRECTION
# step 7) REDUCE LOG CVEC -2.0, TEMPERATURE CORRECTION (UL)
# step 8) REDUCE LOG CVEC -1.0, TEMPERATURE CORRECTION (UL)
# step 9) TEMPERATURE CORRECTION (UL)
#
#
#
# LTE2 model becomes start model
#
MI=${MO}
MO=${HOME}/models/${mod}/${name}_WWW_statonly
#
if test -s ${MO}
then
  MI=${MO}
fi
#
#
#
# NLTE continuum model
#
# last NLTE continuum model becomes start model
#
# step 1) REDUCE LOG CVEC -2.0, STATISTICAL EQUATIONS ONLY
#
# start of: calculate NLTE continuum model only if not already existing
if test -s ${MI} \
-a ! -s ${MO} \
-a ! -s ${MO}.converged \
-a ! -s ${MO}.failed \
-a ! -s ${final_model}
then

```

```

#
cat > DATEN << eos
COMMENT: TMAW calculation of NLTE continuum model, step 1
.
OCCUPATION PROBABILITY FORMALISM FOR H1
OCCUPATION PROBABILITY FORMALISM FOR HE1
OCCUPATION PROBABILITY FORMALISM FOR HE2
OCCUPATION PROBABILITY FORMALISM FOR C1
OCCUPATION PROBABILITY FORMALISM FOR C2
OCCUPATION PROBABILITY FORMALISM FOR C3
OCCUPATION PROBABILITY FORMALISM FOR C4
OCCUPATION PROBABILITY FORMALISM FOR C5
OCCUPATION PROBABILITY FORMALISM FOR C6
OCCUPATION PROBABILITY FORMALISM FOR N1
OCCUPATION PROBABILITY FORMALISM FOR N2
OCCUPATION PROBABILITY FORMALISM FOR N3
OCCUPATION PROBABILITY FORMALISM FOR N4
OCCUPATION PROBABILITY FORMALISM FOR N5
OCCUPATION PROBABILITY FORMALISM FOR N6
OCCUPATION PROBABILITY FORMALISM FOR N7
OCCUPATION PROBABILITY FORMALISM FOR O1
OCCUPATION PROBABILITY FORMALISM FOR O2
OCCUPATION PROBABILITY FORMALISM FOR O3
OCCUPATION PROBABILITY FORMALISM FOR O4
OCCUPATION PROBABILITY FORMALISM FOR O5
OCCUPATION PROBABILITY FORMALISM FOR O6
OCCUPATION PROBABILITY FORMALISM FOR O7
OCCUPATION PROBABILITY FORMALISM FOR O8
.
OPACITY PROJECT RBF DATA: START AT EDGE
OPACITY PROJECT RBF DATA: MISSING HYDROGENIC
.
LAMBDA=3
.
ITMAX=30
NEWMAX=4
.
RADIATIVE EQUILIBRIUM: DIFFERENTIAL/INTEGRAL FORM
.
SWITCH OFF LINES
.
KANTOROVICH=2, SWITCH LIMITS 0-->1, 1-->2, 2-->1 : 0.1 0.01 0.5
REDUCE LOG CVEC -3.0
.
NO TEMPERATURE CORRECTION
.
SOLVE STATISTICAL EQUATIONS ONLY RE-SOLVE PARTICLE CONSERVATION
NO NEGATIVE POPULATION NUMBERS (LTE)
.
PRINT OPTIONS
PRINT MODEL ATOMS (OVERVIEW)
PRINT ABUNDANCES
PRINT MAX. REL. CORRECTIONS EVERY 1 ITERATIONS
PRINT INTEGRATED SURFACE FLUX, ITERATION: EACH
PRINT CP-TIME/ITERATION, EACH
PRINT OUTPUT MODEL, ITERATION: LAST, DEPTH INCREMENT: 1 (STRUCTURE ONLY)
.
MACHINE 'hostname'
eos
#
cp ${MI} MODIN
cp ${RUNTMPDIR}/ATOMS_NLTE ATOMS
cp ${RUNTMPDIR}/FGRID_NLTE FGRID
${DA}/get_OP
#
#
#
#
# ${BI}/${code}_${modcode}${sys} 2>ERRORS < DATEN
cp ${RUNTMPDIR}/${code}_${modcode}${sys} .
./${code}_${modcode}${sys} 2>ERRORS < DATEN

```

```

#
if test -s MODOUT
then
  cp MODOUT ${MO}
  chmod 600 ${MO}
  ls -l ${MO}
  rm MODOUT
  if test -s STOP
  then
    cp STOP ${MO}.converged
  fi
else
  cat "no new model ${MO} created" ERRORS > ${MO}.failed
  cat "no new model ${MO} created" ERRORS > ${MO}.failed@${HOSTNAME}
  echo "no new model ${MO} created"
  cat "no new model ${MO} created" ERRORS > ${final_model}.failed
  rm ${final_model}.calculating*
  # clean up all binaries compiled for this request
  #rm ${BI}/*${modcode}*${sys}
  exit 3
fi
#
rm *
#
# end of: calculate NLTE continuum model only if not already existing
fi
#
#
#
# last NLTE continuum model becomes start model
#
MI=${MO}
MO=${HOME}/models/${mod}/${name}_WWW_cont_0T-2
#
if test -s ${MO}
then
  MI=${MO}
fi
#
#
# NLTE continuum model
#
# step 2) REDUCE LOG CVEC -1.0, STATISTICAL EQUATIONS ONLY
#####
###
### DIE FOLGENDEN SCHRITTE WERDEN ANALOG DURCHGEFUEHRT,
### HIER ABER NICHT GELISTET
###
#####
#
# now check if the last iterations were performed completely
lastit='grep iteration OUTPUT_${DATE} | tail -n 1 | cut -c13-15'
result=$(echo $lastit | egrep ^[[:digit:]]+$)
if [ "$result" = "" ]
then
  lastit=0
fi
if [ $lastit -ne 15 ]
then
  if [ $lastit -ne 0 ]
  then
    cat > MAIL <<EOF
    The job stopped at iteration ${lastit}.
    Something went wrong.
    EOF
    /usr/bin/mail -r "TMAW" -s "TMAW job failed: The job stopped before iteration 15" astro-
    tmaw@listserv.uni-tuebingen.de < MAIL
  rm MAIL

```







```

#
#
if test ! -s ${MI}
then
  echo "no input model: ${MI}"
  exit 3
fi
#
#
GRP=/atmosarray/rauch/group
OPD=/atmosarray/rauch/data/atomic
#
# 1) user's choice
#
# base frequency grid
echo "${WAVBEG} ${WAVEND} ${WAVSTP}" > F_BASE
#
cp ${RUNTMPDIR}/ION_USER ION
cp ${RUNTMPDIR}/LINIEN_USER LINIEN
#
cat > DISTA << eos
2 0.01
4 0.02
6 0.03
8 0.05
9 0.10
10 0.20
11 0.30
12 0.50
13 1.00
14 2.00
15 5.00
16 300.00
17 0.01
eos
#
cat > DATEN << eos
OCCUPATION PROBABILITY FORMALISM FOR H1
OCCUPATION PROBABILITY FORMALISM FOR HE1
OCCUPATION PROBABILITY FORMALISM FOR HE2
OCCUPATION PROBABILITY FORMALISM FOR C1
OCCUPATION PROBABILITY FORMALISM FOR C2
OCCUPATION PROBABILITY FORMALISM FOR C3
OCCUPATION PROBABILITY FORMALISM FOR C4
OCCUPATION PROBABILITY FORMALISM FOR C5
OCCUPATION PROBABILITY FORMALISM FOR C6
OCCUPATION PROBABILITY FORMALISM FOR N1
OCCUPATION PROBABILITY FORMALISM FOR N2
OCCUPATION PROBABILITY FORMALISM FOR N3
OCCUPATION PROBABILITY FORMALISM FOR N4
OCCUPATION PROBABILITY FORMALISM FOR N5
OCCUPATION PROBABILITY FORMALISM FOR N6
OCCUPATION PROBABILITY FORMALISM FOR N7
OCCUPATION PROBABILITY FORMALISM FOR O1
OCCUPATION PROBABILITY FORMALISM FOR O2
OCCUPATION PROBABILITY FORMALISM FOR O3
OCCUPATION PROBABILITY FORMALISM FOR O4
OCCUPATION PROBABILITY FORMALISM FOR O5
OCCUPATION PROBABILITY FORMALISM FOR O6
OCCUPATION PROBABILITY FORMALISM FOR O7
OCCUPATION PROBABILITY FORMALISM FOR O8
.
OPACITY PROJECT RBF DATA: START AT EDGE
OPACITY PROJECT RBF DATA: MISSING HYDROGENIC
.
PRINT ATOMMODELLE(KURZFORM)
PRINT LINIEN UND BLENDS
PLOT EINZELNE LINIEN PLOTTER
PLOT EMERGENTER FLUSS FLAM / LAM, INTERVAL: ${WAVBEG} ${WAVEND}
eos
#

```

```

cp ${MI} MODIN
cp ${RUNTMPDIR}/ATOMS_WFP_LF ATOMS
cp ${GRP}/H1 STARK_H1
cp ${GRP}/HE1 STARK_HE1
cp ${GRP}/HE2 STARK_HE2
chmod 600 STARK_*
${DA}/get_OP > /dev/null
#
cp ${RUNTMPDIR}/${code2}_${modcode}${sys} .
./${code2}_${modcode}${sys} < DATEN
#
#
if test -s IDENT
then
cp IDENT ${jobdir}/${name}_LF_${DATE1}.IDENT
fi
if test -s PLWFP
then
cp PLWFP ${jobdir}/${name}_LF_${DATE1}.WFP
fi
#
rm *
#
#
# 2) ionizing flux
#
# base frequency grid
cat > F_BASE << eos
5.0000000D+00 2.001000D+03 1.0000000D-01
eos
#
echo "H 1" > ION
#
cat > LINIEN << eos
BLENDRANGE      1205
H11      H12      1215 A
eos
#
cat > DISTA <<EOF
2 0.01
4 0.02
6 0.03
8 0.05
9 20.00
10 0.01
EOF
#
#
cat > DATEN <<EOF
OCCUPATION PROBABILITY FORMALISM FOR H1
OCCUPATION PROBABILITY FORMALISM FOR HE1
OCCUPATION PROBABILITY FORMALISM FOR HE2
OCCUPATION PROBABILITY FORMALISM FOR C1
OCCUPATION PROBABILITY FORMALISM FOR C2
OCCUPATION PROBABILITY FORMALISM FOR C3
OCCUPATION PROBABILITY FORMALISM FOR C4
OCCUPATION PROBABILITY FORMALISM FOR C5
OCCUPATION PROBABILITY FORMALISM FOR C6
OCCUPATION PROBABILITY FORMALISM FOR N1
OCCUPATION PROBABILITY FORMALISM FOR N2
OCCUPATION PROBABILITY FORMALISM FOR N3
OCCUPATION PROBABILITY FORMALISM FOR N4
OCCUPATION PROBABILITY FORMALISM FOR N5
OCCUPATION PROBABILITY FORMALISM FOR N6
OCCUPATION PROBABILITY FORMALISM FOR N7
OCCUPATION PROBABILITY FORMALISM FOR O1
OCCUPATION PROBABILITY FORMALISM FOR O2
OCCUPATION PROBABILITY FORMALISM FOR O3
OCCUPATION PROBABILITY FORMALISM FOR O4
OCCUPATION PROBABILITY FORMALISM FOR O5
OCCUPATION PROBABILITY FORMALISM FOR O6

```

OCCUPATION PROBABILITY FORMALISM FOR O7  
OCCUPATION PROBABILITY FORMALISM FOR O8

OPACITY PROJECT RBF DATA: START AT EDGE  
OPACITY PROJECT RBF DATA: MISSING HYDROGENIC

DYNAMIC DISTA 5 2000 10  
PRINT ATOMMODELLE(KURZFORM)  
PRINT LINIEN UND BLENDS  
PLOT EINZELNE LINIEN PLOTTER  
PLOT EMERGENTER FLUSS FLAM / LAM, INTERVAL: 5.0 2000.1  
EOF

```
#
cat DATEN
#
cp ${MI} MODIN
cp ${RUNTMPDIR}/ATOMS_WFP_LF_00005-02000 ATOMS
cp ${GRP}/H1 STARK_H1
cp ${GRP}/HE1 STARK_HE1
cp ${GRP}/HE2 STARK_HE2
chmod 600 STARK_*
${DA}/get_OP > /dev/null
#
cp ${RUNTMPDIR}/${code2}_${modcode}${sys} .
./${code2}_${modcode}${sys} < DATEN
#
if test -s PLWF
then
  ln -sf PLWF in
fi
if test -s PLWFP
then
  cp PLWFP ${jobdir}/${name}_LF_${DATE1}.WFP_00005-02000
fi
#
rm *
#
#
# 3) standard near ultraviolet
#
# base frequency grid
cat > F_BASE << eos
2.00000D03 3.00000D03 1.00000D-01
eos
#
# attention: no lines of H I
#
if test -s ${RUNTMPDIR}/PURE_H
then
#
PURE_H=1
WAVE_H=4000
#
echo "H 1" > ION
#
cat > LINIEN << eos
BLENDRANGE 1800
H12 H110 3798.9 A
eos
#
else
#
PURE_H=0
WAVE_H=3000
#
echo "HE2" > ION
#
cp ${RUNTMPDIR}/ION_02000-03000 ION
cp ${RUNTMPDIR}/LINIEN_02000-03000 LINIEN
#
fi
```

```

#
cat > DISTA << eos
2 0.01
4 0.02
6 0.03
8 0.05
13 0.10
14 50.00
15 0.1
eos
#
cat > DATEN << eos
OCCUPATION PROBABILITY FORMALISM FOR H1
OCCUPATION PROBABILITY FORMALISM FOR HE1
OCCUPATION PROBABILITY FORMALISM FOR HE2
OCCUPATION PROBABILITY FORMALISM FOR C1
OCCUPATION PROBABILITY FORMALISM FOR C2
OCCUPATION PROBABILITY FORMALISM FOR C3
OCCUPATION PROBABILITY FORMALISM FOR C4
OCCUPATION PROBABILITY FORMALISM FOR C5
OCCUPATION PROBABILITY FORMALISM FOR C6
OCCUPATION PROBABILITY FORMALISM FOR N1
OCCUPATION PROBABILITY FORMALISM FOR N2
OCCUPATION PROBABILITY FORMALISM FOR N3
OCCUPATION PROBABILITY FORMALISM FOR N4
OCCUPATION PROBABILITY FORMALISM FOR N5
OCCUPATION PROBABILITY FORMALISM FOR N6
OCCUPATION PROBABILITY FORMALISM FOR N7
OCCUPATION PROBABILITY FORMALISM FOR O1
OCCUPATION PROBABILITY FORMALISM FOR O2
OCCUPATION PROBABILITY FORMALISM FOR O3
OCCUPATION PROBABILITY FORMALISM FOR O4
OCCUPATION PROBABILITY FORMALISM FOR O5
OCCUPATION PROBABILITY FORMALISM FOR O6
OCCUPATION PROBABILITY FORMALISM FOR O7
OCCUPATION PROBABILITY FORMALISM FOR O8
.
OPACITY PROJECT RBF DATA: START AT EDGE
OPACITY PROJECT RBF DATA: MISSING HYDROGENIC
.
DYNAMIC DISTA 2000 3000 1.5
PRINT ATOMMODELLE(KURZFORM)
PRINT LINIEN UND BLENDS
PLOT EINZELNE LINIEN PLOTTER
PLOT EMERGENTER FLUSS FLAM / LAM, INTERVAL: 2000 3000.1
eos
#
cp ${MI} MODIN
cp ${RUNTMPDIR}/ATOMS_WFP_LF_02000-0${WAVE_H} ATOMS
cp ${GRP}/H1 STARK_H1
cp ${GRP}/HE1 STARK_HE1
cp ${GRP}/HE2 STARK_HE2
chmod 600 STARK_*
${DA}/get_OP > /dev/null
#
cp ${RUNTMPDIR}/${code2}_${modcode}${sys} .
./${code2}_${modcode}${sys} < DATEN
#
if test -s PLWFP
then
cp PLWFP ${jobdir}/${name}_LF_${DATE1}.WFP_02000-03000
fi
#
rm *
#
#
# 4) standard optical – infrared
#
# base frequency grid
cat > CONTS_MAN << eos
3000.87493

```

```

55014.99983
eos
cat > F_BASE << eos
2.99950D03 7.00000D03 1.00000D+00
7.00500D03 5.50005D04 5.00000D+00
eos
#
echo "H 1" > ION
#
cat > LINIEN << eos
BLENDRANGE      27000
H15             H110             30392 A
eos
#
cat > DISTA << eos
2      0.01
4      0.05
6      0.10
8      0.20
13     0.50
17     1.00
18     250.00
21     1.00
eos
#
cat > DATEN << eos
OCCUPATION PROBABILITY FORMALISM FOR H1
OCCUPATION PROBABILITY FORMALISM FOR HE1
OCCUPATION PROBABILITY FORMALISM FOR HE2
OCCUPATION PROBABILITY FORMALISM FOR C1
OCCUPATION PROBABILITY FORMALISM FOR C2
OCCUPATION PROBABILITY FORMALISM FOR C3
OCCUPATION PROBABILITY FORMALISM FOR C4
OCCUPATION PROBABILITY FORMALISM FOR C5
OCCUPATION PROBABILITY FORMALISM FOR C6
OCCUPATION PROBABILITY FORMALISM FOR N1
OCCUPATION PROBABILITY FORMALISM FOR N2
OCCUPATION PROBABILITY FORMALISM FOR N3
OCCUPATION PROBABILITY FORMALISM FOR N4
OCCUPATION PROBABILITY FORMALISM FOR N5
OCCUPATION PROBABILITY FORMALISM FOR N6
OCCUPATION PROBABILITY FORMALISM FOR N7
OCCUPATION PROBABILITY FORMALISM FOR O1
OCCUPATION PROBABILITY FORMALISM FOR O2
OCCUPATION PROBABILITY FORMALISM FOR O3
OCCUPATION PROBABILITY FORMALISM FOR O4
OCCUPATION PROBABILITY FORMALISM FOR O5
OCCUPATION PROBABILITY FORMALISM FOR O6
OCCUPATION PROBABILITY FORMALISM FOR O7
OCCUPATION PROBABILITY FORMALISM FOR O8
.
OPACITY PROJECT RBF DATA: START AT EDGE
OPACITY PROJECT RBF DATA: MISSING HYDROGENIC
.
DYNAMIC DISTA 3000 55000 100
PRINT ATOMMODELLE(KURZFORM)
PRINT LINIEN UND BLENDS
PLOT EINZELNE LINIEN PLOTTER
PLOT EMERGENTER FLUSS  FLAM / LAM, INTERVAL: 3000 55001
eos
#
cp ${MI} MODIN
cp ${RUNTMPDIR}/ATOMS_WFP_LF_03000-55000  ATOMS
cp ${GRP}/H1  STARK_H1
cp ${GRP}/HE1 STARK_HE1
cp ${GRP}/HE2 STARK_HE2
chmod 600 STARK_*
${DA}/get_OP > /dev/null
#
cp ${RUNTMPDIR}/${code2}_${modcode}${sys} .
./${code2}_${modcode}${sys} < DATEN

```

```

#
if test -s PLWFP
then
  cp PLWFP ${jobdir}/${name}_LF_${DATE1}.WFP_03000-55000
fi
#
# now check if the LINE1_PROF calculations worked
# if not, send an email
# the existence and the length of the files are checked
#
  cat > MAIL << EOF
Hi,

The model calculation worked, but the SED calculation failed in case of

file ${jobdir}/${name}_LF_${DATE1}.WFP_03000-55000
length $helpWFP
Hostname ${HOSTNAME}

Yours
TMAW Team
EOF
#
# first check the existence
#
if test ! -s ${jobdir}/${name}_LF_${DATE1}.WFP
then
  /usr/bin/mail -r "TMAW" -s "TMAW: LINE1_PROF problem" astro-tmaw@listserv.uni-tuebingen.de < \
  MAIL
  exit 3
fi
#
if test ! -s ${jobdir}/${name}_LF_${DATE1}.WFP_00005-02000
then
  /usr/bin/mail -r "TMAW" -s "TMAW: LINE1_PROF problem" astro-tmaw@listserv.uni-tuebingen.de < \
  MAIL
  exit 3
fi
#
if test ! -s ${jobdir}/${name}_LF_${DATE1}.WFP_02000-03000
then
  /usr/bin/mail -r "TMAW" -s "TMAW: LINE1_PROF problem" astro-tmaw@listserv.uni-tuebingen.de < \
  MAIL
  exit 3
fi
#
if test ! -s ${jobdir}/${name}_LF_${DATE1}.WFP_03000-55000
then
  /usr/bin/mail -r "TMAW" -s "TMAW: LINE1_PROF problem" astro-tmaw@listserv.uni-tuebingen.de < \
  MAIL
  exit 3
fi
#
# now check if they are longer than the header length
#
helpWFP='wc ${jobdir}/${name}_LF_${DATE1}.WFP | cut -c1-8'
counthelp='echo $helpWFP |grep -c " " '
if [ ${counthelp} -ge 1 ]
then
helpWFP='wc ${jobdir}/${name}_LF_${DATE1}.WFP | cut -d " " -f 2'
fi
if [ $helpWFP -lt 40 ]
then
  /usr/bin/mail -r "TMAW" -s "TMAW: LINE1_PROF length problem" astro-tmaw@listserv.uni-
  tuebingen.de < MAIL
  exit 3
fi
#
helpWFP='wc ${jobdir}/${name}_LF_${DATE1}.WFP_00005-02000 | cut -c1-8'
counthelp='echo $helpWFP |grep -c " " '
if [ ${counthelp} -ge 1 ]

```

```

then
helpWFP='wc ${jobdir}/${name}_LF_${DATE1}.WFP_00005-02000 | cut -d " " -f 2'
fi
if [ $helpWFP -lt 40 ]
then
/usr/bin/mail -r "TMAW" -s "TMAW: LINE1_PROF length problem" astro-tmaw@listserv.uni-
tuebingen.de < MAIL
exit 3
fi
#
helpWFP='wc ${jobdir}/${name}_LF_${DATE1}.WFP_02000-03000 | cut -c1-8'
counthelp='echo $helpWFP |grep -c " "'
if [ ${counthelp} -ge 1 ]
then
helpWFP='wc ${jobdir}/${name}_LF_${DATE1}.WFP_02000-03000 | cut -d " " -f 2'
fi
if [ $helpWFP -lt 40 ]
then
/usr/bin/mail -r "TMAW" -s "TMAW: LINE1_PROF length problem" astro-tmaw@listserv.uni-
tuebingen.de < MAIL
exit 3
fi
#
helpWFP='wc ${jobdir}/${name}_LF_${DATE1}.WFP_03000-55000 | cut -c1-8'
counthelp='echo $helpWFP |grep -c " "'
if [ ${counthelp} -ge 1 ]
then
helpWFP='wc ${jobdir}/${name}_LF_${DATE1}.WFP_03000-55000 | cut -d " " -f 2'
fi
if [ $helpWFP -lt 40 ]
then
/usr/bin/mail -r "TMAW" -s "TMAW: LINE1_PROF length problem" astro-tmaw@listserv.uni-
tuebingen.de < MAIL
exit 3
fi
#
rm *
# clean up all binaries compiled for this request
#rm ${BI}/*${modcode}*${sys}
#rm ${RUNTMPDIR}/pro2_${REGFILE}${sys}
#rm ${RUNTMPDIR}/line1_prof_${REGFILE}${sys}
#rm ${RUNTMPDIR}/pro2_${REGFILE}_LF${sys}
#
# do not edit the rest of this file
#####
set +x; ${HOME}/.jobend ${TMPDIR}

```

Listing A.14: Die `checkPARAMETER.sh`-Prozedur wird von `TMAW_iaat_prepare.sh` verwendet, um die Parameter der Atomdaten und Frequenzgitter für die Erstellung der Programmdatei zu überprüfen. Sie wurde im Rahmen dieser Arbeit erstellt.

```
#!/bin/bash
#####
# this procedure is used by TMAW_iaat_prepare to check if the actual
# parameters are larger than those in the PARAMETER file
# In case they are, the PARAMETER file is updated and compiled.
#
#
# LAST_MODIFICATION="24-Jan-2013 18:49:06"
#
# Copyright 2011-2013 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
# CHANGELOG:
#       31.03.2012 default PGI compilation target is x64
#       02.05.2012 check if the executable is younger than
#                   the PARAMETER files
#       25.05.2012 additional comments inserted
#       14.01.2013 compile with p64
#####
#
# first read input values (the code, the name of the PARAMETER file and
# the PARAMETER values) from the current requests
#
#
code=${1}
modcode=${2}
new[0]=${3}
new[1]=${4}
new[2]=${5}
new[3]=${6}
new[4]=${7}
new[5]=${8}
new[6]=${9}
new[7]=${10}
new[8]=${11}
new[9]=${12}
new[10]=${13}
new[11]=${14}
new[12]=${15}
new[13]=${16}
new[14]=${17}
new[15]=${18}
new[16]=${19}
new[17]=${20}
new[18]=${21}
new[19]=${22}
new[20]=${23}
new[21]=${23}
#
#
# now read the existing/old PARAMETER values
#
old=(`${HOME}/bin/read_PARAMETER.Linux_x64 << EOF
${code}
${modcode}
EOF
`)
#
```



```

# compare the old and the new values
# if the new values are bigger, set switch to 1
# (indicates that compiling is necessary)
#
if [ "${code}" = "pro2" ]
then
    if [ ${old[9]} -ne ${old[7]} ]
    then
        old[9]=${old[7]}
    fi
fi
switch=0
i=0
while [ $i -lt 22 ]
do
    if [ ${new[$i]} -gt ${old[$i]} ]
    then
        if [ $i -eq 9 ]
        then
            if [ "${code}" = "line1_prof" ]
            then
                if [ $switch -ne 1 ]
                then
                    switch=0
                fi
            fi
            else
                switch=1
            fi
        else
            new[$i]=${old[$i]}
        fi
        ((i++))
    done
    #
    #
    # if one of the new PARAMETER was bigger, replace this value and
    # compile TMAP with the new PARAMETER
    if [ $switch -eq 1 ]
    then
        #
        #
        ${HOME}/bin/create_PARA.Linux_x64 << EOF
        ${code}
        ${modcode}
        ${new[0]}
        ${new[1]}
        ${new[2]}
        ${new[3]}
        ${new[4]}
        ${new[5]}
        ${new[6]}
        ${new[7]}
        ${new[8]}
        ${new[9]}
        ${new[10]}
        ${new[11]}
        ${new[12]}
        ${new[13]}
        ${new[14]}
        ${new[15]}
        ${new[16]}
        ${new[17]}
        ${new[18]}
        ${new[19]}
        ${new[20]}
        ${new[21]}
        EOF
        #
        ssh ait320 load ${code} ${modcode} p64 2> /home/TMAW/requests/compile.out
        #

```

```

# now check if an error occured while compiling
# notify the administrator if necessary
#
count='grep -c 'error in f90' /home/TMAW/requests/compile.out'
#
if [ $count -gt 0 ]
then
    cat > MAIL << EOF
The compilation failed because the parameter combination is too large.
EOF
    /usr/bin/mail -r "TMAW" -s "TMAW Problem: PARAMETER too large" \
    astro-tmaw@listserv.uni-tuebingen.de < MAIL
    rm MAIL
fi
#
#
fi
#
# in case that modification of PARAMETER is more
# recent than that of executable re-compile
#
counthelp='ls -ltr ~/bimod/${code}_${modcode}.Linux_x64 \
~/PARAMETER/${code}/PARAM_${modcode}.INC ~/PARAMETER/${code}/PARAM1_${modcode}.INC \
| tail -1 | grep bimod | wc | cut -c1-10'
echo "number of latest files in bimod:${counthelp}"
if [ "$counthelp" = "          1 " ]
then
ls -ltr ~/bimod/${code}_${modcode}.Linux_x64
else
ssh ait320 load ${code} ${modcode} p64 2> /home/TMAW/requests/compile.out
fi
#
# and check again if an error occured while compiling
#
count='grep -c 'error in f90' /home/TMAW/requests/compile.out'
#
if [ $count -gt 0 ]
then
    cat > MAIL << EOF
The compilation failed because the parameter combination is too large.
EOF
    /usr/bin/mail -r "TMAW" -s "TMAW Problem: PARAMETER too large" \
    astro-tmaw@listserv.uni-tuebingen.de < MAIL
    rm MAIL
fi

```

## B.2. TMAW-Prozeduren, die regelmäßig automatisch laufen

Zusätzlich zu den eben beschriebenen Hauptprozeduren, gibt es noch einige, die im Rahmen der Qualitätskontrolle automatisch ausgeführt werden. Sie wurden im Rahmen dieser Arbeit erstellt und benachrichtigen den TMAW-Administrator regelmäßig.

Listing A.15: Die `resetPARAM.sh`-Prozedur wird einmal pro Woche ausgeführt und setzt die Parameter auf 0 zurück, um zu garantieren, dass regelmäßig neu kompiliert und kleinstmögliche Parameter in der Atmosphärenrechnung verwendet werden.

```
#!/bin/sh
#####
# This procedure is used by a cron-job every month to reset the
# parameter values of the PARAM_MASTER files to 0
#
# LAST_MODIFICATION="25-Jan-2013 08:09:15"
#
# Copyright 2011-2013 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
# remove all executables first
rm ${BI}/line1_prof*
rm ${BI}/pro2_*
#
# initialize PARAMETER files
#
cd ${HOME}/PARAMETER/pro2/
cp PARAM_MASTER_0.INC PARAM_MASTER.INC
cp PARA1_MASTER_0.INC PARA1_MASTER.INC
#
# the same for the lineformation PARAMETER
cp PARAM_MASTER_0.INC PARAM_MASTER_LF.INC
cp PARA1_MASTER_0.INC PARA1_MASTER_LF.INC
#
# and for the LINE1_PROF PARAMETER
cd ${HOME}/PARAMETER/line1_prof/
cp PARAM_MASTER_0.INC PARAM_MASTER.INC
cp PARA1_MASTER_0.INC PARA1_MASTER.INC
#
#####
```

Listing A.16: Die `compile_MASTER.sh`-Prozedur wird einmal pro Woche ausgeführt, um *TMAP* mit den aktuellen Parametern neu zu kompilieren. Dadurch wird die Verwendung der aktuellsten Programmversion gewährleistet.

```
#!/bin/bash
#####
# This procedure compiles PRO2 and LINE1_PROF using
# the MASTER PARAMETER files.
# It is used by a cronjob once a week.
#
# LAST_MODIFICATION="25-Jan-2013 08:10:55"
#
# Copyright 2011-2013 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
#
# clean up
#
if test -s /home/TMAW/requests/compile_pro2_MASTER.out
then
rm /home/TMAW/requests/compile_pro2_MASTER.out
fi
if test -s /home/TMAW/requests/compile_pro2_MASTER_LF.out
then
rm /home/TMAW/requests/compile_pro2_MASTER_LF.out
fi
if test -s /home/TMAW/requests/compile_line1_prof_MASTER.out
then
rm /home/TMAW/requests/compile_line1_prof_MASTER.out
fi
#
#
# compile
#
# PRO2
#####
ssh ait320 load pro2 MASTER p64 > compile_p64.out
ssh aitp20 load pro2 MASTER n64 > compile_n64.out
cat compile_p64.out compile_n64.out > /home/TMAW/requests/compile_pro2_MASTER.out
rm compile_?64.out
#
# check if an error occured while compiling
#
count='grep -c 'error in f90' /home/TMAW/requests/compile_pro2_MASTER.out'
#
# if there was an error, send an email notification
#
if [ $count -gt 0 ]
then
cat > MAIL << EOF
The compilation failed because the PARAMETER combination is too large (MASTER).
EOF
/usr/bin/mail -r "TMAW" \
-s "TMAW Problem: PARAMETER too large" \
-a /home/TMAW/requests/compile_pro2_MASTER.out \
astro-tmaw@listserv.uni-tuebingen.de < MAIL

rm MAIL
fi
#
# PRO2 (LF)
#####
ssh ait320 load pro2 MASTER_LF p64 > compile_p64.out
```

```

ssh aitp20 load pro2 MASTER_LF n64 > compile_n64.out
cat compile_p64.out compile_n64.out > /home/TMAW/requests/compile_pro2_MASTER_LF.out
rm compile_?64.out
#
count='grep -c 'error in f90' /home/TMAW/requests/compile.out'
if [ $count -gt 0 ]
then
    cat > MAIL << EOF
The compilation failed because the PARAMETER combination is too large (MASTER_LF).
EOF
    /usr/bin/mail -r "TMAW" \
        -s "TMAW Problem: PARAMETER too large" \
        -a /home/TMAW/requests/compile_pro2_MASTER_LF.out \
        astro-tmaw@listserv.uni-tuebingen.de < MAIL

rm MAIL
fi
#
# LINE1_PROF
#####
ssh ait320 load line1_prof MASTER p64 > compile_p64.out
ssh aitp20 load line1_prof MASTER n64 > compile_n64.out
cat compile_p64.out compile_n64.out > /home/TMAW/requests/compile_line1_prof_MASTER.out
rm compile_?64.out
#
count='grep -c 'error in f90' /home/TMAW/requests/compile_line1_prof_MASTER.out'
if [ $count -gt 0 ]
then
    cat > MAIL << EOF
The compilation failed because the parameter combination is too large (LINE1_PROF).
EOF
    /usr/bin/mail -r "TMAW" \
        -s "TMAW Problem: PARAMETER too large" \
        -a /home/TMAW/requests/compile_line1_prof_MASTER.out \
        astro-tmaw@listserv.uni-tuebingen.de < MAIL

rm MAIL
fi
#
echo "TMAP was compiled with the MASTER PARAMETER files." > MAIL
/usr/bin/mail -r "TMAW" \
    -s "PARAMETER MASTER" \
    astro-tmaw@listserv.uni-tuebingen.de < MAIL

rm MAIL
#####

```

Listing A.17: Die `jobcontrol.sh`-Prozedur überprüft täglich einmal, ob ein Request im `/requests/running` Verzeichnis unplanmäßig gestoppt hat.

```
#!/bin/sh
#####
# This script is used once a day to check if there are jobs in the running
# directory that are not running on the computer.
#
# uses jobcontrol2.bat
#
# LAST_MODIFICATION="25-Jan-2013 08:17:15"
#
# Copyright 2011-2013 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
# go to the running directory
#
cd ~/requests/running/
rm ~/requests/jobcontrol_file
#
# for every running request start ~/scripts/jobcontrol2.sh
#
cat > Reneu << eos
while [ \ $# -gt 0 ]
do
/home/TMAW/scripts/jobcontrol2.sh \ $1
#
shift
done
eos
#
chmod u+ux Reneu
./Reneu `find /home/TMAW/requests/running -mindepth 1 -maxdepth 1 -name "req*" \
| cut -c29-73 | sort `

rm ./Reneu
#
# if there were more requests than running on a specific machine,
# notify the administrator,
# attach the file with the requests that could have stopped
#
if test -s ~/requests/jobcontrol_file
then
#
    cat > MAIL << EOF
Dear TMAW administrator ,

some of the requests in ~/requests/running are idle.
Please check this immediately.

Yours
TMAW
EOF
/usr/bin/mail -r "TMAW" -s "TMAW INFO: Some jobs have stopped" \
-a ~/requests/jobcontrol_file astro-tmaw@listserv.uni-tuebingen.de < MAIL
rm MAIL
#
fi
#####
```

Listing A.18: Die `jobcontrol2.sh`-Prozedur wird von der `jobcontrol.sh`-Prozedur verwendet.

```
#!/bin/sh
#####
# This script is used by jobcontrol.bat once a day to check if there are
# jobs in the running directory that are not running on the computer.
#
# LAST_MODIFICATION="25-Jan-2013 08:24:18"
#
# Copyright 2011-2013 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
# take the name of the machine from the name of the request
#
reqfile=${1}
machine='echo $reqfile | cut -c36-45'
#
# log in on the machine and check how many TMAW jobs are running
#
runcount='ssh ${machine} ps -aefl | grep "0 R TMAW" | grep ${reqfile} | wc | cut -c1-8'
#
# if there are less jobs running than requests in the running directory,
# write the requests of this machine to a file
#
if [ $runcount -lt 1 ]
then
echo ${machine} >> ~/requests/jobcontrol_file
fi
#####
```

Listing A.19: Die `checktestgridH.sh`-Prozedur berechnet bei Bedarf einmal pro Woche ein kleines Testgitter und vergleicht die Ergebnisse mit einer früheren Berechnung. Damit ist garantiert, dass die Berechnung konstant nach gleichen Maßstäben funktioniert.

```
#!/bin/sh
#####
# This script is used every week in order to check whether
# the requests are still calculating without problems.
#
# LAST_MODIFICATION="25-Jan-2013 08:26:15"
#
# Copyright 2011-2013 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
# define directories and script to start preparation and calculation
#
REQDIR=${HOME}/requests
RUNDIR=${REQDIR}/running
GRIDDIR=${REQDIR}/GRID
IAAT_RUNSCRIPT=${HOME}/scripts/TMAW_iaat_run.sh
#
# move all results to the testgridH directory to avoid that the new testgrid
# run will stop without calculation (existing result file)
#
mv /home/TMAW/results/0100000_7.00_H_1.000E+00_HE_0.000E+00_C_0.000E+00_N\
_0.000E+00_O_0.000E+00*tgz ~/testgridH/results/
mv /home/TMAW/results/0100000_6.00_H_1.000E+00_HE_0.000E+00_C_0.000E+00_N\
_0.000E+00_O_0.000E+00*tgz ~/testgridH/results/
mv /home/TMAW/results/0100000_8.00_H_1.000E+00_HE_0.000E+00_C_0.000E+00_N\
_0.000E+00_O_0.000E+00*tgz ~/testgridH/results/
mv /home/TMAW/results/0120000_7.00_H_1.000E+00_HE_0.000E+00_C_0.000E+00_N\
_0.000E+00_O_0.000E+00*tgz ~/testgridH/results/
mv /home/TMAW/results/0120000_6.00_H_1.000E+00_HE_0.000E+00_C_0.000E+00_N\
_0.000E+00_O_0.000E+00*tgz ~/testgridH/results/
mv /home/TMAW/results/0120000_8.00_H_1.000E+00_HE_0.000E+00_C_0.000E+00_N\
_0.000E+00_O_0.000E+00*tgz ~/testgridH/results/
mv /home/TMAW/results/0080000_7.00_H_1.000E+00_HE_0.000E+00_C_0.000E+00_N\
_0.000E+00_O_0.000E+00*tgz ~/testgridH/results/
mv /home/TMAW/results/0080000_6.00_H_1.000E+00_HE_0.000E+00_C_0.000E+00_N\
_0.000E+00_O_0.000E+00*tgz ~/testgridH/results/
mv /home/TMAW/results/0080000_8.00_H_1.000E+00_HE_0.000E+00_C_0.000E+00_N\
_0.000E+00_O_0.000E+00*tgz ~/testgridH/results/
#
# remove the lineformation models to ensure that the testgrid will be
# calculated again, at least for the last step
rm ~/models/TMAW/0100000_6.00_H_1.000E+00_HE_0.000E+00_C_0.000E+00_N\
_0.000E+00_O_0.000E+00_NE_0.000E+00_MG_0.000E+00_WWWW_LF
rm ~/models/TMAW/0100000_7.00_H_1.000E+00_HE_0.000E+00_C_0.000E+00_N\
_0.000E+00_O_0.000E+00_NE_0.000E+00_MG_0.000E+00_WWWW_LF
rm ~/models/TMAW/0100000_8.00_H_1.000E+00_HE_0.000E+00_C_0.000E+00_N\
_0.000E+00_O_0.000E+00_NE_0.000E+00_MG_0.000E+00_WWWW_LF
rm ~/models/TMAW/0120000_6.00_H_1.000E+00_HE_0.000E+00_C_0.000E+00_N\
_0.000E+00_O_0.000E+00_NE_0.000E+00_MG_0.000E+00_WWWW_LF
rm ~/models/TMAW/0120000_7.00_H_1.000E+00_HE_0.000E+00_C_0.000E+00_N\
_0.000E+00_O_0.000E+00_NE_0.000E+00_MG_0.000E+00_WWWW_LF
rm ~/models/TMAW/0120000_8.00_H_1.000E+00_HE_0.000E+00_C_0.000E+00_N\
_0.000E+00_O_0.000E+00_NE_0.000E+00_MG_0.000E+00_WWWW_LF
rm ~/models/TMAW/0080000_6.00_H_1.000E+00_HE_0.000E+00_C_0.000E+00_N\
_0.000E+00_O_0.000E+00_NE_0.000E+00_MG_0.000E+00_WWWW_LF
rm ~/models/TMAW/0080000_7.00_H_1.000E+00_HE_0.000E+00_C_0.000E+00_N\
_0.000E+00_O_0.000E+00_NE_0.000E+00_MG_0.000E+00_WWWW_LF
```



```

rm ~/models/TMAW/0080000_8.00__H__1.000E+00_HE_0.000E+00_C__0.000E+00_N\
_0.000E+00_O__0.000E+00_NE_0.000E+00_MG_0.000E+00_WWW_LF
#
# copy the test-requests to the request directory
cp ~/testgridH/req* ~/requests/
#
# start the H requests with /home/TMAW/scripts/checktestgridH2.sh
#
cd ${REQDIR}

cat > Re << eos
while [ \ $# -gt 1 ]
do
REQFILE=${1}
/home/TMAW/scripts/checktestgridH2.sh \ $1
#
shift
done
eos
chmod u=u+x Re
./Re 'find . -name "req_iaat*2012-03-08_13:39:39" -print | cut -c3-70'
rm Re
#
# start the last request separately to ensure waiting a whole calculation time
#
REQFILE=$(ls -t1 req_iaat_*2012-03-08_13:39:39 | head -1)
mv ${REQFILE} ${RUNDIR}/${REQFILE}_${HOSTNAME}
REQFILE=${REQFILE}_${HOSTNAME}
#
RUNFILE=${RUNDIR}/${REQFILE}
echo "${REQFILE}" >> ${RUNFILE}
cp ${RUNFILE} ${GRIDDIR}/${REQFILE}_named
#
#
namedlines=$(wc -l ${GRIDDIR}/${REQFILE}_named | cut -c1-2)
if [ ${namedlines} -ne 18 ]
then
echo "error in request file ${GRIDDIR}/${REQFILE}_named"
exit
fi
#
nice -19 ${IAAT_RUNSCRIPT} < ${GRIDDIR}/${REQFILE}_named 2> ERROR_CHECK1_${HOSTNAME} \
> OUTPUT_CHECK1_${HOSTNAME}
rm ${GRIDDIR}/${REQFILE}_named
#
# wait another 10 seconds to ensure all requests finished calculation
sleep 10
#
# now all requests are finished
# check if all requests produced a result
#
switch=0
if test ! -s /home/TMAW/results/0100000_7.00__H__1.000E+00_HE_0.000E+00_C__\
0.000E+00_N__0.000E+00_O__0.000E+00*tgz
then
switch=1
fi
if test ! -s /home/TMAW/results/0120000_7.00__H__1.000E+00_HE_0.000E+00_C__\
0.000E+00_N__0.000E+00_O__0.000E+00*tgz
then
switch=1
fi
if test ! -s /home/TMAW/results/0080000_7.00__H__1.000E+00_HE_0.000E+00_C__\
0.000E+00_N__0.000E+00_O__0.000E+00*tgz
then
switch=1
fi
if test ! -s /home/TMAW/results/0100000_6.00__H__1.000E+00_HE_0.000E+00_C__\
0.000E+00_N__0.000E+00_O__0.000E+00*tgz
then
switch=1
fi

```

```

fi
if test ! -s /home/TMAW/results/0120000_6.00__H__1.000E+00_HE_0.000E+00_C__\
0.000E+00_N__0.000E+00_O__0.000E+00*tgz
then
switch=1
fi
if test ! -s /home/TMAW/results/0080000_6.00__H__1.000E+00_HE_0.000E+00_C__\
0.000E+00_N__0.000E+00_O__0.000E+00*tgz
then
switch=1
fi
if test ! -s /home/TMAW/results/0100000_8.00__H__1.000E+00_HE_0.000E+00_C__\
0.000E+00_N__0.000E+00_O__0.000E+00*tgz
then
switch=1
fi
if test ! -s /home/TMAW/results/0120000_8.00__H__1.000E+00_HE_0.000E+00_C__\
0.000E+00_N__0.000E+00_O__0.000E+00*tgz
then
switch=1
fi
if test ! -s /home/TMAW/results/0080000_8.00__H__1.000E+00_HE_0.000E+00_C__\
0.000E+00_N__0.000E+00_O__0.000E+00*tgz
then
switch=1
fi
# if one request did not produce a result, send an email notification
if [ $switch -eq 1 ]
then
echo "The testgrid run was not successful. Not all files have created a result" > MAIL
/usr/bin/mail -r "TMAW" -s "TMAW INFO: Testgrid run was not successful" \
astro-tmaw@listserv.uni-tuebingen.de < MAIL
rm MAIL
fi
#
# copy the new results to the testgridH directory
#
cp /home/TMAW/results/0100000_7.00__H__1.000E+00_HE_0.000E+00_C__0.000E+00_N\
_0.000E+00_O__0.000E+00*tgz ~/testgridH/results/
cp /home/TMAW/results/0100000_6.00__H__1.000E+00_HE_0.000E+00_C__0.000E+00_N\
_0.000E+00_O__0.000E+00*tgz ~/testgridH/results/
cp /home/TMAW/results/0100000_8.00__H__1.000E+00_HE_0.000E+00_C__0.000E+00_N\
_0.000E+00_O__0.000E+00*tgz ~/testgridH/results/
cp /home/TMAW/results/0120000_7.00__H__1.000E+00_HE_0.000E+00_C__0.000E+00_N\
_0.000E+00_O__0.000E+00*tgz ~/testgridH/results/
cp /home/TMAW/results/0120000_6.00__H__1.000E+00_HE_0.000E+00_C__0.000E+00_N\
_0.000E+00_O__0.000E+00*tgz ~/testgridH/results/
cp /home/TMAW/results/0120000_8.00__H__1.000E+00_HE_0.000E+00_C__0.000E+00_N\
_0.000E+00_O__0.000E+00*tgz ~/testgridH/results/
cp /home/TMAW/results/0080000_7.00__H__1.000E+00_HE_0.000E+00_C__0.000E+00_N\
_0.000E+00_O__0.000E+00*tgz ~/testgridH/results/
cp /home/TMAW/results/0080000_6.00__H__1.000E+00_HE_0.000E+00_C__0.000E+00_N\
_0.000E+00_O__0.000E+00*tgz ~/testgridH/results/
cp /home/TMAW/results/0080000_8.00__H__1.000E+00_HE_0.000E+00_C__0.000E+00_N\
_0.000E+00_O__0.000E+00*tgz ~/testgridH/results/
#
# start /home/TMAW/scripts/checktestgridH3.sh to check if a job failed
#
cd ~/testgridH/results/
rm /home/TMAW/testgridH/results/failed
cat > Re1 << EOF
while [ \ $# -gt 0 ]
do
echo "\${1}"
/home/TMAW/scripts/checktestgridH3.sh \$1
shift
done
EOF
chmod u+u+x Re1
./Re1 `find . -mindepth 1 -maxdepth 1 -name "*tgz" -print | cut -c3-170`
#

```

```
# send an email notification if a job failed or if job was successful
#
if test -s /home/TMAW/testgridH/results/failed
then
  echo "The testgrid run on IAAT was not successful." > MAIL
  /usr/bin/mail -r "TMAW" -s "TMAW INFO: Testgrid run on IAAT was not successful" \
  astro-tmaw@listserv.uni-tuebingen.de < MAIL
  rm MAIL
fi
if test ! -s /home/TMAW/testgridH/results/failed
then
  echo "The testgrid run on IAAT was successful." > MAIL
  /usr/bin/mail -r "TMAW" -s "TMAW INFO: Testgrid run on IAAT was successful" \
  astro-tmaw@listserv.uni-tuebingen.de < MAIL
  rm MAIL
fi
#
```

Listing A.20: Die checktestgridH2.sh-Prozedur wird von der checktestgridH.sh Prozedur verwendet.

```
#!/bin/sh
#####
# This script is used every week in order to check whether
# the requests are still calculating without problems.
#
# LAST_MODIFICATION="25-Jan-2013 08:28:15"
#
# Copyright 2011-2013 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
# define the directories
REQDIR=${HOME}/requests
RUNDIR=${REQDIR}/running
GRIDDIR=${REQDIR}/GRID
IAAT_RUNSCRIPT=${HOME}/scripts/TMAW_iaat_run.sh
REQFILE=${1}
mv ${REQFILE} ${RUNDIR}/${REQFILE}_${HOSTNAME}
REQFILE=${REQFILE}_${HOSTNAME}
#
RUNFILE=${RUNDIR}/${REQFILE}
echo "${REQFILE}" >> ${RUNFILE}
cp ${RUNFILE} ${GRIDDIR}/${REQFILE}_named
#
#
namedlines='wc -l ${GRIDDIR}/${REQFILE}_named | cut -c1-2'
if [ ${namedlines} -ne 18 ]
then
    echo "error in request file ${GRIDDIR}/${REQFILE}_named"
    exit
fi
#
# start the request
#
echo 'w | head -1' > load.dat
load=~rauch/tools/readload.Linux_X86 | tail -1'
while [ ${load} -ge 4 ]
do
    sleep 600
echo 'w | head -1' > load.dat
load=~rauch/tools/readload.Linux_X86 | tail -1'
done
    nice -19 ${IAAT_RUNSCRIPT} < ${GRIDDIR}/${REQFILE}_named 2> ERROR_CHECKtest_${HOSTNAME} \
> OUTPUT_CHECKtest_${HOSTNAME}&
    sleep 600
    rm ${GRIDDIR}/${REQFILE}_named
```

Listing A.21: Die checktestgridH3.sh-Prozedur wird von der checktestgridH2.sh-Prozedur verwendet.

```
#!/bin/sh
#####
# This script is used every week in order to check whether
# the requests are still calculating without problems.
#
# LAST_MODIFICATION="25-Jan-2013 08:28:15"
#
# Copyright 2011-2013 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
# unpack the result files
# and check if the SEDs deviate from those calculated earlier
#
cd ~/testgridH/results/
part='echo ${1} | cut -c1-30'
tar -xvf ${1}
diff /home/TMAW/testgridH/results/WFP/WFP_${1} /home/TMAW/testgridH/results/${part}*.WFP \
> /home/TMAW/testgridH/results/diffWFP
cd ~/testgridH/results/
count='wc /home/TMAW/testgridH/results/diffWFP |cut -d ' ' -f 1'
if [ "$count" = "" ]
then
count='wc /home/TMAW/testgridH/results/diffWFP |cut -d ' ' -f 2'
fi
if [ "$count" = "" ]
then
count='wc /home/TMAW/testgridH/results/diffWFP |cut -d ' ' -f 3'
fi
if [ "$count" = "" ]
then
count=50000
fi
# if a SED file deviates, write the file /home/TMAW/testgridH/results/failed
if [ ${count} -gt 6 ]
then
echo "${part} failed" > /home/TMAW/testgridH/results/failed
fi
```

### B.3. TMAW-Prozeduren zum Überprüfen der Requests nach Bedarf

Zusätzlich zu den automatisch ausgeführten Prozeduren wurden einige Prozeduren erstellt, die nach Bedarf „von Hand“ ausgeführt werden können.

Listing A.22: Die `status_check.sh`-Prozedur überprüft, in welchem Rechenschritt sich die einzelnen Berechnungen befinden.

```
#!/bin/sh
#####
# This script is to check the status of the jobs that are
# currently running.
#
# LAST_MODIFICATION="25-Jan-2013 08:32:35"
#
# Copyright 2011-2013 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
# go to the directory where the running requests are located
#
cd /home/TMAW/requests/running/
cat > Re << eos
  while [ \ $# -gt 1 ]
  do
    echo \ ${1}
    # use status_check2.sh to log in on the machine the job is running
    /home/TMAW/scripts/status_check2.sh \ $1
    shift
  done
eos
chmod u=u+x Re
# find all running jobs
./Re 'find . -name "req_iaat*" -print | cut -c3-70'
rm Re
#
# write an email that has a file with the states of the jobs attached
echo "The file attached shows the status of one specific job in the \
/requests/running directory. All other status files can be found in \
~/requests/running. They are named Requestname.out." > MAIL
/usr/bin/mail -r "TMAW" -s "TMAW INFO: Status of jobs" -a \
/home/TMAW/requests/running/*out astro-tmaw@listserv.uni-tuebingen.de < MAIL
rm MAIL
```

Listing A.23: Die `status_check2.sh`-Prozedur wird von der `status_check.sh`-Prozedur verwendet.

```
#!/bin/sh
#####
# This script is to check the status of the jobs that are
# currently running.
#
# LAST_MODIFICATION="25-Jan-2013 08:33:45"
#
# Copyright 2011-2013 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
#
cd /home/TMAW/requests/running/
#
# take the request's name from input
Reqname=${1}
# cut the name of the machine the job is running on
machine='echo $Reqname | cut -c36-45'
# log in on the machine and start /home/TMAW/scripts/status_check3.sh
ssh $machine /home/TMAW/scripts/status_check3.sh $Reqname;
#
#####
```

Listing A.24: Die status\_check3 . sh-Prozedur wird von der status\_check2 . sh-Prozedur verwendet.

```
#!/bin/sh
#####
# This script is to check the status of the jobs that are
# currently running.
#
# LAST_MODIFICATION="25-Jan-2013 08:34:06"
#
# Copyright 2011-2013 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
#
cd /home/TMAW/requests/running/
#
# take the request's name from input
Reqname=${1}
# go to the scratch directory and take the line with the state of
# the job from the output file
# write it to a file
  cd /scratch/weekly/TMAW/
  helpname='ls /scratch/weekly/TMAW/*/$Reqname | cut -c22-45'
  grep COMMENT ${helpname}/OUTPUT_2* > /home/TMAW/requests/running/${Reqname}.out
#####
```



Listing A.25: Mit der `delete_lastline.sh`-Prozedur können abgebrochene oder fertige Modelle neu gestartet werden.

```
#!/bin/sh
# usage : delete_lastcolumn.bat 2011-08-19_13:05:51
#####
# This script restarts all files of a given date.
#
# LAST_MODIFICATION="25-Jan-2013 08:35:15"
#
# Copyright 2011-2013 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
# ensure the script is used the correct way
case ${1} in
  201?-??-??:??:??:??) echo 'start ';;
  ' ') exit;;
esac
# restart all files with the given date
series=${1}
cat > Re << \eos
  while [ $# -gt 0 ]
  do
    set -vx
    echo $1
    echo 'START'
    newname='echo $1 |cut -c1-34'
    echo $newname
    echo "now updating $1"
    head -17 /home/TMAW/requests/failed/$1 > /home/TMAW/requests/$newname
    rm /home/TMAW/requests/failed/$1
  shift
  done
eos
chmod u=u+x Re
./Re 'find /home/TMAW/requests/failed/ -mindepth 1 -maxdepth 1 \
-name "req*${series}*" -print | cut -c28-74'
#####
```

Listing A.26: Alternativ zur `delete_lastline.sh`-Prozedur können abgebrochene oder fertige Modelle auch mit `restart.bat` neu gestartet werden. Diese Prozedur verschiebt jeweils nur einen Request in das Warteverzeichnis. Sie wurde nicht im Rahmen dieser Arbeit entwickelt.

```
#!/bin/sh
#####
## restart failed requests
#####
#
# Procedure to restart requests.
# It deletes the request_machine line in the request file ,
# eliminates the machine name from the file and
# moves it back to ~/requests.
#
#
# LAST_MODIFICATION="11-Jan-2013 08:03:20"
#
# Copyright 2011-2013 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#
#####
#
set -vx
#
case ${1} in
  '') echo "usage restart.bat <request file>"; exit;;
esac
#
requesthost=${1}
requestraw=echo ${requesthost} | cut -c1-34'
#
head -17 ${requesthost} > ${requestraw}
mv ${requestraw} ~/requests
rm ${requesthost}
#
ls -l ../${requestraw}
```

Listing A.27: Die Prozedur `restartall.bat` wurde zusammen mit `restart.bat` entwickelt und verwendet diese Prozedur, um mehrere Requests in das Warteverzeichnis.

```
#!/bin/sh
#####
## restart all failed requests in directory
#####
#
# Procedure to restart all requests of one date.
#
#
# LAST_MODIFICATION="11-Jan-2013 08:04:45"
#
# Copyright 2011-2013 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#
#####
#
case ${1} in
    '') echo "usage restartall.bat <time stamp>"; exit;;
esac
date=${1}
#
cat > Re << eos
while [ \ $# -gt 0 ]
do
    ./restart.bat \${1}
shift
done
eos
chmod u=u+x Re
./Re 'find . -name "req_*${date}*" -print '
#
rm ./Re
```

Listing A.28: Die `models_cleanup.sh`-Prozedur wurde erstellt, um das finale, konvergente Modell einer Rechnung zusammen mit den Atomdaten und Frequenzgittern in ein tar-Archiv zu packen. Die nicht mehr benötigten „Zwischenmodelle“ werden gelöscht.

```
#!/bin/sh
#####
# This script deletes all files that are not needed any more when a model
# is converged. Then it puts the converged model into a tarball.
#
# LAST_MODIFICATION="25-Jan-2013 08:37:15"
#
# Copyright 2011-2013 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
# stop if this script is not used as indicated
#
case ${1} in
    '') echo "usage models_cleanup.bat <modelname>"; exit ;;
esac
#
name=${1}
#
# check if the model is converged
# then remove the models from intermediate steps
# put everything into a tarball and rename it
#
if test -s /home/TMAW/models/TMAW/${name}_WWW.converged
then
ls /home/TMAW/models/TMAW/${name}_WWW_20??-??-??_??:??:??
ls /home/TMAW/models/TMAW/${name}_WWW_cont*
rm /home/TMAW/models/TMAW/${name}_WWW_20??-??-??_??:??:??
rm /home/TMAW/models/TMAW/${name}_WWW_cont*
rm /home/TMAW/models/TMAW/${name}_WWW_OT*
rm /home/TMAW/models/TMAW/${name}_WWW_UL*
rm /home/TMAW/models/TMAW/${name}_WWW_statonly*
gzip /home/TMAW/models/TMAW/${name}_WWW.ATOMS
gzip /home/TMAW/models/TMAW/${name}_WWW.FGRID
#
cd /home/TMAW/models/TMAW/
tar -czf ${name}.tgz ${name}_WWW*
mv ${name}.tgz ${name}_WWW.tgz
#
fi
#####
```

Listing A.29: Die `maintenance.sh`-Prozedur wurde nicht im Rahmen dieser Arbeit entwickelt. Sie wird für Wartungsphasen verwendet, in denen kein Request abgeschickt oder bearbeitet werden soll.

```
#!/bin/sh
#
# procedure to deactivate TMAW in case of maintenace
# and to activate it after its end
#
# LAST_MODIFICATION="24-Apr-2012 09:28:00"
#
# Copyright 2011 – 2013 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any changes are strictly prohibited
#
case $1 in
'end') rm -f /home/TMAW/MAINTENANCE/inprogress ;
      ln -sf /home/TMAW/public_html/TMAW_active.shtml /home/TMAW/public_html/index.\
      html ;
      echo "TMAW activated after maintenance" > .TMAW_MAIL ;
      /usr/bin/mail -r "TMAW" -s "TMAW INFO: TMAW maintenance ended" astro-tmaw@listserv.\
      uni-tuebingen.de < .TMAW_MAIL ;
      rm .TMAW_MAIL ;
      ;;
*) touch /home/TMAW/MAINTENANCE/inprogress ;
   ln -sf /home/TMAW/public_html/TMAW_maintenance.shtml /home/TMAW/public_html/index.\
   html ;
   echo "TMAW deactivated for maintenance" > .TMAW_MAIL ;
   /usr/bin/mail -r "TMAW" -s "TMAW INFO: TMAW maintenance started" astro-tmaw@listserv\
   .uni-tuebingen.de < .TMAW_MAIL ;
   rm .TMAW_MAIL ;
   ;;
esac
```

## C. TMAW-Prozeduren für die Berechnung über das AstroGrid-D

Für die Berechnung der Modelle auf AstroGrid-D-zugänglichen Computern startet `checkreq.sh` die zugehörigen `TMAW_grid_XXX.sh`-Prozeduren. Die Vorbereitung (Erzeugung der Atomdaten, Frequenzgitter und Programmdateien) läuft analog zu den Berechnungen auf IAAT-Rechnern ab. Die einzelnen Rechenschritte werden ebenfalls analog ausgeführt. Dafür wird die `TMAW_grid_jobfile.sh`-Prozedur mit an die AstroGrid-D-zugänglichen Rechner übergeben. Der Transfer vom IAAT zu über das AstroGrid-D erreichbaren Ressourcen läuft über den extra installierten AstroGrid-D @IAAT-Server, der sich außerhalb der IAAT-Firewall befindet. Die eingepackten tar-Archive werden zuerst auf den AstroGrid-D @IAAT-Server kopiert (`scp2li04.bat`), dann mit einer Perl-Prozedur auf AstroGrid-D-zugängliche Rechner transferiert und dort wird `TMAW_grid_jobfile.sh` gestartet. Nach der Berechnung werden die fertigen Modelle in ihrem tar-Archiv mit `scp2IAAT.bat` zurück auf die IAAT-Rechner kopiert. Auf dem AstroGrid-D @IAAT-Server wurde ebenfalls die automatische Berechnung eines kleinen Testgitters eingeführt (`checkgridH.bat`), um zu garantieren, dass die Berechnungen dauerhaft funktionieren. Es kann außerdem mit `checkquality.bat` überprüft werden, welche der Modelle eines Gitters konvergiert oder abgebrochen sind. Diese beiden Prozeduren wurden im Rahmen dieser Arbeit erzeugt, die anderen waren bereits vorhanden und wurden modifiziert.

Listing A.30: `TMAW_grid_run.sh`, die eigentliche Hauptprozedur. Stand: November 2012.

```
#!/bin/bash
set +x;. ${HOME}/scripts/runstart.sh
#####
# Procedure is started by a TMAW cronjob. It uses TMAW_grid_prepare.sh
# to produce model atoms, frequency grid files and executables. Then it
# puts everything into a tarball that it can be send to AstroGrid-D.
#
#
# LAST_MODIFICATION="27-Dec-2011 11:38:15"
#
# Copyright 2011-2012 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
# CHANGELOG:
#      26.10.2006 started and some tests
#      31.10.2006 new runstart.sh for second tmpdir
#                  tar and mail results
#      16.11.2006 added wavelength range input by user
#      19.12.2006 changed $name to global variable using export
#      08.01.2007 added plotting of T-structure
#      09.01.2007 added CNO abundances
#      21.02.2007 general revision
#      10.03.2007 test phase inside firewall
#      29.10.2007 four-machine tests
#      02.11.2007 output is sent to user compressed
#      24.11.2008 create tar ball for AstroGrid-D submission
#      25.11.2008 default 32-bit (X86) compilation
#      25.11.2008 user resolution
#      27.01.2010 astro-tmaw members excluded from info mails
#      04.02.2010 copy request file in working directory
#      01.09.2010 OP data included (limited to ground states)
#      10.11.2010 put the .TMPDIR-file into the tar file
#      01.02.2011 give AstroGrid-D jobs another ending (AGD)
#      28.11.2011 OP data are on the mashines -> correct/complete data
#                  Eddingtonlimit check
#                  everything including LINE1_PROF is calculated on AGD
```

```

#                 everything in 64 bit instead of 32
#                 27.12.2011 FLUXINTERVAL is given to run.bat to create plots on AGD
#####
#
iaatjobname=TMAW_iaat_jobfile.sh
iaatjobfile=${HOME}/scripts/${iaatjobname}
#
gridjobname=TMAW_grid_jobfile.sh
gridjobfile=${HOME}/scripts/${gridjobname}
#
preparejob=${HOME}/scripts/TMAW_grid_prepare.sh
#
REQDIR=${HOME}/requests
FAILDIR=${REQDIR}/failed
FINDIR=${REQDIR}/finished
RESDIR=${HOME}/jobs/TMAP_WEB
RESDIR2=${HOME}/results
VODIR=${HOME}/public_html/VO/fluxtables/H+He+C+N+O
WFPDIR=${HOME}/WFP
#
export RESDIR2
export RUNTMPDIR=${RUNTMPDIR}
#
i=0
while read line
do i=$((i+1))
    readarr[$i]=$line
    echo "$i : ${readarr[$i]}"
done
#
    logg=${readarr[1]}
    teff=${readarr[2]}
    addy=${readarr[3]}
    NAME_first=${readarr[4]}
    NAME_last=${readarr[5]}
    DATE=${readarr[6]}
    ABUNDHin=${readarr[7]}
    ABUNDHein=${readarr[8]}
    ABUNDCin=${readarr[9]}
    ABUNDNin=${readarr[10]}
    ABUNDOin=${readarr[11]}
    WAVBEG=${readarr[12]}
    WAVEND=${readarr[13]}
    WAVSTP=${readarr[14]}
    FLUXINTERVAL=${readarr[15]}
    REGFILE=${readarr[16]}
#
# global shell variable
export REGFILE
#
    REQFILE=${REQDIR}/running/${REGFILE}
    cp ${REQFILE} .
#
echo ${teff} > TEFF_I7.in
    TEFF='/home/rauch/bimod/TEFF_I7${sys} < TEFF_I7.in 2>/dev/null 1>&1'
echo ${logg} > LOGG.in
    LOGG='/home/rauch/bimod/LOGG${sys} < LOGG.in 2>/dev/null 1>&1'
    rm TEFF_I7.in LOGG.in
#
# check if request is above Eddington limit
    Eddi='${HOME}/bin/checkeddington.Linux_x64 << EOF
    ${TEFF} ${LOGG}
    EOF
    '
#
# check if this is a single request or a grid of requests
    REGFILEPART='echo ${REGFILE} | cut -c15-34'
    Reqnr1=$(ls ~/requests/req_*${REGFILEPART} | wc -l)
    Reqnr2=$(ls ~/requests/running/req_*${REGFILEPART}* | wc -l)
    Reqnr='expr $Reqnr1 + $Reqnr2'
#

```

```

if [ ${Eddi} -le 0 ]
then
# mv ${REQFILE} ~/requests/failed_eddi/
cat > MAIL << EOF
Dear ${NAME_first} ${NAME_last},

the requested parameters are above the Eddington limit.
The model-atmosphere calculation with a static code like TMAP
is numerically instable and presently impossible via TMAW.
Thus, your request has to be rejected.

Yours
TMAW Team
EOF
#
if [ ${Reqnr} -le 1 ]
then
echo $addy
/usr/bin/mail -r "TMAW" -s "TMAW INFO: Eddington limit exceeded, \
TMAW request rejected" -a ${REQFILE} ${addy} < MAIL
case ${addy} in
    rauch@astro.uni-tuebingen.de) /usr/bin/mail -r "TMAW" -s \
"TMAW INFO: Eddington limit exceeded, TMAW request rejected" -a \
${REQFILE} astro-tmaw@listserv.uni-tuebingen.de < MAIL ;;
    ringat@astro.uni-tuebingen.de) /usr/bin/mail -r "TMAW" -s \
"TMAW INFO: Eddington limit exceeded, TMAW request rejected" -a \
${REQFILE} astro-tmaw@listserv.uni-tuebingen.de < MAIL ;;
    *) /usr/bin/mail -r "TMAW" -s \
"TMAW INFO: Eddington limit exceeded, TMAW request rejected" -a \
${REQFILE} astro-tmaw@listserv.uni-tuebingen.de < MAIL ;;
esac
rm MAIL
fi
mv ${REQFILE} ~/requests/failed_eddi/
exit
fi
#
# normalize abundances (for model name)
#
echo "5 1 ${ABUNDHin} ${ABUNDHEin} ${ABUNDCin} ${ABUNDNin} ${ABUNDOin}" > normH
echo "5 2 ${ABUNDHin} ${ABUNDHEin} ${ABUNDCin} ${ABUNDNin} ${ABUNDOin}" > normHE
echo "5 3 ${ABUNDHin} ${ABUNDHEin} ${ABUNDCin} ${ABUNDNin} ${ABUNDOin}" > normC
echo "5 4 ${ABUNDHin} ${ABUNDHEin} ${ABUNDCin} ${ABUNDNin} ${ABUNDOin}" > normN
echo "5 5 ${ABUNDHin} ${ABUNDHEin} ${ABUNDCin} ${ABUNDNin} ${ABUNDOin}" > normO
#
ABUNDH='/home/rauch/bimod/TMAW_norm${sys} < normH 2>/dev/null 1>&1'
ABUNDHE='/home/rauch/bimod/TMAW_norm${sys} < normHE 2>/dev/null 1>&1'
ABUNDC='/home/rauch/bimod/TMAW_norm${sys} < normC 2>/dev/null 1>&1'
ABUNDN='/home/rauch/bimod/TMAW_norm${sys} < normN 2>/dev/null 1>&1'
ABUNDO='/home/rauch/bimod/TMAW_norm${sys} < normO 2>/dev/null 1>&1'
rm normH normHE normC normN normO
#
export HEmassfraction=${ABUNDHE}
name=${TEFF}_${LOGG}_${H}_${ABUNDH}_${HE}_${ABUNDHE}_${C}_${ABUNDC}_${N}_${ABUNDN}_${O}_${ABUNDO}_AGD
export $name
export namepart=__H_${ABUNDH}_${HE}_${ABUNDHE}_${C}_${ABUNDC}_${N}_${ABUNDN}_${O}_${ABUNDO}
#
#
# check if exactly the same parameters have already been done
#
# user choice
Wa=${WAVBEG}
We=${WAVEND}
#
if [ ${Wa} -lt 10000 ]
then
if [ ${Wa} -lt 1000 ]
then
if [ ${Wa} -lt 100 ]
then
if [ ${Wa} -lt 10 ]

```



```

        then
            Wanf="0000${Wa} "
        else
            Wanf="000${Wa} "
        fi
    else
        Wanf="00${Wa} "
    fi
else
    Wanf="0${Wa} "
fi
else
    Wanf="${Wa} "
fi
#
if [ ${We} -lt 10000 ]
then
    if [ ${We} -lt 1000 ]
    then
        if [ ${We} -lt 100 ]
        then
            if [ ${We} -lt 10 ]
            then
                Wend="0000${We} "
            else
                Wend="000${We} "
            fi
        else
            Wend="00${We} "
        fi
    else
        Wend="0${We} "
    fi
else
    Wend="${We} "
fi
#
USERINTERVAL=${Wanf}-${Wend}
#
#
# individual choice of wavelength ranges
#
if [ ${Reqnr} -le 1 ]
then
if test -s ${RESDIR2}/${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}.tgz
then
    cat > MAIL << EOF
Dear ${NAME_first} ${NAME_last},

your TMAW request (${DATE}) was successfully completed.
The result files can be retrieved via
    wget http://astro.uni-tuebingen.de/~TMAW/TMAW/results/${DATE}/all_models.tar .

If you have further questions, please do not hesitate to contact \
astro-tmaw@listserv.uni-tuebingen.de.

When publishing research making use of this service, please acknowledge:
"The TMAW service (http://astro.uni-tuebingen.de/~TMAW) used to calculate
theoretical spectra for this paper was constructed as part of the
activities of the German Astrophysical Virtual Observatory."

Yours
TMAW Team
EOF
#
cp ${RESDIR2}/${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}.tgz \
~/public_html/TMAW/results/${DATE}
#
ls ${RESDIR2}/*${namepart}*${FLUXINTERVAL}_${USERINTERVAL}.tgz | cut -c21-32 > \
/home/TMAW/public_html/TMAW/results/${DATE}/results
#

```

```

sed -e 's\_\ \g' /home/TMAW/public_html/TMAW/results/${DATE}/results > \
/home/TMAW/public_html/TMAW/results/${DATE}/results2
#
cp /home/TMAW/public_html/TMAW/plots/Display_request_${DATE}.plot Display_result.plot
sed -e 's\END\ \g' Display_result.plot > Display_results.plot
echo ' N=? PLOTSYMBOL=2 PEN=3 XYTABLE SELECT 1 2 SYMBOLSIZE=-.3 COLOR=1' >> \
Display_results.plot
echo 'COMMAND XLOG' >> Display_results.plot
echo 'COMMAND INCLUDE results2' >> Display_results.plot
echo 'FINISH' >> Display_results.plot
echo 'END' >> Display_results.plot
cp /atmosarray/TMAW/models/Edli_H.dat .
cp /home/TMAW/public_html/TMAW/results/${DATE}/results2 .
setenv PATH_WRPLOT /home/rauch/wrplot.dir
/home/rauch/wrplot.dir/proc.dir/wrps.com Display_results 2>/dev/null
ps2pdf Display_results.ps Display_results.pdf
mv Display_results.pdf /home/TMAW/public_html/TMAW/results/${DATE}/
chmod 664 /home/TMAW/public_html/TMAW/results/${DATE}/Display_results.pdf
#
tar -cvPf all_models.tar /home/TMAW/public_html/TMAW/results/${DATE}/*
mv all_models.tar /home/TMAW/public_html/TMAW/results/${DATE}/
chmod 644 /home/TMAW/public_html/TMAW/results/${DATE}/all_models.tar
#
/usr/bin/mail -r "TMAW" -s "TMAW: Final Results" -a \
/home/TMAW/public_html/TMAW/results/${DATE}/Display_results.pdf ${addy} < MAIL
case ${addy} in
    rauch@astro.uni-tuebingen.de) ;;
    ringat@astro.uni-tuebingen.de) ;;
    *) /usr/bin/mail -r "TMAW" -s \
"TMAW INFO: Final Results (${HOSTNAME})" -a \
/home/TMAW/public_html/TMAW/results/${DATE}/Display_results.pdf \
astro-tmaw@listserv.uni-tuebingen.de < MAIL ;;
esac
mv ${REQFILE} ${FINDIR}
rm MAIL
exit
fi
else
if test -s ${RESDIR2}/${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}.tgz
then
cp ${RESDIR2}/${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}.tgz \
~/public_html/TMAW/results/${DATE}/
chmod 644 ~/public_html/TMAW/results/${DATE}/
#
cat > MAIL <<EOF
Dear ${NAME_first} ${NAME_last},

the model ${name}_LF from your TMAW request ${DATE} was successfully calculated.
The result file can be retrieved via

wget http://astro.uni-tuebingen.de/~TMAW/TMAW/results/ \
${DATE}/${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}.tgz

You will be notified at the end of your request calculations to download all models at once.
If you have further questions,
please do not hesitate to contact astro-tmaw@listserv.uni-tuebingen.de.

When publishing research making use of this service, please acknowledge:
"The TMAW service (http://astro-uni-tuebingen.de/~TMAW) used to calculate
theoretical spectra for this paper was constructed as part of the
activities of the German Astrophysical Virtual Observatory."

Yours
TMAW Team
EOF
#
/usr/bin/mail -r "TMAW" -s "TMAW: Results" ${addy} < MAIL
case ${addy} in
    rauch@astro.uni-tuebingen.de) ;;
    ringat@astro.uni-tuebingen.de) ;;
    *) /usr/bin/mail -r "TMAW" -s \

```



```

    mv ${REGFILENEU} .
#   mv ${HOME}/requests/running/${REGFILE} .
#
# iaatjobfile will be executed at IAAT after AstroGrid-D model calculation
cp ${iaatjobfile} .
#
# put .TMPDIR-file into tarball
echo /scratch/weekly/${LOGNAME} > ${HOME}/.TMPDIR
cp ${HOME}/.TMPDIR .
#
# gridjobfile is executed on AstroGrid-D resources by run.bat
cp ${gridjobfile} .
mkdir bimod
#
# default 32-bit binaries (X86)
cp /home/rauch/bimod/lte2.Linux_* bimod
cp /home/TMAW/bimod/pro2_MASTER*.Linux_* bimod
cp /home/TMAW/bimod/line1_prof_MASTER*.Linux_* bimod
cd bimod
cd ..
#
#
mkdir models
mkdir models/TMAW/
mkdir jobs/
mkdir jobs/TMAP_WEB
cp /home/TMAW/models/TMAW/${name}_WWW models/TMAW/
mkdir WFP
mkdir results
#
# cleanup file (deletes unnecessary files after model-atmosphere calculation)
echo "rm -rf data" > cleanup.bat
#
chmod 700 bimod/* cleanup.bat run.bat
#
# tar ball for AstroGrid-D submission
cd ..
TARNAME='echo ${REGFILE} | sed -e 's\:\_\g''
tar -czf /home/TMAW/GRID/${TARNAME}.tgz TMAP
#
#
TARNAME2='echo ${TARNAME} | cut -c16-50'
#
echo "${TARNAME2}" > /home/TMAW/GRID/waiting
exit
fi
#
# do not edit the rest of this file
#####
#set +x; ${HOME}/.jobend ${RUNTMPDIR}

```

Listing A.31: Die TMAW\_grid\_prepare.sh-Prozedur zur Erzeugung der für die Modellatmosphärenrechnung benötigten Atomdaten, Frequenzgitter und Programmdateien. Stand: November 2012.

```
#!/bin/ksh
set +x;. ${HOME}/.jobstart
# do not edit the beginning of this file
#####
# Procedure to create model atoms, frequency grid file and
# executables from the corresponding Parameter values.
#
#
# LAST_MODIFICATION="21-Mar-2012 12:38:15"
#
# Copyright 2007-2012 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
# CHANGELOG:
# 10.01.2007 started, ATOMS2 call
# 11.01.2007 SETF2 call
# 12.01.2007 parameter extraction
# 15.01.2007 added second SETF2 call for NLTE FGRID
# 21.02.2007 general revision
# 05.03.2007 normalize abundances
# formatted parameters coded in model name
# 28.06.2007 Teff-dependent model ions for carbon
# 24.10.2007 individual binaries (machine + request date)
# 13.12.2007 C LF model ions
# 11.01.2008 new paths for model-atom files
# 12.01.2008 Teff-dependent model ions for oxygen
# 28.01.2008 Teff-dependent model ions for nitrogen
# 19.02.2008 O LF model ions
# 23.02.2008 N LF model ions
# 29.02.2008 default PARAMETER for LINE1_PROF increased
# 13.03.2008 default PARAMETER for LINE1_PROF increased
# 25.11.2008 user resolution
# 26.11.2008 compilation of GRID binaries
# 05.01.2010 default PARAMETER for LINE1_PROF increased
# 25.03.2010 default PARAMETER for LINE1_PROF increased
# 03.09.2010 rm TMAW_preparing moved to IAAT_grid_run.sh
# 08.09.2010 IABUNDXX variable introduced
# 07.10.2010 IABUNDXX bug eliminated
# 08.08.2011 wait 240 sec for existing TMAW_preparing
# 17.08.2011 LF files for ATOMS, FGRID, PARAMETER
# 22.08.2011 only one MASTER parameter file is used-> no compiling
#
# 29.11.2011 the MASTER parameter file is checked and eventually
# created again by checkPARA.sh
# 07.12.2011 TMAW_preparing, generated by checkreq.sh is deleted
# after the compilation part
# 09.01.2012 NRBFMAW has to be increased to 39999999
# wrange for OUTPUT from ATOMS2 was corrected
# 13.01.2012 NRBFMAW has to be increased to 99999999
# 19.03.2012 rm /home/TMAW/bimod/TMAW_preparing*
# NRBBMAD smaller for LINE1_PROF
# 21.03.2012 diretory structure in adaten/rawdata changed -> path adjusted
#####
# IMPORTANT:
#
# TMAW_prepare
# needs the following parameters to run
#
```

```

# First command line parameter: T_eff
# Second           : H abundance
# Third            : He abundance
# Fourth           : C abundance
# Fifth            : N abundance
# Sixth            : O abundance
# Seventh          : lower limit of wavelength interval
# Eighth           : upper limit of wavelength interval
# Ninth            : user resolution
# Tenth            : request date
#####
#
# TMAW_prepare job following ##
#####
#
# own definitions
#
ADATADIR=${HOME}/adaten/rawdata
H_mod=H/MODELING
H_syn=H/SYNSPEC
HEmod=HE/MODELING
HEsyn=HE/SYNSPEC
C_mod=C/MODELING
C_syn=C/SYNSPEC
N_mod=N/MODELING
N_syn=N/SYNSPEC
O_mod=O/MODELING
O_syn=O/SYNSPEC
H_mod=H/MODELING
#
BI=/home/rauch/bimod
load=/home/rauch/jobs/load/load
#
runsys=${sys}
#
# check if all parameters are present
#
if [ $# -lt 9 ]
then
    echo "TMAW_prepare: wrong number of arguments"
    exit 3
fi
TEFF=${1}
ABUNDH=${2}
ABUNDHE=${3}
ABUNDC=${4}
ABUNDN=${5}
ABUNDO=${6}
WAVBEG=${7}
WAVEND=${8}
WAVSTP=${9}
DATE=${10}
#
# combination of input model atoms for ATOMS2 according to elements requested
#
#
# convert abundances into normalized number fractions (for LTE2)
#
echo "5 1 ${ABUNDH} ${ABUNDHE} ${ABUNDC} ${ABUNDN} ${ABUNDO}" > normH
echo "5 2 ${ABUNDH} ${ABUNDHE} ${ABUNDC} ${ABUNDN} ${ABUNDO}" > normHE
echo "5 3 ${ABUNDH} ${ABUNDHE} ${ABUNDC} ${ABUNDN} ${ABUNDO}" > normC
echo "5 4 ${ABUNDH} ${ABUNDHE} ${ABUNDC} ${ABUNDN} ${ABUNDO}" > normN
echo "5 5 ${ABUNDH} ${ABUNDHE} ${ABUNDC} ${ABUNDN} ${ABUNDO}" > normO
#
Hfnum='/home/rauch/bimod/TMAW_mass2num${sys} < normH      2>/dev/null 1>&1'
HEfnum='/home/rauch/bimod/TMAW_mass2num${sys} < normHE     2>/dev/null 1>&1'
Cfnum='/home/rauch/bimod/TMAW_mass2num${sys} < normC       2>/dev/null 1>&1'
Nfnum='/home/rauch/bimod/TMAW_mass2num${sys} < normN       2>/dev/null 1>&1'
Ofnum='/home/rauch/bimod/TMAW_mass2num${sys} < normO       2>/dev/null 1>&1'
#
echo ${Hfnum} > IABUNDHin

```

```

echo ${HEfnum} > IABUNDHEin
echo ${Cfnum} > IABUNDCin
echo ${Nfnum} > IABUNDNin
echo ${Ofnum} > IABUNDOin
#
IABUNDH= '/home/rauch/bimod/TMAW_gt${sys} < IABUNDHin 2>/dev/null 1>&1'
IABUNDHE= '/home/rauch/bimod/TMAW_gt${sys} < IABUNDHEin 2>/dev/null 1>&1'
IABUNDC= '/home/rauch/bimod/TMAW_gt${sys} < IABUNDCin 2>/dev/null 1>&1'
IABUNDN= '/home/rauch/bimod/TMAW_gt${sys} < IABUNDNin 2>/dev/null 1>&1'
IABUNDO= '/home/rauch/bimod/TMAW_gt${sys} < IABUNDOin 2>/dev/null 1>&1'
#
rm norm*
#
# using H as base for model atoms
cat ${ADATADIR}/${H_mod}/H > source
cat ${ADATADIR}/${H_mod}/H > source_LF
echo "ABUNDANCE H ${Hfnum}" > ABUNDIN
#
#
# model atoms used depend on T_eff
#
# selection of model atoms for atmosphere calculation
#
if [ ${TEFF} -lt 20000 ]
then
HE_model=HE_cool
C_model=C_I-IV
N_model=N_I-IV
O_model=O_I-IV
else
if [ ${TEFF} -le 30000 ]
then
HE_model=HE_cool
C_model=C_I-V
N_model=N_I-V
O_model=O_I-V
else
if [ ${TEFF} -le 40000 ]
then
HE_model=HE_cool
C_model=C_I-V
N_model=N_I-V
O_model=O_II-V
else
if [ ${TEFF} -le 70000 ]
then
HE_model=HE_cool
C_model=C_II-V
N_model=N_II-VI
O_model=O_II-VI
else
if [ ${TEFF} -le 100000 ]
then
HE_model=HE_cool
C_model=C_III-V
N_model=N_III-VI
O_model=O_III-VII
else
if [ ${TEFF} -le 140000 ]
then
HE_model=HE_hot
C_model=C_III-V
N_model=N_III-VI
O_model=O_III-VII
else
if [ ${TEFF} -le 150000 ]
then
HE_model=HE_hot
C_model=C_III-V
N_model=N_IV-VII
O_model=O_IV-VII

```

```

else
  if [ ${TEFF} -le 180000 ]
  then
    HE_model=HE_hot
    C_model=C_IV-VI
    N_model=N_IV-VII
    O_model=O_IV-VII
  else
    if [ ${TEFF} -le 200000 ]
    then
      HE_model=HE_hot
      C_model=C_IV-VI
      N_model=N_IV-VII
      O_model=O_V-VII
    else
      if [ ${TEFF} -le 300000 ]
      then
        HE_model=HE_hot
        C_model=C_IV-VII
        N_model=N_V-VIII
        O_model=O_V-VII
      else
        HE_model=HE_hot
        C_model=C_IV-VII
        N_model=N_V-VIII
        O_model=O_V-VIII
      fi
    fi
  fi
fi
fi
fi
fi
fi
fi
fi
fi
#
#
# combination of model atoms
#
PURE_H=1
#
if [ ${IABUNDHE} -gt 0 ]
then
  cat ${ADATADIR}/${HEmod}/${HE_model} >> source
  cat ${ADATADIR}/${HEmod}/${HE_model}_LF >> source_LF
  echo "ABUNDANCE HE ${HEfnum}" >> ABUNDIN
  PURE_H=0
fi
if [ ${IABUNDC} -gt 0 -a ${C_model} != 'none' ]
then
  cat ${ADATADIR}/${C_mod}/${C_model} >> source
  cat ${ADATADIR}/${C_mod}/${C_model}_LF >> source_LF
  echo "ABUNDANCE C ${Cfnum}" >> ABUNDIN
  PURE_H=0
fi
if [ ${IABUNDN} -gt 0 -a ${N_model} != 'none' ]
then
  cat ${ADATADIR}/${N_mod}/${N_model} >> source
  cat ${ADATADIR}/${N_mod}/${N_model}_LF >> source_LF
  echo "ABUNDANCE N ${Nfnum}" >> ABUNDIN
  PURE_H=0
fi
if [ ${IABUNDO} -gt 0 -a ${O_model} != 'none' ]
then
  cat ${ADATADIR}/${O_mod}/${O_model} >> source
  cat ${ADATADIR}/${O_mod}/${O_model}_LF >> source_LF
  echo "ABUNDANCE O ${Ofnum}" >> ABUNDIN
  PURE_H=0
fi
#

```



```

if [ ${PURE_H} -eq 1 ]
then
  echo "${PURE_H}" > ${RUNTMPDIR}/PURE_H
fi
#
cp ABUNDIN ${RUNTMPDIR}
#
#
# creation of atomic data files with ATOMS2
# extract parameters for PRO2 and LINE1_PROF
#
#
# LTE2 model atom
#
if [ ${IABUNDHE} -gt 0 ]
then
  HEX=''
else
  HEX='.'
fi
#
cat > opti << EOF
AUTO ION H      I    10  16 -10
${HEX}AUTO ION HE  II   14  32 -14
CBB-AUTO-FILL
CBF-AUTO-FILL
CBX-AUTO-FILL
RBB-AUTO-FILL  ( NONE)
RBF-AUTO-FILL  ( NOOP)
RDI-AUTO-FILL  ( NONE)
END OPTIONS
EOF
cat opti source > SOURCE
${BI}/atoms2${sys} > OUTPUT_ATOMS2_LTE
cp ATOMS ${RUNTMPDIR}/ATOMS_LTE
echo "new LTE model atom created"
ls -l ${RUNTMPDIR}/ATOMS_LTE
#
rm opti ATOMS
#
#
# PRO2 model atom
#
cat > opti << EOF
AUTO ION H      I    10  16 -10
${HEX}AUTO ION HE  II   14  32 -14
CBB-AUTO-FILL
CBF-AUTO-FILL
CBX-AUTO-FILL
RBF-AUTO-FILL
RDI-AUTO-FILL  ( NONE)
END OPTIONS
EOF
cat opti source > SOURCE
#
${BI}/atoms2${sys} > OUTPUT_ATOMS2_NLTE
cp ATOMS ${RUNTMPDIR}/ATOMS_NLTE
echo "new NLTE model atom created"
ls -l ${RUNTMPDIR}/ATOMS_NLTE
#
rm opti source ATOMS
#
#
# PRO2 lineformation model atom
#
cat > opti << EOF
AUTO ION H      I    15  16 -15
${HEX}AUTO ION HE  II   16  32 -16
CBB-AUTO-FILL
CBF-AUTO-FILL
CBX-AUTO-FILL

```

```

RBF-AUTO-FILL
RDI-AUTO-FILL ( NONE)
END OPTIONS
EOF
cat opti source_LF > SOURCE
#
${BI}/atoms2${sys} > OUTPUT_ATOMS2_NLTE_LF
cp ATOMS ${RUNTMPDIR}/ATOMS_NLTE_LF
echo "new NLTE lineformation model atom created"
ls -l ${RUNTMPDIR}/ATOMS_NLTE_LF
#
rm opti source_LF ATOMS
#
#
# LINE1_PROF model atoms
#
#
# using H_synspec as base for model atoms
cat ${ADATADIR}/${H_syn}/H_synspec > source
#
#
# model atoms used depend on T_eff
#
# selection of model atoms for synthetic spectrum
#
if [ ${TEFF} -lt 20000 ]
then
  HE_model=HE_cool_synspec
  C_model=C_I-IV_synspec
  N_model=N_I-IV_synspec
  O_model=O_I-IV_synspec
else
  if [ ${TEFF} -le 30000 ]
  then
    HE_model=HE_cool_synspec
    C_model=C_I-V_synspec
    N_model=N_I-V_synspec
    O_model=O_I-V_synspec
  else
    if [ ${TEFF} -le 40000 ]
    then
      HE_model=HE_cool_synspec
      C_model=C_I-V_synspec
      N_model=N_I-V_synspec
      O_model=O_II-V_synspec
    else
      if [ ${TEFF} -le 70000 ]
      then
        HE_model=HE_cool_synspec
        C_model=C_II-V_synspec
        N_model=N_II-VI_synspec
        O_model=O_II-VI_synspec
      else
        if [ ${TEFF} -le 100000 ]
        then
          HE_model=HE_cool_synspec
          C_model=C_III-V_synspec
          N_model=N_III-VI_synspec
          O_model=O_III-VII_synspec
        else
          if [ ${TEFF} -le 140000 ]
          then
            HE_model=HE_hot_synspec
            C_model=C_III-V_synspec
            N_model=N_III-VI_synspec
            O_model=O_III-VII_synspec
          else
            if [ ${TEFF} -le 150000 ]
            then
              HE_model=HE_hot_synspec
              C_model=C_III-V_synspec

```

```

        N_model=N_IV-VII_synspec
        O_model=O_IV-VII_synspec
    else
        if [ ${TEFF} -le 180000 ]
        then
            HE_model=HE_hot_synspec
            C_model=C_IV-VI_synspec
            N_model=N_IV-VII_synspec
            O_model=O_IV-VII_synspec
        else
            if [ ${TEFF} -le 200000 ]
            then
                HE_model=HE_hot_synspec
                C_model=C_IV-VI_synspec
                N_model=N_IV-VII_synspec
                O_model=O_IV-VII_synspec
            else
                if [ ${TEFF} -le 300000 ]
                then
                    HE_model=HE_hot_synspec
                    C_model=C_IV-VII_synspec
                    N_model=N_V-VIII_synspec
                    O_model=O_V-VII_synspec
                else
                    HE_model=HE_hot_synspec
                    C_model=C_IV-VII_synspec
                    N_model=N_V-VIII_synspec
                    O_model=O_V-VIII_synspec
                fi
            fi
        fi
    fi
fi
fi
fi
fi
fi
fi
fi
#
#
# combination of model atoms for lineformation
#
if [ ${IABUNDHE} -gt 0 ]
then
    cat ${ADATADIR}/${HEsyn}/${HE_model} >> source_LF
fi
if [ ${IABUNDC} -gt 0 -a ${C_model} != 'none' ]
then
    cat ${ADATADIR}/${C_syn}/${C_model} >> source_LF
fi
if [ ${IABUNDN} -gt 0 -a ${N_model} != 'none' ]
then
    cat ${ADATADIR}/${N_syn}/${N_model} >> source_LF
fi
if [ ${IABUNDO} -gt 0 -a ${O_model} != 'none' ]
then
    cat ${ADATADIR}/${O_syn}/${O_model} >> source_LF
fi
#
#
#
#####
# user's choice
Wa=${WAVBEG}
We=${WAVEND}
#
if [ ${Wa} -lt 10000 ]
then
    if [ ${Wa} -lt 1000 ]
    then
        if [ ${Wa} -lt 100 ]

```

```

    then
    if [ ${Wa} -lt 10 ]
    then
        Wanf="0000${Wa}"
    else
        Wanf="000${Wa}"
    fi
    else
        Wanf="00${Wa}"
    fi
else
    Wanf="0${Wa}"
fi
else
    Wanf="${Wa}"
fi
#
if [ ${We} -lt 10000 ]
then
    if [ ${We} -lt 1000 ]
    then
        if [ ${We} -lt 100 ]
        then
            if [ ${We} -lt 10 ]
            then
                Wend="0000${We}"
            else
                Wend="000${We}"
            fi
        else
            Wend="00${We}"
        fi
    else
        Wend="0${We}"
    fi
else
    Wend="${We}"
fi
#
#
cat > opti << EOF
AUTO ION H    l    10  16 -10
${HEX}AUTO ION HE ll    14  32 -14
LINEFORMATION - RBB-INTERVALL=[ ${Wanf}, ${Wend}], ION= NON
RBF-AUTO-FILL
RDI-AUTO-FILL ( NONE)
END OPTIONS
EOF
cat opti source > SOURCE
${BI}/atoms2${sys} > OUTPUT_ATOMS2_WFP
cp ATOMS ${RUNTMPDIR}/ATOMS_WFP
echo "new WFP model atom created"
ls -l ${RUNTMPDIR}/ATOMS_WFP
#
echo "${TEFF}" > SDATEN
echo "PRINT ASCII" >> SDATEN
echo "1.30E+16" >> SDATEN
echo "-1" >> SDATEN
#
${BI}/setf2${sys} < SDATEN > OUTPUT_SETF2_UsersChoice
cp ION_USER LINIEN_USER ${RUNTMPDIR}
#
mv F_BASE F_BASE_UsersChoice
rm opti source ATOMS
#
cat > opti << EOF
AUTO ION H    l    15  16 -15
${HEX}AUTO ION HE ll    14  32 -14
LINEFORMATION - RBB-INTERVALL=[ ${Wanf}, ${Wend}], ION= NON
RBF-AUTO-FILL
RDI-AUTO-FILL ( NONE)

```

```

END OPTIONS
EOF
cat opti source_LF > SOURCE
${Bl}/atoms2${sys} > OUTPUT_ATOMS2_WFP_LF
cp ATOMS ${RUNTMPDIR}/ATOMS_WFP_LF
echo "new WFP model atom created"
ls -l ${RUNTMPDIR}/ATOMS_WFP_LF
#
echo "${TEFF}" > SDATEN
echo "PRINT ASCII" >> SDATEN
echo "1.30E+16" >> SDATEN
echo "-1" >> SDATEN
#
${Bl}/setf2${sys} < SDATEN > OUTPUT_SETF2_UsersChoice_LF
cp ION_USER ${RUNTMPDIR}/ION_USER_LF
cp LINIEN_USER ${RUNTMPDIR}/LINIEN_USER_LF
#
mv F_BASE F_BASE_UsersChoice_LF
rm opti source_LF ATOMS
#
#
# using H_synspec as base for model atoms
cat ${ADATADIR}/${H_syn}/H_synspec > source_LF
#
#
# model atoms used depend on T_eff
#
if [ ${IABUNDHE} -gt 0 ]
then
cat ${ADATADIR}/${HEsyn}/${HE_model} >> source_LF
fi
if [ ${IABUNDC} -gt 0 -a ${C_model} != 'none' ]
then
cat ${ADATADIR}/${C_syn}/${C_model} >> source_LF
fi
if [ ${IABUNDN} -gt 0 -a ${N_model} != 'none' ]
then
cat ${ADATADIR}/${N_syn}/${N_model} >> source_LF
fi
if [ ${IABUNDO} -gt 0 -a ${O_model} != 'none' ]
then
cat ${ADATADIR}/${O_syn}/${O_model} >> source_LF
fi
#
#####
# standard ionizing flux
Wa=5
We=2000
#
if [ ${Wa} -lt 10000 ]
then
if [ ${Wa} -lt 1000 ]
then
if [ ${Wa} -lt 100 ]
then
if [ ${Wa} -lt 10 ]
then
Wanf="0000${Wa}"
else
Wanf="000${Wa}"
fi
else
Wanf="00${Wa}"
fi
else
Wanf="0${Wa}"
fi
else
Wanf="${Wa}"
fi
#

```

```

if [ ${We} -lt 10000 ]
then
  if [ ${We} -lt 1000 ]
  then
    if [ ${We} -lt 100 ]
    then
      if [ ${We} -lt 10 ]
      then
        Wend="0000${We}"
      else
        Wend="000${We}"
      fi
    else
      Wend="00${We}"
    fi
  else
    Wend="0${We}"
  fi
fi
Wend="${We}"
#
#
cat > opti << EOF
AUTO ION H    l    10  16 -10
${HEX}AUTO ION HE ll  14  32 -14
LINEFORMATION - RBB-INTERVALL=[ ${Wanf}, ${Wend}], ION= NON
RBF-AUTO-FILL
RDI-AUTO-FILL ( NONE)
END OPTIONS
EOF
cat opti source > SOURCE
${BI}/atoms2${sys} > OUTPUT_ATOMS2_WFP_${Wanf}-${Wend}
cp ATOMS ${RUNTMPDIR}/ATOMS_WFP_${Wanf}-${Wend}
echo "new WFP model atom created"
ls -l ${RUNTMPDIR}/ATOMS_WFP_${Wanf}-${Wend}
#
rm opti source ATOMS
#
#
cat > opti << EOF
AUTO ION H    l    15  16 -15
${HEX}AUTO ION HE ll  14  32 -14
LINEFORMATION - RBB-INTERVALL=[ ${Wanf}, ${Wend}], ION= NON
RBF-AUTO-FILL
RDI-AUTO-FILL ( NONE)
END OPTIONS
EOF
cat opti source_LF > SOURCE
${BI}/atoms2${sys} > OUTPUT_ATOMS2_WFP_LF_${Wanf}-${Wend}
cp ATOMS ${RUNTMPDIR}/ATOMS_WFP_LF_${Wanf}-${Wend}
echo "new WFP model atom created"
ls -l ${RUNTMPDIR}/ATOMS_WFP_LF_${Wanf}-${Wend}
#
rm opti source_LF ATOMS
#
# using H_synspec as base for model atoms
cat ${ADATADIR}/${H_syn}/H_synspec > source_LF
#
#
# model atoms used depend on T_eff
#
if [ ${IABUNDHE} -gt 0 ]
then
  cat ${ADATADIR}/${HEsyn}/${HE_model} >> source_LF
fi
if [ ${IABUNDC} -gt 0 -a ${C_model} != 'none' ]
then
  cat ${ADATADIR}/${C_syn}/${C_model} >> source_LF
fi
if [ ${IABUNDN} -gt 0 -a ${N_model} != 'none' ]

```

```

then
  cat ${ADATADIR}/${N_syn}/${N_model} >> source_LF
fi
if [ ${IABUNDO} -gt 0 -a ${O_model} != 'none' ]
then
  cat ${ADATADIR}/${O_syn}/${O_model} >> source_LF
fi
#
#####
# standard near ultraviolet
Wa=2000
if [ ${PURE_H} -eq 1 ]
then
  We=4000
else
  We=3000
fi
#
if [ ${Wa} -lt 10000 ]
then
  if [ ${Wa} -lt 1000 ]
  then
    if [ ${Wa} -lt 100 ]
    then
      if [ ${Wa} -lt 10 ]
      then
        Wanf="0000${Wa} "
      else
        Wanf="000${Wa} "
      fi
    else
      Wanf="00${Wa} "
    fi
  else
    Wanf="0${Wa} "
  fi
else
  Wanf="${Wa} "
fi
#
if [ ${We} -lt 10000 ]
then
  if [ ${We} -lt 1000 ]
  then
    if [ ${We} -lt 100 ]
    then
      if [ ${We} -lt 10 ]
      then
        Wend="0000${We} "
      else
        Wend="000${We} "
      fi
    else
      Wend="00${We} "
    fi
  else
    Wend="0${We} "
  fi
else
  Wend="${We} "
fi
#
#
cat > opti << EOF
AUTO ION H      I  10 16 -10
${HEX}AUTO ION HE  II  14 32 -14
LINEFORMATION - RBB-INTERVALL=[ ${Wanf} , ${Wend} ] , ION= NON
RBF-AUTO-FILL
RDI-AUTO-FILL ( NONE)
END OPTIONS
EOF

```

```

cat opti source > SOURCE
${BI}/atoms2${sys} > OUTPUT_ATOMS2_WFP_${Wanf}-${Wend}
cp ATOMS ${RUNTMPDIR}/ATOMS_WFP_${Wanf}-${Wend}
echo "new WFP model atom created"
ls -l ${RUNTMPDIR}/ATOMS_WFP_${Wanf}-${Wend}
#
rm opti source ATOMS
#
#
cat > opti << EOF
AUTO ION H 1 15 16 -15
${HEX}AUTO ION HE 11 14 32 -14
LINEFORMATION - RBB-INTERVALL=[ ${Wanf}, ${Wend}], ION= NON
RBF-AUTO-FILL
RDI-AUTO-FILL ( NONE)
END OPTIONS
EOF
cat opti source_LF > SOURCE
${BI}/atoms2${sys} > OUTPUT_ATOMS2_WFP_LF_${Wanf}-${Wend}
cp ATOMS ${RUNTMPDIR}/ATOMS_WFP_LF_${Wanf}-${Wend}
echo "new WFP model atom created"
ls -l ${RUNTMPDIR}/ATOMS_WFP_LF_${Wanf}-${Wend}
#
rm opti source_LF ATOMS
#
# using H_synspec as base for model atoms
cat ${ADATADIR}/${H_syn}/H_synspec > source
#
#
# model atoms used depend on T_eff
#
if [ ${IABUNDHE} -gt 0 ]
then
cat ${ADATADIR}/${HEsyn}/${HE_model} >> source_LF
fi
if [ ${IABUNDC} -gt 0 -a ${C_model} != 'none' ]
then
cat ${ADATADIR}/${C_syn}/${C_model} >> source_LF
fi
if [ ${IABUNDN} -gt 0 -a ${N_model} != 'none' ]
then
cat ${ADATADIR}/${N_syn}/${N_model} >> source_LF
fi
if [ ${IABUNDO} -gt 0 -a ${O_model} != 'none' ]
then
cat ${ADATADIR}/${O_syn}/${O_model} >> source_LF
fi
#
#####
# standard optical - infrared
Wa=3000
We=55000
#
if [ ${Wa} -lt 10000 ]
then
if [ ${Wa} -lt 1000 ]
then
if [ ${Wa} -lt 100 ]
then
if [ ${Wa} -lt 10 ]
then
Wanf="0000${Wa}"
else
Wanf="000${Wa}"
fi
else
Wanf="00${Wa}"
fi
else
Wanf="0${Wa}"
fi

```



```

else
  Wanf="${Wa}"
fi
#
if [ ${We} -lt 10000 ]
then
  if [ ${We} -lt 1000 ]
  then
    if [ ${We} -lt 100 ]
    then
      if [ ${We} -lt 10 ]
      then
        Wend="0000${We}"
      else
        Wend="000${We}"
      fi
    else
      Wend="00${We}"
    fi
  else
    Wend="0${We}"
  fi
fi
else
  Wend="${We}"
fi
#
#
cat > opti << EOF
AUTO ION H      I      10  16 -10
${HEX}AUTO ION HE  II      14  32 -14
LINEFORMATION - RBB-INTERVALL=[ ${Wanf}, ${Wend}], ION= NON
RBF-AUTO-FILL
RDI-AUTO-FILL ( NONE)
END OPTIONS
EOF
cat opti source > SOURCE
${Bl}/atoms2${sys} > OUTPUT_ATOMS2_WFP_${Wanf}-${Wend}
cp ATOMS ${RUNTMPDIR}/ATOMS_WFP_${Wanf}-${Wend}
echo "new WFP model atom created"
ls -l ${RUNTMPDIR}/ATOMS_WFP_${Wanf}-${Wend}
#
rm opti source ATOMS
#
#
cat > opti << EOF
AUTO ION H      I      15  16 -15
${HEX}AUTO ION HE  II      14  32 -14
LINEFORMATION - RBB-INTERVALL=[ ${Wanf}, ${Wend}], ION= NON
RBF-AUTO-FILL
RDI-AUTO-FILL ( NONE)
END OPTIONS
EOF
cat opti source_LF > SOURCE
${Bl}/atoms2${sys} > OUTPUT_ATOMS2_WFP_LF_${Wanf}-${Wend}
cp ATOMS ${RUNTMPDIR}/ATOMS_WFP_LF_${Wanf}-${Wend}
echo "new WFP model atom created"
ls -l ${RUNTMPDIR}/ATOMS_WFP_LF_${Wanf}-${Wend}
#
rm opti source_LF ATOMS
#
# creation of frequency-grid files with SETF2
# extract parameters for PRO2 and LINE1_PROF
#
#####
# LTE frequency grid
#
cp ${RUNTMPDIR}/ATOMS_LTE ATOMS
#
echo "2.100D+02 8.000D+02 1.000D+01" > section1
echo "8.100D+02 1.000D+03 1.000D+00" > section2

```

```

${BI}/prep_conts_man${sys} > F_BASE < section1
${BI}/prep_conts_man${sys} >> F_BASE < section2
#
echo "${TEFF}" > SDATEN
echo "PRINT CHECK" >> SDATEN
echo "1.30E+16" >> SDATEN
echo "-1" >> SDATEN
#
if [ ${TEFF} -lt 81000 ]
then
  if [ ${TEFF} -lt 50000 ]
  then
    if [ ${TEFF} -lt 30000 ]
    then
      echo "100 10" >> SDATEN
    else
      echo "100 10" >> SDATEN
    fi
  else
    echo " 30 10" >> SDATEN
  fi
else
  echo " 10 10" >> SDATEN
fi
#
${BI}/setf2${sys} < SDATEN > OUTPUT_SETF2_LTE
#
if test -f FGRID
then
  cp FGRID ${RUNTMPDIR}/FGRID_LTE
  chmod 600 ${RUNTMPDIR}/FGRID_LTE
  echo "new LTE frequency grid created"
  ls -l ${RUNTMPDIR}/FGRID_LTE
  rm section* ATOMS F_BASE FGRID SDATEN
else
  echo "no LTE frequency grid created"
  exit 3
fi
#
#####
# NLTE frequency grid
#
cp ${RUNTMPDIR}/ATOMS_NLTE ATOMS
#
if [ ${TEFF} -ge 50000 ]
then
  if [ ${TEFF} -ge 100000 ]
  then
    echo "1.100D+01 8.000D+02 1.000D+01" > section1
  else
    echo "1.000D+02 8.000D+02 1.000D+01" > section1
  fi
else
  if [ ${TEFF} -lt 30000 ]
  then
    echo "1.810D+02 8.000D+02 1.000D+01" > section1
  else
    echo "1.010D+02 8.000D+02 1.000D+01" > section1
  fi
fi
echo "8.0100D+02 9.0000D+02 1.0000D+00" > section2
echo "9.0010D+02 1.0000D+03 1.0000D-01" > section3
echo "1.0010D+03 1.3000D+03 5.0000D-01" > section4
echo "1.3020D+03 1.0000D+04 1.0000D+00" > section5
echo "1.0005D+04 5.5000D+04 5.0000D+00" > section6
#
${BI}/prep_conts_man${sys} > F_BASE < section1
${BI}/prep_conts_man${sys} >> F_BASE < section2
${BI}/prep_conts_man${sys} >> F_BASE < section3
${BI}/prep_conts_man${sys} >> F_BASE < section4

```

```

${Bl}/prep_conts_man${sys} >> F_BASE < section5
${Bl}/prep_conts_man${sys} >> F_BASE < section6
#
echo "${TEFF}" > SDATEN
echo "PRINT CHECK" >> SDATEN
echo "1.30E+16" >> SDATEN
echo "1" >> SDATEN
#
if [ ${TEFF} -lt 81000 ]
then
  if [ ${TEFF} -lt 50000 ]
  then
    if [ ${TEFF} -lt 30000 ]
    then
      echo "100 10" >> SDATEN
    else
      echo "100 10" >> SDATEN
    fi
  else
    echo " 30 10" >> SDATEN
  fi
else
  echo " 10 50" >> SDATEN
fi
#
${Bl}/setf2${sys} < SDATEN > OUTPUT_SETF2_NLTE
#
if test -f FGRID
then
  cp FGRID ${RUNTMPDIR}/FGRID_NLTE
  chmod 600 ${RUNTMPDIR}/FGRID_NLTE
  echo "new NLTE frequency grid created"
  ls -l ${RUNTMPDIR}/FGRID_NLTE
  rm section* ATOMS F_BASE FGRID SDATEN
else
  echo "no NLTE frequency grid created"
  exit 3
fi
#
#####
# NLTE frequency grid for lineformation
#
cp ${RUNTMPDIR}/ATOMS_NLTE_LF ATOMS
#
if [ ${TEFF} -ge 50000 ]
then
  if [ ${TEFF} -ge 100000 ]
  then
    echo "1.100D+01 8.000D+02 1.000D+01" > section1
  else
    echo "1.000D+02 8.000D+02 1.000D+01" > section1
  fi
else
  if [ ${TEFF} -lt 30000 ]
  then
    echo "1.810D+02 8.000D+02 1.000D+01" > section1
  else
    echo "1.010D+02 8.000D+02 1.000D+01" > section1
  fi
fi
echo "8.0100D+02 9.0000D+02 1.0000D+00" > section2
echo "9.0010D+02 1.0000D+03 1.0000D-01" > section3
echo "1.0010D+03 1.3000D+03 5.0000D-01" > section4
echo "1.3020D+03 1.0000D+04 1.0000D+00" > section5
echo "1.0005D+04 5.5000D+04 5.0000D+00" > section6
#
${Bl}/prep_conts_man${sys} > F_BASE < section1
${Bl}/prep_conts_man${sys} >> F_BASE < section2
${Bl}/prep_conts_man${sys} >> F_BASE < section3
${Bl}/prep_conts_man${sys} >> F_BASE < section4
${Bl}/prep_conts_man${sys} >> F_BASE < section5

```

```

${BI}/prep_conts_man${sys} >> F_BASE < section6
#
echo "${TEFF}" > SDATEN
echo "PRINT CHECK" >> SDATEN
echo "1.30E+16" >> SDATEN
echo "1" >> SDATEN
#
if [ ${TEFF} -lt 81000 ]
then
  if [ ${TEFF} -lt 50000 ]
  then
    if [ ${TEFF} -lt 30000 ]
    then
      echo "100 10" >> SDATEN
    else
      echo "100 10" >> SDATEN
    fi
  else
    echo " 30 10" >> SDATEN
  fi
else
  echo " 10 50" >> SDATEN
fi
#
${BI}/setf2${sys} < SDATEN > OUTPUT_SETF2_NLTE_LF
#
if test -f FGRID
then
  cp FGRID ${RUNTMPDIR}/FGRID_NLTE_LF
  chmod 600 ${RUNTMPDIR}/FGRID_NLTE_LF
  echo "new NLTE frequency grid created"
  ls -l ${RUNTMPDIR}/FGRID_NLTE_LF
  rm section* ATOMS F_BASE FGRID SDATEN
else
  echo "no NLTE frequency grid for lineformation created"
  exit 3
fi
#
# now check the outputs and if the parameter file must be changed,
# then re-compile
##
#####
## PARA_MASTER
#####
# Extracting from ATOMS2 output
NATOMAX='grep NATOMAX OUTPUT_ATOMS2_NLTE | awk '{ print $4 }''
NIONMAX='grep NIONMAX OUTPUT_ATOMS2_NLTE | awk '{ print $4 }''
NLMAX='grep NLMAX OUTPUT_ATOMS2_NLTE | awk '{ print $4 }''
LTEMAX='grep LTEMAX OUTPUT_ATOMS2_NLTE | awk '{ print $4 }''
NRBBMAX='grep NRBBMAX OUTPUT_ATOMS2_NLTE | awk '{ print $4 }''
NRBFMAX='grep NRBFMAX OUTPUT_ATOMS2_NLTE | awk '{ print $4 }''
NRDIMAX='grep NRDIMAX OUTPUT_ATOMS2_NLTE | awk '{ print $4 }''
NRFFMAX='grep NRFFMAX OUTPUT_ATOMS2_NLTE | awk '{ print $4 }''
NRLLMAX='grep NRLLMAX OUTPUT_ATOMS2_NLTE | awk '{ print $4 }''
NRLUMAX='grep NRLUMAX OUTPUT_ATOMS2_NLTE | awk '{ print $4 }''
NCBBMAX='grep NCBBMAX OUTPUT_ATOMS2_NLTE | awk '{ print $4 }''
NCBFMAX='grep NCBFMAX OUTPUT_ATOMS2_NLTE | awk '{ print $4 }''
NCBXMAY='grep NCBXMAY OUTPUT_ATOMS2_NLTE | awk '{ print $4 }''
# Extracting from SETF2 output
NFMAX='grep NFMAX OUTPUT_SETF2_NLTE | awk '{ print $4 }''
NRBBMAX='grep NRBBMAX OUTPUT_SETF2_NLTE | awk '{ print $4 }''
NRBBMAD='grep NRBBMAD OUTPUT_SETF2_NLTE | awk '{ print $4 }''
NRBFMAW='grep NRBFMAW OUTPUT_SETF2_NLTE | awk '{ $foo=$9 ; sub(/\)/, "", $foo); print $foo \
}'
#
NRBLMAX=${NRBBMAX}
NCBBMAD=6
NCBFMAD=2
NRFFMAD=2
NCBXMAD=8
NRLFMAX=1

```

```

NRBFMAD=6
NRLLMAD=1
NRLUMAD=1
if [ ${NRLLMAX} -eq 0 ]
then
  NRLLMAX=1
fi
if [ ${NRLUMAX} -eq 0 ]
then
  NRLUMAX=1
fi
NL=1
#
~/scripts/checkPARAM.sh pro2 MASTER ${NLMAX} ${NIONMAX} ${NATOMAX} ${NFMAX} ${LTEMAX} ${NLMAX} \
  ${NRBBMAX} ${NRBFMAX} ${NCBBMAX} ${NCBFMAX} ${NRFFMAX} ${NCBMAX} ${NRLLMAX} ${NRLUMAX} \
  ${NRBBMAD} ${NRBFMAD} ${NCBBMAD} ${NCBFMAD} ${NRFFMAD} ${NCBMAD} ${NL} ${NL} 2>~/\
requests/checkPARAM_values

#
#
#####
## PARAMETER_MASTER_LF
#####
#### Extracting from ATOMS2 output
NATOMAX='grep NATOMAX OUTPUT_ATOMS2_NLTE_LF | awk '{ print $4 }''
NIONMAX='grep NIONMAX OUTPUT_ATOMS2_NLTE_LF | awk '{ print $4 }''
NLMAX='grep NLMAX OUTPUT_ATOMS2_NLTE_LF | awk '{ print $4 }''
LTEMAX='grep LTEMAX OUTPUT_ATOMS2_NLTE_LF | awk '{ print $4 }''
NRBBMAX='grep NRBBMAX OUTPUT_ATOMS2_NLTE_LF | awk '{ print $4 }''
NRBFMAX='grep NRBFMAX OUTPUT_ATOMS2_NLTE_LF | awk '{ print $4 }''
NRDIMAX='grep NRDIMAX OUTPUT_ATOMS2_NLTE_LF | awk '{ print $4 }''
NRFFMAX='grep NRFFMAX OUTPUT_ATOMS2_NLTE_LF | awk '{ print $4 }''
NRLLMAX='grep NRLLMAX OUTPUT_ATOMS2_NLTE_LF | awk '{ print $4 }''
NRLUMAX='grep NRLUMAX OUTPUT_ATOMS2_NLTE_LF | awk '{ print $4 }''
NCBBMAX='grep NCBBMAX OUTPUT_ATOMS2_NLTE_LF | awk '{ print $4 }''
NCBFMAX='grep NCBFMAX OUTPUT_ATOMS2_NLTE_LF | awk '{ print $4 }''
NCBMAX='grep NCBMAX OUTPUT_ATOMS2_NLTE_LF | awk '{ print $4 }''
# Extracting from SETF2 output
NFMAX='grep NFMAX OUTPUT_SETF2_NLTE_LF | awk '{ print $4 }''
NRBBMAX='grep NRBBMAX OUTPUT_SETF2_NLTE_LF | awk '{ print $4 }''
NRBBMAD='grep NRBBMAD OUTPUT_SETF2_NLTE_LF | awk '{ print $4 }''
NRBFMAW='grep NRBFMAW OUTPUT_SETF2_NLTE_LF | awk '{ $foo=$9 ; sub(/\)/, "", $foo); print \
  $foo }''

#
NRBLMAX=${NRBBMAX}
NCBBMAD=6
NCBFMAD=2
NRFFMAD=2
NCBMAD=8
NRLFMAX=1
NRBFMAD=6
NRLLMAD=1
NRLUMAD=1
if [ ${NRLLMAX} -eq 0 ]
then
  NRLLMAX=1
fi
if [ ${NRLUMAX} -eq 0 ]
then
  NRLUMAX=1
fi
NL=1
#
~/scripts/checkPARAM.sh pro2 MASTER_LF ${NLMAX} ${NIONMAX} ${NATOMAX} ${NFMAX} ${LTEMAX} ${\
  NLMAX} ${NRBBMAX} ${NRBFMAX} ${NCBBMAX} ${NCBFMAX} ${NRFFMAX} ${NCBMAX} ${NRLLMAX} ${\
  NRLUMAX} ${NRBBMAD} ${NRBFMAD} ${NCBBMAD} ${NCBFMAD} ${NRFFMAD} ${NCBMAD} ${NL} ${NL}

#
#####
## PARAMETER_MASTER_LINE1_PROF
#####
#### 1) user's choice
wrange=""

```

```

#
# Extracting from ATOMS2 output
NATOMAX='grep NATOMAX OUTPUT_ATOMS2_WFP_LF | awk '{ print $4 }''
NIONMAX='grep NIONMAX OUTPUT_ATOMS2_WFP_LF | awk '{ print $4 }''
NLMAX='grep NLMAX OUTPUT_ATOMS2_WFP_LF | awk '{ print $4 }''
LTEMAX='grep LTEMAX OUTPUT_ATOMS2_WFP_LF | awk '{ print $4 }''
NRBBMAX='grep NRBBMAX OUTPUT_ATOMS2_WFP_LF | awk '{ print $4 }''
NRBFMAX='grep NRBFMAX OUTPUT_ATOMS2_WFP_LF | awk '{ print $4 }''
NRDIMAX='grep NRDIMAX OUTPUT_ATOMS2_WFP_LF | awk '{ print $4 }''
NRFFMAX='grep NRFFMAX OUTPUT_ATOMS2_WFP_LF | awk '{ print $4 }''
NRLLMAX='grep NRLLMAX OUTPUT_ATOMS2_WFP_LF | awk '{ print $4 }''
NRLUMAX='grep NRLUMAX OUTPUT_ATOMS2_WFP_LF | awk '{ print $4 }''
#
NRBLMAX=${NRBBMAX}
NCBBMAD=6
NCBFMAD=2
NRFFMAD=2
NCBXMAD=8
NRLFMAX=1
NRBFMAD=6
NRLLMAD=1
NRLUMAD=1
# default settings
NFMAX=10000
NRBBMAX=1200
NRBBMAD=29999
NRBFMAW=99999999
NCBBMAX=1
NCBFMAX=1
NCBXMAD=1
#
if [ ${NRLLMAX} -eq 0 ]
then
NRLLMAX=1
fi
if [ ${NRLUMAX} -eq 0 ]
then
NRLUMAX=1
fi
NL=1
#
~/scripts/checkPARA.sh line1_prof MASTER ${NLMAX} ${NIONMAX} ${NATOMAX} ${NFMAX} ${LTEMAX} ${\
NL} ${NRBBMAX} ${NRBFMAX} ${NCBBMAX} ${NRBFMAW} ${NRFFMAX} ${NCBXMAD} ${NRLLMAX} ${\
NRLUMAX} ${NRBBMAD} ${NRBFMAD} ${NCBBMAD} ${NCBFMAD} ${NRFFMAD} ${NCBXMAD} ${NL} ${NL}
##
# 2) standard UV
wrange='_00005-02000'
#
# Extracting from ATOMS2 output
NATOMAX='grep NATOMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NIONMAX='grep NIONMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NLMAX='grep NLMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
LTEMAX='grep LTEMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRBBMAX='grep NRBBMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRBFMAX='grep NRBFMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRDIMAX='grep NRDIMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRFFMAX='grep NRFFMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRLLMAX='grep NRLLMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRLUMAX='grep NRLUMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
#
NRBLMAX=${NRBBMAX}
NCBBMAD=6
NCBFMAD=2
NRFFMAD=2
NCBXMAD=8
NRLFMAX=1
NRBFMAD=6
NRLLMAD=1
NRLUMAD=1
# default settings
NFMAX=10000

```

```

NRBBMAX=1200
NRBBMAD=29999
NRBFMAW=99999999
NCBBMAX=1
NCBFMAX=1
NCBXMAD=1
#
if [ ${NRLLMAX} -eq 0 ]
then
  NRLLMAX=1
fi
if [ ${NRLUMAX} -eq 0 ]
then
  NRLUMAX=1
fi
NL=1
#
~/scripts/checkPARAM.sh line1_prof MASTER ${NLMAX} ${NIONMAX} ${NATOMAX} ${NFMAX} ${LTEMAX} ${\
NL} ${NRBBMAX} ${NRBFMAX} ${NCBBMAX} ${NCBFMAX} ${NRFFMAX} ${NCBXMAD} ${NRLLMAX} ${\
NRLUMAX} ${NRBBMAD} ${NRBFMAD} ${NCBBMAD} ${NCBFMAD} ${NRFFMAD} ${NCBXMAD} ${NL} ${NL}
#
## 3) standard near infrared
if [ ${PURE_H} -eq 1 ]
then
  wrange='_02000-04000'
else
  wrange='_02000-03000'
fi
#
# Extracting from ATOMS2 output
NATOMAX='grep NATOMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NIONMAX='grep NIONMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NLMAX='grep NLMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
LTEMAX='grep LTEMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRBBMAX='grep NRBBMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRBFMAX='grep NRBFMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRDIMAX='grep NRDIMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRFFMAX='grep NRFFMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRLLMAX='grep NRLLMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRLUMAX='grep NRLUMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
#
NRBLMAX=${NRBBMAX}
NCBBMAD=6
NCBFMAD=2
NRFFMAD=2
NCBXMAD=8
NRLFMAX=1
NRBFMAD=6
NRLLMAD=1
NRLUMAD=1
# default settings
NFMAX=100000
NRBBMAX=1200
NRBBMAD=29999
NRBFMAW=99999999
NCBBMAX=1
NCBFMAX=1
NCBXMAD=1
#
if [ ${NRLLMAX} -eq 0 ]
then
  NRLLMAX=1
fi
if [ ${NRLUMAX} -eq 0 ]
then
  NRLUMAX=1
fi
#
NL=1
~/scripts/checkPARAM.sh line1_prof MASTER ${NLMAX} ${NIONMAX} ${NATOMAX} ${NFMAX} ${LTEMAX} ${\
NL} ${NRBBMAX} ${NRBFMAX} ${NCBBMAX} ${NCBFMAX} ${NRFFMAX} ${NCBXMAD} ${NRLLMAX} ${\

```

```

    NRLUMAX} ${NRBBMAD} ${NRBFMAD} ${NCBBMAD} ${NCBFMAD} ${NRFFMAD} ${NCBXMAX} ${NL} ${NL}
#
# 4) standard optical – infrared
wrange='_03000–55000'
#
# Extracting from ATOMS2 output
NATOMAX='grep NATOMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NIONMAX='grep NIONMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NLMAX='grep NLMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
LTEMAX='grep LTEMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRBBMAX='grep NRBBMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRBFMAX='grep NRBFMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRDIMAX='grep NRDIMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRFFMAX='grep NRFFMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRLLMAX='grep NRLLMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
NRLUMAX='grep NRLUMAX OUTPUT_ATOMS2_WFP_LF${wrange} | awk '{ print $4 }''
#
NRBLMAX=${NRBBMAX}
NCBBMAD=6
NCBFMAD=2
NRFFMAD=2
NCBXMAX=8
NRLFMAX=1
NRBFMAD=6
NRLLMAD=1
NRLUMAD=1
# default settings
NFMAX=100000
NRBBMAX=1200
NRBBMAD=29999
NRBFMAW=99999999
NCBBMAX=1
NCBFMAX=1
NCBXMAX=1
#
if [ ${NRLLMAX} -eq 0 ]
then
    NRLLMAX=1
fi
if [ ${NRLUMAX} -eq 0 ]
then
    NRLUMAX=1
fi
NL=1
#
~/scripts/checkPARAM.sh line1_prof MASTER ${NLMAX} ${NIONMAX} ${NATOMAX} ${NFMAX} ${LTEMAX} ${\
NL} ${NRBBMAX} ${NRBFMAX} ${NCBBMAX} ${NCBFMAX} ${NRFFMAX} ${NCBXMAX} ${NRLLMAX} ${\
NRLUMAX} ${NRBBMAD} ${NRBFMAD} ${NCBBMAD} ${NCBFMAD} ${NRFFMAD} ${NCBXMAX} ${NL} ${NL}
#
rm /home/TMAW/bimod/TMAW_preparing*
#
# do not edit the rest of this file
#####
set +x; ${HOME}/.jobend ${TMPDIR}

```



Listing A.32: Auszug aus der TMAW\_grid\_jobfile.sh-Prozedur. Diese Prozedur ist für die eigentliche Berechnung der Atmosphäre zuständig. Stand: November 2012.

```
#!/bin/sh
# do not edit the beginning of this file
#####
# Procedure to calculate a model in 9 (continuum) + 7 (line) + 3 (post) steps
# Afterwards SEDs are produced for all wavelength ranges.
#
#
# LAST_MODIFICATION="24-May-2012 10:23:15"
#
# Copyright 2006-2012 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
# CHANGELOG:
# 19.10.2006 changing TR grid jobfile to TMAP_WEB jobfile
# 19.12.2006 $name is now set globally in TMAW_iaat_run.sh
# 08.01.2007 added second NLTE job for lines, first NLTE
# run now without lines
# 15.01.2007 changed to TMAW_jobfile
# using ATOMS, FGRID and binaries prepared in
# TMAW_prepare.sh now.
# 21.02.2007 general revision
# 05.03.2007 4-step model calculation
# 10.10.2007 2000-3000A problem
# 24.10.2007 individual binaries (machine + request date)
# 02.11.2007 increased degree of convergence
# 04.11.2007 final_model
# 02.03.2008 occupation probability for all ions
# 03.03.2008 5-step continuum-model calculation
# 06.03.2008 7-step continuum-model calculation
# 07.03.2008 6-step line-model calculation
# 10.03.2008 7-step line-model calculation
# 22.05.2008 9-step continuum-model calculation
# 17.11.2008 DISTA for high g extended
# 21.11.2008 occupation probabilities for all ions
# 24.11.2008 grid version
# 25.11.2008 user resolution
# 09.04.2009 refined convergence strategy
# 08.07.2009 DISTA for high g extended (preview)
# 11.08.2009 database transfer only in case of convergence
# 14.01.2010 increase ITMAX and steps for convergence
# 02.03.2011 give out flux and T-structure in the last run
# 14.05.2011 change $HOME to $NEW_HOME
# the data directory is now laying on the frontend
# machine, therefore $HOME is needed
# 15.06.2011 OP files are located at the grid computers now
# -> change Path
# 17.08.2011 two additional steps with real T-correction and
# one with lineformation are implemented
# 14.12.2011 T-correction intervall changed to -3-10
# the plot and tar process is included
# 27.12.2011 FLUXINTERVAL is given to run.bat to do the plots
# the plot commands are adjusted because they run different on AGD
# 26.03.2012 post calculation steps adjusted to IAAT calculations
# 24.05.2012 USERINTERVAL was not defined->WAVBEG-WAVEND
#####
#
# IMPORTANT:
# JOBFIL FOR TMAW JOBS
#
```

```

# Needs parameters to run
# First command line parameter: T_eff
# Second           : log g
# Third           : lower limit of wavelength interval
# Fourth          : upper limit of wavelength interval
# Fifth           : user resolution
# Sixth           : request date
# Seventh          : model name
#####
#
# check if all parameters are present
#
if [ $# -lt 8 ]
then
    echo "wrong number of arguments"
    exit 3
fi
#
# TMAW job following
#####
#
# own definitions
#
# default 64-bit compilation
sys='.Linux_x64'
#
# for grid calculations only – actual directory is ${NEWHOME}
NEWHOME=${PWD}
RUNTMPDIR=${PWD}
#
# create local tmpdir
TMPDIR=${RUNTMPDIR}/TmpDir
mkdir ${TMPDIR}
cd ${TMPDIR}
#
mod=TMAW
#
#
name=${7}
FLUXINTERVAL=${8}
#
final_model=${NEWHOME}/models/${mod}/${name}_WWW
export final_model
#
code=pro2
code2=line1_prof
#
DATE=${6}
#
modcode=MASTER
modcode2=MASTER_LF
#
DA=${HOME}/data_GRID
BI=${NEWHOME}/bimod
JO=${NEWHOME}/jobs
jobdir=${JO}/TMAP_WEB
export jobdir
#
#— model parameters
#
#— Teff
TT=${1}
#
#— log g
GG=${2}
#
#— wavelength interval for flux table
WAVBEG=${3}
WAVEND=${4}
WAVSTP=${5}

```

```

#
#
if test -s ${final_model}.failed
then
  echo "a previous model calculation for ${name} failed , please wait for instructions"
  exit 3
fi
#
# start of: sequence, if model does not exist yet and is not currently calculating
#
#
if test ! -s ${final_model}.calculating \
  -a ! -s ${final_model}.converged \
  -a ! -s ${final_model}.failed
then
  echo "sequence started for ${name} at 'date +%H:%M:%S' 'date +%d-%m-%Y' at ${HOSTNAME}"
  echo "sequence started for ${name} at 'date +%H:%M:%S' 'date +%d-%m-%Y' at ${HOSTNAME}" >\
  ${final_model}.calculating
  touch ${final_model}.calculating@${HOSTNAME}
#
#
#
#
#####
#
# # # # #
# # # # #
# # # # #
# # # # #
# # # # #
# # # # #
# # # # #
#####
#
# start model
#
if test ! -s ${NEWHOME}/models/lte
then
  mkdir -p ${NEWHOME}/models/lte
fi
MO=${NEWHOME}/models/lte/${name}
#
#
# start of: calculate LTE2 model only if not already existing
if test ! -s ${MO}
then
#
#
echo "5000" > CPUTIME
#
if [ ${TT} -lt 30000 ]
then
  ITMAX=1
else
  ITMAX=100
fi
#
cat > LDATEN << eos
T EFF ${TT}
LOG G ${GG}
DAMP=0.1
ITMAX=${ITMAX}
EPS=1.0E-6
TAU SCALE 86 90 -2
.TAU SCALE MINIMUM -9
TAU SCALE MAXIMUM 4.0
PRINT INTEGRATED EDDINGTON FLUX,ITERATION:LAST
PRINT TEMPERATURE CORRECTIONS,ITERATION:LAST,ALL DEPTHS
.PRINT MODEL ATOMS (OVERVIEW)
eos
cat ${RUNTMPDIR}/ABUNDIN >> LDATEN

```

```

#
cp ${RUNTMPDIR}/ATOMS_LTE ATOMS
cp ${RUNTMPDIR}/FGRID_LTE FGRID
#
${BI}/lte2${sys}< LDATEN
#
if test -s MODELL
then
  cp MODELL ${MO}
  chmod 600 ${MO}
  echo "new LTE2 model ${MO} created"
  ls -l ${MO}
else
  cat "no new model ${MO} created" ERRORS > ${MO}.failed
  cat "no new model ${MO} created" ERRORS > ${MO}.failed@${HOSTNAME}
  echo "no LTE2 model ${MO} created"
  cat "no new model ${MO} created" ERRORS > ${final_model}.failed
  rm ${final_model}.calculating*
  # clean up all binaries compiled for this request
# rm ${BI}/*${modcode}*${sys}
  exit 3
fi
#
rm *
#
# end of: calculate LTE2 model only if not already existing
fi
#
#
#
#####
#
# ##### ##### ##### #####
# # # # # # #
# # # # # # #
# ##### ##### # # #####
# # # # # # #
# # # # # # #
# # # # ##### #####
#
#####
# NLTE model
#
#
if test ! -s ${NEWHOME}/models/${mod}
then
  mkdir -p ${NEWHOME}/models/${mod}
fi
#
#
# NLTE continuum model
# in nine steps
# step 1) REDUCE LOG CVEC -2.0, NO TEMPERATURE CORRECTION , STATISTICAL EQUATIONS ONLY
# step 2) REDUCE LOG CVEC -1.0, NO TEMPERATURE CORRECTION , STATISTICAL EQUATIONS ONLY
# step 3) NO TEMPERATURE CORRECTION , STATISTICAL EQUATIONS ONLY
# step 4) REDUCE LOG CVEC -2.0, NO TEMPERATURE CORRECTION
# step 5) REDUCE LOG CVEC -1.0, NO TEMPERATURE CORRECTION
# step 6) NO TEMPERATURE CORRECTION
# step 7) REDUCE LOG CVEC -2.0, TEMPERATURE CORRECTION (UL)
# step 8) REDUCE LOG CVEC -1.0, TEMPERATURE CORRECTION (UL)
# step 9) TEMPERATURE CORRECTION (UL)
#
#
#
# LTE2 model becomes start model
#
MI=${MO}
MO=${NEWHOME}/models/${mod}/${name}_WWW_statonly
#

```

```

if test -s ${MO}
then
  MI=${MO}
fi
#
#
#
# NLTE continuum model
#
# step 1) REDUCE LOG CVEC -2.0, STATISTICAL EQUATIONS ONLY
#
# start of: calculate NLTE continuum model only if not already existing
if test -s ${MI} \
-a ! -s ${MO} \
-a ! -s ${MO}.converged \
-a ! -s ${MO}.failed \
-a ! -s ${final_model}
then
#
cat > DATEN << eos
COMMENT: TMAW calculation of NLTE continuum model, step 1
.
OCCUPATION PROBABILITY FORMALISM FOR H1
OCCUPATION PROBABILITY FORMALISM FOR HE1
OCCUPATION PROBABILITY FORMALISM FOR HE2
OCCUPATION PROBABILITY FORMALISM FOR C1
OCCUPATION PROBABILITY FORMALISM FOR C2
OCCUPATION PROBABILITY FORMALISM FOR C3
OCCUPATION PROBABILITY FORMALISM FOR C4
OCCUPATION PROBABILITY FORMALISM FOR C5
OCCUPATION PROBABILITY FORMALISM FOR C6
OCCUPATION PROBABILITY FORMALISM FOR N1
OCCUPATION PROBABILITY FORMALISM FOR N2
OCCUPATION PROBABILITY FORMALISM FOR N3
OCCUPATION PROBABILITY FORMALISM FOR N4
OCCUPATION PROBABILITY FORMALISM FOR N5
OCCUPATION PROBABILITY FORMALISM FOR N6
OCCUPATION PROBABILITY FORMALISM FOR N7
OCCUPATION PROBABILITY FORMALISM FOR O1
OCCUPATION PROBABILITY FORMALISM FOR O2
OCCUPATION PROBABILITY FORMALISM FOR O3
OCCUPATION PROBABILITY FORMALISM FOR O4
OCCUPATION PROBABILITY FORMALISM FOR O5
OCCUPATION PROBABILITY FORMALISM FOR O6
OCCUPATION PROBABILITY FORMALISM FOR O7
OCCUPATION PROBABILITY FORMALISM FOR O8
.
OPACITY PROJECT RBF DATA: START AT EDGE
OPACITY PROJECT RBF DATA: MISSING HYDROGENIC
.
LAMBDA=3
.
ITMAX=30
NEVMAX=4
.
RADIATIVE EQUILIBRIUM: DIFFERENTIAL/INTEGRAL FORM
.INNER BOUNDARY: LAMBDA-ITERATION
.LINEARIZE HYDROSTATIC EQUATION
.SKIP OCCDRVF AFTER 1ST LINEARIZATION
.
SWITCH OFF LINES
.
KANTOROVICH=2, SWITCH LIMITS 0-->1, 1-->2, 2-->1 : 0.1 0.01 0.5
REDUCE LOG CVEC -2.0
.
NO TEMPERATURE CORRECTION
.UNSOELD-LUCY TEMPERATURE CORRECTION DAMP=0.1
.UNSOELD-LUCY PARAMETERS PRINT LIMIT 0.1 TAU-WTS .1 1.
.UNSOELD-LUCY PARAMETERS LIMIT 0.1 TAU-WTS .1 1.
.

```

```

SOLVE STATISTICAL EQUATIONS ONLY RE-SOLVE PARTICLE CONSERVATION
NO NEGATIVE POPULATION NUMBERS (LTE)
.
PRINT OPTIONS
PRINT MODEL ATOMS (OVERVIEW)
PRINT ABUNDANCES
PRINT MAX. REL. CORRECTIONS EVERY 1 ITERATIONS
PRINT INTEGRATED SURFACE FLUX, ITERATION: EACH
PRINT CP-TIME/ITERATION, EACH
PRINT OUTPUT MODEL, ITERATION: LAST, DEPTH INCREMENT: 1 (STRUCTURE ONLY)
.
MACHINE 'hostname'
eos
#
cp ${MI} MODIN
cp ${RUNTMPDIR}/ATOMS_NLTE ATOMS
cp ${RUNTMPDIR}/FGRID_NLTE FGRID
${DA}/get_OP
#
#
#
${BI}/${code}_${modcode}${sys} 2>ERRORS < DATEN
#
if test -s MODOUT
then
cp MODOUT ${MO}
chmod 600 ${MO}
ls -l ${MO}
rm MODOUT
if test -s STOP
then
cp STOP ${MO}.converged
fi
else
cat "no new model ${MO} created" ERRORS > ${MO}.failed
cat "no new model ${MO} created" ERRORS > ${MO}.failed@${HOSTNAME}
echo "no new model ${MO} created"
cat "no new model ${MO} created" ERRORS > ${final_model}.failed
rm ${final_model}.calculating*
# clean up all binaries compiled for this request
# rm ${BI}/*${modcode}*${sys}
exit 3
fi
#
rm *
#
# end of: calculate NLTE continuum model only if not already existing
fi
#
#
#
#####
###
###
### DIE FOLGENDEN SCHRITTE WERDEN ANALOG DURCHGEFUEHRT,
### HIER ABER NICHT GELISTET
###
#####
#
# NLTE line model
# in seven steps
# step 1) REDUCE LOG CVEC -2.0, NO TEMPERATURE CORRECTION
# step 2) REDUCE LOG CVEC -1.0, NO TEMPERATURE CORRECTION
# step 3) NO TEMPERATURE CORRECTION
# step 4) REDUCE LOG CVEC -2.0, TEMPERATURE CORRECTION (UL)
# step 5) REDUCE LOG CVEC -1.0, TEMPERATURE CORRECTION (UL)
# step 6) REDUCE LOG CVEC -0.5, TEMPERATURE CORRECTION (UL)
# step 7) TEMPERATURE CORRECTION (UL)
#
#

```

```

#
##### output for flux ,...
#
cat > DATEN << eos
COMMENT: TMAW calculation of atmospheric structure etc.
.
OCCUPATION PROBABILITY FORMALISM FOR H1
OCCUPATION PROBABILITY FORMALISM FOR HE1
OCCUPATION PROBABILITY FORMALISM FOR HE2
OCCUPATION PROBABILITY FORMALISM FOR C1
OCCUPATION PROBABILITY FORMALISM FOR C2
OCCUPATION PROBABILITY FORMALISM FOR C3
OCCUPATION PROBABILITY FORMALISM FOR C4
OCCUPATION PROBABILITY FORMALISM FOR C5
OCCUPATION PROBABILITY FORMALISM FOR C6
OCCUPATION PROBABILITY FORMALISM FOR N1
OCCUPATION PROBABILITY FORMALISM FOR N2
OCCUPATION PROBABILITY FORMALISM FOR N3
OCCUPATION PROBABILITY FORMALISM FOR N4
OCCUPATION PROBABILITY FORMALISM FOR N5
OCCUPATION PROBABILITY FORMALISM FOR N6
OCCUPATION PROBABILITY FORMALISM FOR N7
OCCUPATION PROBABILITY FORMALISM FOR O1
OCCUPATION PROBABILITY FORMALISM FOR O2
OCCUPATION PROBABILITY FORMALISM FOR O3
OCCUPATION PROBABILITY FORMALISM FOR O4
OCCUPATION PROBABILITY FORMALISM FOR O5
OCCUPATION PROBABILITY FORMALISM FOR O6
OCCUPATION PROBABILITY FORMALISM FOR O7
OCCUPATION PROBABILITY FORMALISM FOR O8
.
OPACITY PROJECT RBF DATA: START AT EDGE
OPACITY PROJECT RBF DATA: MISSING HYDROGENIC
.
LAMBDA=3
.
ITMAX=1
NEWMAX=1
.
RADIATIVE EQUILIBRIUM: DIFFERENTIAL/INTEGRAL FORM
.INNER BOUNDARY: LAMBDA-ITERATION
LINEARIZE HYDROSTATIC EQUATION
SKIP OCCDRVF AFTER 1ST LINEARIZATION
.
DEPTH DEPENDENT LINE PROFILES
.STEP UP F-VALUES: MODEL-START: H1 1.0E-5 1.0 100
.
KANTOROVICH=2, SWITCH LIMITS 0-->1, 1-->2, 2-->1 : 1.0E+33 0.01 0.5
.REDUCE LOG CVEC -2.0
.
NO TEMPERATURE CORRECTION
UNSOELD-LUCY TEMPERATURE CORRECTION DAMP=0.1
UNSOELD-LUCY PARAMETERS LIMIT 0.1 TAU-WTS .1 1.
.
.SOLVE STATISTICAL EQUATIONS ONLY RE-SOLVE PARTICLE CONSERVATION
NO NEGATIVE POPULATION NUMBERS (LTE)
.
PRINT OPTIONS
PRINT MODEL ATOMS (OVERVIEW)
PRINT ABUNDANCES
PRINT MAX. REL. CORRECTIONS EVERY 1 ITERATIONS
PRINT INTEGRATED SURFACE FLUX, ITERATION: EACH
PRINT CP-TIME/ITERATION, EACH
PRINT OCCUPATION PROBABILITIES, ITERATION: LAST, DEPTH INCREMENT: 1
PRINT OUTPUT MODEL, ITERATION: LAST, DEPTH INCREMENT: 1 (STRUCTURE ONLY)
PLOT EMERGENT FLUX, ITERATION: LAST
.
MACHINE 'hostname'
eos
#
cp ${MI} MODIN

```

```

cp ${RUNTMPDIR}/ATOMS_NLTE_LF ATOMS
cp ${RUNTMPDIR}/FGRID_NLTE_LF FGRID
${DA}/get_OP
#
#
#
${BI}/${code}_${modcode2}${sys} 2>ERRORS > OUTPUT_FINAL < DATEN
#
#
# copy data files
#
if test -s PRFLUX
then
cp PRFLUX ${jobdir}/${name}_LF.flux
fi
if test -s STRUCTURE
then
cp STRUCTURE ${jobdir}/${name}_LF.T-structure
fi
#
cp OUTPUT_FINAL ${jobdir}/${name}_LF.TMAW_out
#
#
#####
#
#      ### # # ##### #          ##### ##### ##### #####
#      #  ## # #          ##          #  #  # #  # #
#      #  # # # #          # #          #  # #  # #  # #
#      #  # # # ##### #          ##### ##### #  #####
#      #  # # # # # #          #          #  # #  # #  # #
#      #  # # # # # #          #          #  # #  # #  # #
#      ##### ### # # ##### ##### ##### #  #  # ##### #
#
#####
# synthetic spectra
#
#
if test ! -s ${MI}
then
echo "no input model: ${MI}"
exit 3
fi
#
#
GRP=$HOME/group
OPD=$HOME/data_GRID
#
# 1) user's choice
#
rm FGRID
# base frequency grid
echo "${WAVBEG} ${WAVEND} ${WAVSTP}" > F_BASE
#
cp ${RUNTMPDIR}/ION_USER_LF ION
cp ${RUNTMPDIR}/LINIEN_USER_LF LINIEN
#
cat > DISTA << eos
2 0.01
4 0.02
6 0.03
8 0.05
9 0.10
10 0.20
11 0.30
12 0.50
13 1.00
14 2.00
15 5.00
16 300.00
17 0.01
eos

```



```

#
cat > DATEN << eos
OCCUPATION PROBABILITY FORMALISM FOR H1
OCCUPATION PROBABILITY FORMALISM FOR HE1
OCCUPATION PROBABILITY FORMALISM FOR HE2
OCCUPATION PROBABILITY FORMALISM FOR C1
OCCUPATION PROBABILITY FORMALISM FOR C2
OCCUPATION PROBABILITY FORMALISM FOR C3
OCCUPATION PROBABILITY FORMALISM FOR C4
OCCUPATION PROBABILITY FORMALISM FOR C5
OCCUPATION PROBABILITY FORMALISM FOR C6
OCCUPATION PROBABILITY FORMALISM FOR N1
OCCUPATION PROBABILITY FORMALISM FOR N2
OCCUPATION PROBABILITY FORMALISM FOR N3
OCCUPATION PROBABILITY FORMALISM FOR N4
OCCUPATION PROBABILITY FORMALISM FOR N5
OCCUPATION PROBABILITY FORMALISM FOR N6
OCCUPATION PROBABILITY FORMALISM FOR N7
OCCUPATION PROBABILITY FORMALISM FOR O1
OCCUPATION PROBABILITY FORMALISM FOR O2
OCCUPATION PROBABILITY FORMALISM FOR O3
OCCUPATION PROBABILITY FORMALISM FOR O4
OCCUPATION PROBABILITY FORMALISM FOR O5
OCCUPATION PROBABILITY FORMALISM FOR O6
OCCUPATION PROBABILITY FORMALISM FOR O7
OCCUPATION PROBABILITY FORMALISM FOR O8
.
OPACITY PROJECT RBF DATA: START AT EDGE
OPACITY PROJECT RBF DATA: MISSING HYDROGENIC
.
PRINT ATOMMODELLE(KURZFORM)
PRINT LINIEN UND BLENDS
PLOT EINZELNE LINIEN PLOTTER
PLOT EMERGENTER FLUSS FLAM / LAM, INTERVAL: ${WAVBEG} ${WAVEND}
eos
#
cp ${MI} MODIN
cp ${RUNTMPDIR}/ATOMS_WFP_LF ATOMS
cp ${GRP}/H1 STARK_H1
cp ${GRP}/HE1 STARK_HE1
cp ${GRP}/HE2 STARK_HE2
chmod 600 STARK_*
${DA}/get_OP > /dev/null
#
${BI}/${code2}_${modcode}${sys} < DATEN
#
if test -s IDENT
then
cp IDENT ${jobdir}/${name}_LF_${DATE}.IDENT
fi
if test -s PLWFP
then
cp PLWFP ${jobdir}/${name}_LF_${DATE}.WFP
fi
#
#
# 2) ionizing flux
#
rm FGRID
# base frequency grid
cat > F_BASE << eos
5.000000D+00 2.001000D+03 1.000000D-01
eos
#
echo "H 1" > ION
#
cat > LINIEN << eos
BLENDRANGE 1205
H11 H12 1215 A
eos
#

```

```

cat > DISTA << eos
2 0.01
4 0.02
6 0.03
8 0.05
13 0.10
16 0.20
19 0.30
22 0.50
25 1.00
28 2.00
32 3.00
34 5.00
36 8.00
38 10.00
40 20.00
eos
#
cat > DATEN << eos
OCCUPATION PROBABILITY FORMALISM FOR H1
OCCUPATION PROBABILITY FORMALISM FOR HE1
OCCUPATION PROBABILITY FORMALISM FOR HE2
OCCUPATION PROBABILITY FORMALISM FOR C1
OCCUPATION PROBABILITY FORMALISM FOR C2
OCCUPATION PROBABILITY FORMALISM FOR C3
OCCUPATION PROBABILITY FORMALISM FOR C4
OCCUPATION PROBABILITY FORMALISM FOR C5
OCCUPATION PROBABILITY FORMALISM FOR C6
OCCUPATION PROBABILITY FORMALISM FOR N1
OCCUPATION PROBABILITY FORMALISM FOR N2
OCCUPATION PROBABILITY FORMALISM FOR N3
OCCUPATION PROBABILITY FORMALISM FOR N4
OCCUPATION PROBABILITY FORMALISM FOR N5
OCCUPATION PROBABILITY FORMALISM FOR N6
OCCUPATION PROBABILITY FORMALISM FOR N7
OCCUPATION PROBABILITY FORMALISM FOR O1
OCCUPATION PROBABILITY FORMALISM FOR O2
OCCUPATION PROBABILITY FORMALISM FOR O3
OCCUPATION PROBABILITY FORMALISM FOR O4
OCCUPATION PROBABILITY FORMALISM FOR O5
OCCUPATION PROBABILITY FORMALISM FOR O6
OCCUPATION PROBABILITY FORMALISM FOR O7
OCCUPATION PROBABILITY FORMALISM FOR O8
.
OPACITY PROJECT RBF DATA: START AT EDGE
OPACITY PROJECT RBF DATA: MISSING HYDROGENIC
.
PRINT ATOMMODELLE(KURZFORM)
PRINT LINIEN UND BLENDS
PLOT EINZELNE LINIEN PLOTTER
PLOT EMERGENTER FLUSS FLAM / LAM, INTERVAL: 5.0 2000.1
eos
#
cp ${M1} MODIN
cp ${RUNTMPDIR}/ATOMS_WFP_LF_00005-02000 ATOMS
cp ${GRP}/H1 STARK_H1
cp ${GRP}/HE1 STARK_HE1
cp ${GRP}/HE2 STARK_HE2
chmod 600 STARK_*
${DA}/get_OP > /dev/null
#
${BI}/${code2}_${modcode}_00005-02000${sys} < DATEN
#
if test -s PLWF
then
ln -sf PLWF in
fi
if test -s PLWFP
then
cp PLWFP ${jobdir}/${name}_LF_${DATE}.WFP_00005-02000
fi

```

```

#
# 3) standard near ultraviolet
rm FGRID
#
# base frequency grid
cat > F_BASE << eos
2.00000D03 3.00000D03 1.00000D-01
eos
#
# attention: no lines of H I
#
if test -s ${RUNTMPDIR}/PURE_H
then
#
PURE_H=1
WAVE_H=4000
#
echo "H 1" > ION
#
cat > LINIEN << eos
BLENDRANGE      1800
H12             H110             3798.9 A
eos
#
else
#
PURE_H=0
WAVE_H=3000
#
echo "HE2" > ION
#
cat > LINIEN << eos
BLENDRANGE      520
HE23            HE27            2511.9 A
eos
#
fi
#
cat > DISTA << eos
2  0.01
4  0.02
6  0.03
8  0.05
13 0.10
16 0.20
19 0.30
22 0.50
25 1.00
28 2.00
32 3.00
34 5.00
36 8.00
38 10.00
40 20.00
42 50.00
44 80.00
45 100.00
eos
#
cat > DATEN << eos
OCCUPATION PROBABILITY FORMALISM FOR H1
OCCUPATION PROBABILITY FORMALISM FOR HE1
OCCUPATION PROBABILITY FORMALISM FOR HE2
OCCUPATION PROBABILITY FORMALISM FOR C1
OCCUPATION PROBABILITY FORMALISM FOR C2
OCCUPATION PROBABILITY FORMALISM FOR C3
OCCUPATION PROBABILITY FORMALISM FOR C4
OCCUPATION PROBABILITY FORMALISM FOR C5
OCCUPATION PROBABILITY FORMALISM FOR C6
OCCUPATION PROBABILITY FORMALISM FOR N1
OCCUPATION PROBABILITY FORMALISM FOR N2

```

```

OCCUPATION PROBABILITY FORMALISM FOR N3
OCCUPATION PROBABILITY FORMALISM FOR N4
OCCUPATION PROBABILITY FORMALISM FOR N5
OCCUPATION PROBABILITY FORMALISM FOR N6
OCCUPATION PROBABILITY FORMALISM FOR N7
OCCUPATION PROBABILITY FORMALISM FOR O1
OCCUPATION PROBABILITY FORMALISM FOR O2
OCCUPATION PROBABILITY FORMALISM FOR O3
OCCUPATION PROBABILITY FORMALISM FOR O4
OCCUPATION PROBABILITY FORMALISM FOR O5
OCCUPATION PROBABILITY FORMALISM FOR O6
OCCUPATION PROBABILITY FORMALISM FOR O7
OCCUPATION PROBABILITY FORMALISM FOR O8
.
OPACITY PROJECT RBF DATA: START AT EDGE
OPACITY PROJECT RBF DATA: MISSING HYDROGENIC
.
PRINT ATOMMODELLE(KURZFORM)
PRINT LINIEN UND BLENDS
PLOT EINZELNE LINIEN PLOTTER
PLOT EMERGENTER FLUSS FLAM / LAM, INTERVAL: 2000 3000.1
eos
#
cp ${MI} MODIN
cp ${RUNTMPDIR}/ATOMS_WFP_LF_02000-0${WAVE_H} ATOMS
cp ${GRP}/H1 STARK_H1
cp ${GRP}/HE1 STARK_HE1
cp ${GRP}/HE2 STARK_HE2
chmod 600 STARK_*
${DA}/get_OP > /dev/null
#
${BI}/${code2}_${modcode}_02000-0${WAVE_H}${sys} < DATEN
#
if test -s PLWFP
then
  cp PLWFP ${jobdir}/${name}_LF_${DATE}.WFP_02000-03000
fi
#
# 4) standard optical - infrared
rm FGRID
#
# base frequency grid
cat > CONTS_MAN << eos
  3000.87493
  55014.99983
eos
cat > F_BASE << eos
  2.99950D03 7.00000D03 1.00000D+00
  7.00500D03 5.50005D04 5.00000D+00
eos
#
echo "H 1" > ION
#
cat > LINIEN << eos
BLENDRANGE 27000
H15 H110 30392 A
eos
#
cat > DISTA << eos
  2 0.01
  4 0.02
  6 0.03
  8 0.05
 13 0.10
 16 0.20
 19 0.30
 22 0.50
 25 1.00
 26 25000.00
 37 1.00
eos

```



```

then
  if [ ${WTCK} -lt 10 ]
  then
    WTCK=10
    WLAB=100
  else
    WTCK=100
    WLAB=1000
  fi
else
  WTCK=200
  WLAB=2000
fi
else
  WTCK=500
  WLAB=5000
fi
#
#
cat > ${name}_LF_${DATE}_flux.plot <<EOF
PAPERFORMAT A4Q
PLOT :
\OFS 2.5 2.5
\FONT=HELVET
\LETTERSIZ=0.2
\NOCOPYRIGHT
\INBOX
\LATEBOX
\PAUSE
\COLOR=1
\DATE
\FILE
HEADER :\CENTER\ TMAW: ${name}_LF_${DATE} flux
X-ACHSE:\CENTER\ wavelength / \A
Y-ACHSE:\CENTER\ log &RF&N&T#l#&M / erg cm&H-2&M sec&H-1&M cm&H-1&M
MASSTAB MINIMUM MAXIMUM TEILUNGEN BESCHRIFT. DARUNTER
X: AUTO 25CM ${WAVBEG} ${WAVEND} ${WTCK} ${WLAB} 0.
Y: 16CM 14.4 19.2 .1 1. 0.
N= ? PLOTSYMBOL 5 PEN 2 XYTABLE SELECT 1 2 COLOR=2
COMMAND Y-CUT 1.0E-33 1.E38
COMMAND YLOG
COMMAND INCLUDE "${name}_LF_${DATE}.WFP"
FINISH
\PEN=3
END
EOF
chmod 700 ${name}_LF_${DATE}_flux.plot
chmod 700 ${name}_LF_${DATE}.WFP
${HOME}/wrplot.dir/proc.dir/wrplot.com ${name}_LF_${DATE}_flux.plot +
mv wrplot.ps.1 ${name}_LF_${DATE}_flux.ps
# Making plotfile and plotting temperature structure
#
cat > ${name}_LF_${DATE}_T-structure.plot <<EOF
PAPERFORMAT A4Q
PLOT :
\OFS 2.5 2.5
\FONT=HELVET
\LETTERSIZ=0.2
\NOCOPYRIGHT
\INBOX
\LATEBOX
\PAUSE
\DATE
\PEN=3
\COLOR=1
HEADER :\CENTER\ TMAW: ${name}_LF T structure
X-ACHSE:\CENTER\ log &Fm&N / g cm&H-2&M
Y-ACHSE:\CENTER\ &RT&N / kK
MASSTAB MINIMUM MAXIMUM TEILUNGEN BESCHRIFT. DARUNTER
X: 25CM -7.9 4.3 1. 1. 0.
Y: 16CM 0. 999. 100. 200. 0.

```

```

N= ? PLOTSYMBOL 5 PEN 3 XYTABLE SELECT 1 2 COLOR=2
COMMAND INCLUDE "${name}_LF.T-structure"
FINISH
\PEN=3
END
EOF

#
chmod 700 ${name}_LF_${DATE}_T-structure.plot
chmod 700 ${name}_LF.T-structure
${HOME}/wrplot.dir/proc.dir/wrplot.com ${name}_LF_${DATE}_T-structure.plot +
mv wrplot.ps.1 ${name}_LF_${DATE}_T-structure.ps
#
#####
#
# ##### ## #####
# # # # # #
# # # # # #
# # ##### #####
# # # # # #
# # # # # #
#
#####
#
USERINTERVAL=${WAVBEG}-${WAVEND}
WFPDIR=${NEWHOME}/WFP
cp ${name}_LF_${DATE}.WFP_????-???? ${WFPDIR} | chmod 644 ${WFPDIR}/${name}_LF_${DATE}.\
WFP_????-????
if [ ${FLUXINTERVAL} == "UV" ]
then
cp ${name}_LF.flux ${name}_LF_${DATE}.flux
cp ${name}_LF.TMAW_out ${name}_LF_${DATE}.TMAW_out
tar czf ${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}.tgz README ${name}_LF_${DATE}.\
TMAW_out ${name}_LF_${DATE}.flux ${name}_LF.IDENT_${DATE} ${name}_LF_${DATE}.WFP ${\
name}_LF_${DATE}.WFP_0005-02000 ${name}_LF_*.ps && mv ${name}_LF_${FLUXINTERVAL}_${\
USERINTERVAL}.tgz ${NEWHOME}/results && rm ${name}_LF*
fi
if [ ${FLUXINTERVAL} == "NUV" ]
then
cp ${name}_LF.flux ${name}_LF_${DATE}.flux
cp ${name}_LF.TMAW_out ${name}_LF_${DATE}.TMAW_out

tar czf ${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}.tgz README ${name}_LF_${DATE}.\
TMAW_out ${name}_LF_${DATE}.flux ${name}_LF.IDENT_${DATE} ${name}_LF_${DATE}.WFP ${\
name}_LF_${DATE}.WFP_02000-03000 ${name}_LF_*.ps && mv ${name}_LF_${FLUXINTERVAL}_${\
USERINTERVAL}.tgz ${NEWHOME}/results/ && rm ${name}_LF*
fi
if [ ${FLUXINTERVAL} == "OPTIR" ]
then
cp ${name}_LF.flux ${name}_LF_${DATE}.flux
cp ${name}_LF.TMAW_out ${name}_LF_${DATE}.TMAW_out

tar czf ${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}.tgz README ${name}_LF_${DATE}.\
TMAW_out ${name}_LF_${DATE}.flux ${name}_LF.IDENT_${DATE} ${name}_LF_${DATE}.WFP ${\
name}_LF_${DATE}.WFP_03000-55000 ${name}_LF_*.ps && mv ${name}_LF_${FLUXINTERVAL}_${\
USERINTERVAL}.tgz ${NEWHOME}/results/ && rm ${name}_LF*
fi
#
cp ${name}_LF_${FLUXINTERVAL}_${USERINTERVAL}.tgz ${NEWHOME}/results/
#
#####

```

Listing A.33: Die scp2li04.bat-Prozedur wurde erstellt, um die tar-Archiv vom IAAT zum AstroGrid-D @IAAT-Server zu transferieren und dort die Berechnung mit der tmap\_grid.pl Prozedur zu starten. Diese Prozedur war schon zu Beginn dieser Arbeit vorhanden.

```
#!/bin/sh
#####
# This script copies the .tgz files from TMAW to tmaw,
# creates input and output directories,
# and starts calculating the jobs on AstroGrid-D.
#
# LAST_MODIFICATION="25-Jan-2013 08:39:38"
#
# Copyright 2006-2013 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
#
# the script is started by a cron-script now
# first check if there are tarballs waiting for the calculation on AGD
# (if yes, then a file called waiting exists in which the date of
# the request is written)
# check also that the input and output directories do not exist
# (that it has not already been calculated)
#
#
rm /home/tmaw/TMAP/waiting
scp TMAW@aitp19:./GRID/waiting /home/tmaw/TMAP/
if test ! -s /home/tmaw/TMAP/waiting
then
exit
fi
#
ReqName='cat /home/tmaw/TMAP/waiting | head -1'
if test -s /home/tmaw/TMAP/${ReqName}.in
then
ls -ltr /home/tmaw/TMAP/${ReqName}.in
exit
fi
#
#
echo ${ReqName} > /home/tmaw/TMAP/started
#
cat > MAIL << EOF
Dear TMAW Administrator ,

the requests have been sent to AstroGrid-D and the calculation started.

Yours
TMAW Team
EOF
#

/usr/bin/mail -r "AGD" -s "AGD:calculation started" \
ringat@astro.uni-tuebingen.de < MAIL
rm MAIL
# define the new input and output directory names
#
Gridinputdir=${ReqName}'.in'
Gridoutputdir=${ReqName}'.out'
#
# create the input directory and set the correct permit
#
cd ~/TMAP/
```



```

ls -ltr ${Gridinputdir}
if test ! -s ${Gridinputdir}
then
    mkdir      ${Gridinputdir}
    chmod 750  ${Gridinputdir}
    chgrp tmap ${Gridinputdir}
fi
#
# go to the input directory and copy the request file
# (including the atomic data and frequency grid files) to this directory
# and set the correct permit
#
cd ${Gridinputdir}
#
scp TMAW@aitp20:"GRID/*${ReqName}.tgz" .
chmod 640 *.tgz
chgrp tmap *.tgz
#
cd ~/TMAP/
#
# now create the output directory
#
if test ! -s ${Gridoutputdir}
then
    mkdir      ${Gridoutputdir}
    chmod 770  ${Gridoutputdir}
    chgrp tmap ${Gridoutputdir}
fi
#
# start the perl script that logs in on the AstroGrid-D machine,
# unpacks the tarball, and starts the calculation of the model atmosphere
#
tmap_grid.pl --status ${Gridoutputdir} `find /home/tmaw/TMAP/${Gridinputdir}/ \
-name "req_grid*.tgz" `
#####

```

Listing A.34: Die scp2IAAT.bat-Prozedur wurde erstellt, um die tar-Archiv nach der Berechnung der Modelle vom AstroGrid-D @IAAT-Server zum IAAT zu transferieren.

```
#!/bin/sh
#####
# This script copies the files calculated on
# AstroGrid-D back to IAAT machine.
#
# LAST_MODIFICATION="25-Jan-2013 08:40:42"
#
# Copyright 2006-2013 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
# the script is used by a cron job now
# when a calculation is running, a file /home/tmaw/TMAP/started exists
#
if test ! -s /home/tmaw/TMAP/started
then
exit
fi
# check if all files have produced an output by now
ReqName='cat /home/tmaw/TMAP/started | head -1'
runin='ls -ltr /home/tmaw/TMAP/${ReqName}.in | wc | cut -c1-10'
runout='ls -ltr /home/tmaw/TMAP/${ReqName}.out | wc | cut -c1-10'
if [ $runout -lt $runin ]
then
exit
fi
#
cat > MAIL << EOF
Dear TMAW Administrator ,

the calculation on AstroGrid-D finished.
The files are transfered to IAAT now.

Yours
TMAW Team
EOF
#

/usr/bin/mail -r "AGD" -s "AGD: calculation finished" \
astro-tmaw@listserv.uni-tuebingen.de < MAIL
rm MAIL
# go to the output directory and copy the tarballs to IAAT machines
#
Gridoutputdir=${ReqName}'.out'
cd /home/tmaw/TMAP/${Gridoutputdir}
chgrp tmap *tgz
scp *.tgz TMAW@aitp19:./GRID_results/
cd ~/TMAP
#
# copy only once:
rm /home/tmaw/TMAP/started
#####
```

Listing A.35: Die checkgridH.bat-Prozedur wird einmal pro Woche automatisch ausgeführt. Es berechnet ein kleines Testgitter auf den über das AstroGrid-D erreichbaren Rechnern, um zu testen, ob die Berechnung dauerhaft mit gleichen Ergebnissen funktioniert.

```
#!/bin/bash
#####
# This script is used in order to check if jobs are
# calculated correctly on AstroGrid-D
# Therefore the grid is calculated every Friday.
#
# LAST_MODIFICATION="25-Jan-2013 09:32:46"
#
# Copyright 2010-2013 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
# write an email notification that the calculation of the testgrid started
# and start the calculation with the perl script
#
    echo "The testgrid run on AstroGrid-D started, the jobs are send to AstroGrid-D" > MAIL
    /usr/bin/mail -r "TMAW" -s "TMAW INFO: Testgrid run on AstroGrid-D started" \
    astro-tmaw@listserv.uni-tuebingen.de < MAIL
    rm MAIL
echo "jobs started on 'hostname' at 'date +%Y-%m-%d_%H:%M:%S'" >> \
~/TMAP/testgridH.out/logfileH
cd ~/TMAP/
./tmap_grid.pl --status testgridH.out 'find /home/tmaw/TMAP/testgridH.in/ -name \
"req_grid*.tgz"'
# >> testgridH.out/testgridout
#
# after the calculation is done, notify the administrator about the end of the calculation
#
echo "jobs ended on 'hostname' at 'date +%Y-%m-%d_%H:%M:%S'" >> \
~/TMAP/testgridH.out/logfileH
    echo "The testgrid run on AstroGrid-D ended on tmaw. The files were calculated on AstroGrid\
-D." \
> MAIL
    /usr/bin/mail -r "TMAW" -s "TMAW INFO: Testgrid run on AstroGrid-D ended on tmaw" \
astro-tmaw@listserv.uni-tuebingen.de < MAIL
    rm MAIL
#
# go to the output directory and check the differences of the actual calculation
# and a previous calculation
# if there are diffences, notify the administrator
#
cd /home/tmaw/TMAP/testgridH.out/
rm /home/tmaw/TMAP/testgridH.out/failed
cat > Re << eos
while [ \ $# -gt 0 ]
do
echo \${1}
/home/tmaw/TMAP/checkgridH2.bat \${1}
shift
done
eos
chmod u=u+x Re
./Re 'find /home/tmaw/TMAP/testgridH.out/ -name "req_grid*_2011-01-03_17_49_01.tgz" -print | \
\
cut -c31-74'
rm Re

if test -s /home/tmaw/TMAP/testgridH.out/failed
```

```
then
  echo "The testgrid run on AstroGrid-D was not successful." > MAIL
  /usr/bin/mail -r "TMAW" -s "TMAW INFO: Testgrid run on AstroGrid-D was not successful" \
  astro-tmaw@listserv.uni-tuebingen.de < MAIL
  rm MAIL
else
  echo "The testgrid run on AstroGrid-D was successful." > MAIL
  /usr/bin/mail -r "TMAW" -s "TMAW INFO: Testgrid run on AstroGrid-D was successful" \
  astro-tmaw@listserv.uni-tuebingen.de < MAIL
  rm MAIL
fi
#####
```

Listing A.36: Die checkgridH2.bat-Prozedur wird von der checkgridH.bat-Prozedur verwendet.

```
#!/bin/bash
#####
# This script is used by checkgridH.bat
#
# LAST_MODIFICATION="25-Jan-2013 09:35:59"
#
# Copyright 2010-2013 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
#
filepart='echo ${1} | cut -c13-14'
rm -rf /home/tmaw/TMAP/testgridH.out/TMAP/
tar -xvf ${1}
cd /home/tmaw/TMAP/testgridH.out/TMAP
counthelp='ls -l results |cut -d ' ' -f 2'
if [ ${counthelp} -eq 0 ]
then
    echo "failed, no result files were created" > /home/tmaw/TMAP/testgridH.out/failed
    exit
fi
cd results
tar -xvf *.tgz
diff /home/tmaw/TMAP/testgridH.out/WFP_${1} \
/home/tmaw/TMAP/testgridH.out/TMAP/results/*.WFP > /home/tmaw/TMAP/testgridH.out/diffWFP
cd /home/tmaw/TMAP/testgridH.out/
count='wc -l /home/tmaw/TMAP/testgridH.out/diffWFP |cut -d ' ' -f1'
if [ ${count} -gt 6 ]
then
    echo "failed, wrong number of lines in WFP file" > /home/tmaw/TMAP/testgridH.out/failed
fi

cd /home/tmaw/TMAP/testgridH.out/
#####
```

Listing A.37: Die checkquality.bat-Prozedur kann für die Überprüfung der Modelle eines Gitters auf Konvergenz, Abbruch oder erreichte Temperatur verwendet werden.

```
#!/bin/bash
#####
# This script is used in order to check the convergence state
# of jobs calculated on AstroGrid-D.
#
# LAST_MODIFICATION="25-Jan-2013 09:38:12"
#
# Copyright 2010-2013 by TMAW team
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#
# the use of this procedure is free
# any change is strictly prohibited
#####
# first enter the date of the request that has to be checked
FileName=2013-01-02_19_58_43
cd ${FileName}'.out'
#
# unpack the tarballs and check if one model failed or converged
# grep the last effective temperature and write everything to a file
#
tar -xvf req_grid_00001_${FileName}.tgz
#
echo 'head -2 TMAP/req_grid_00020_${FileName}' > help
firstpara='head -1 help'
rm help
#
cat > Re << \eos
  while [ $# -gt 0 ]
  do
ReqName=$1
tar -xvf ${ReqName}.tgz
cd TMAP
echo 'head -2 ${ReqName}' > ../help
para='head -1 ../help'
rm ../help
#
if test -s models/TMAW/*WWW.converged
then
echo "$para c ${ReqName}" >> ../inputfile
fi
if test -s models/TMAW/*WWW.failed
then
echo "$para f ${ReqName}" >> ../inputfile
fi
if test ! -s models/TMAW/*WWW.failed -a ! -s models/TMAW/*WWW.converged
then
echo "$para n ${ReqName}" >> ../inputfile
fi
#
awk '/integr/' OUTPUT_2* > ../STDOUT_T_help
awk '/integr/ {print $12}' OUTPUT_2* >> ../STDOUT_T_help
echo $para >> ../STDOUT_T
echo 'tail -1 ../STDOUT_T_help' >> ../STDOUT_T
rm ../STDOUT_T_help
cd ..
rm -rf ~/TMAP/${FileName}'.out' /TMAP/
shift
done
eos
chmod u+ux Re
./Re 'find /home/tmaw/TMAP/${FileName}.out/ -name "req_grid_*.tgz" -print | cut -c41-74'
#####
```

## D. TEUV-Programme und Skripte

Der neue *TEUV*-Dienst wird mit Hilfe von PHP- und eines Unix-Shell-Skripts gesteuert. Grundlage dieses Dienstes ist das FORTRAN-Programm *TEUV.f90* bzw. *TEUV1.f90*. Sie wurden alle im Rahmen dieser Arbeit erstellt. Der Stand der Programme und des Unix-Shell-Skripts am 25.01.2013 wird im Folgenden gelistet.

Listing A.38: Das dem *TEUV*-Dienst zugrunde liegende FORTRAN-Programm *TEUV.f90* bringt eine Korrektur durch interstellare Absorption an das vom Nutzer hochgeladene Spektrum an. Die Subroutinen *gaussd*, *gaussf* und *convolg* existierten bereits vor dieser Arbeit und wurden modifiziert.

```
PROGRAM TEUV
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! This program applies interstellar absorption to
! theoretical stellar fluxes.
!
! LAST_MODIFICATION="25-Jan-2013 09:39:23"
!
! Copyright 2012 - 2013 by E. Mueller-Ringat
!
! Institute for Astronomy und Astrophysics
! Kepler Center for Astro and Particle Physics
! Eberhard Karls University
! Sand 1
! D-72076 Tuebingen
! Germany
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
  implicit none

  integer nx, i, j, imax, imaxheii, imaxhi, ilambda, icool, iwarm
  integer imaxci, imaxcii, imaxciii, imaxni, imaxnii, imaxniii
  integer imaxoi, imaxoii, imaxoiii, nx24

  parameter nx=200000
  parameter nx24=4800000

  character (len=190) filename, date_e, filenameout, firstname, lastname, IP
  character (len=90) head
  character (len=35) filename_op(12)

  real lambda(nx), lambda_cool(nx), lambda_warm(nx)
  real flux(nx), ismflux(nx), ismfluxcorr(nx)
  real v_rad_cool, v_turb_cool, T_cool
  real v_rad_warm, v_turb_warm, T_warm
  real resolution, c, edge(12)
  real tau_int_cool(nx), tau_cool(nx)
  real tau_int_warm(nx), tau_warm(nx)
  real Hcol_cool, HEIcol_cool, HEIIcol_cool
  real CIcol_cool, CIIcol_cool, CIIIcol_cool
  real NIcol_cool, NIIcol_cool, NIIIcol_cool
  real OIcol_cool, OIIcol_cool, OIIIcol_cool
  real Hcol_warm, HEIcol_warm, HEIIcol_warm
  real CIcol_warm, CIIcol_warm, CIIIcol_warm
  real NIcol_warm, NIIcol_warm, NIIIcol_warm
  real OIcol_warm, OIIcol_warm, OIIIcol_warm
  real cs_hi_cool(nx), cs_hi_warm(nx)
  real cs_hei_cool(nx), cs_hei_warm(nx)
  real cs_heii_cool(nx), cs_heii_warm(nx)
  real cs_ci_cool(nx), cs_ci_warm(nx)
  real cs_cii_cool(nx), cs_cii_warm(nx)
  real cs_ciii_cool(nx), cs_ciii_warm(nx)
  real cs_ni_cool(nx), cs_ni_warm(nx)
  real cs_nii_cool(nx), cs_nii_warm(nx)
  real cs_niii_cool(nx), cs_niii_warm(nx)
  real cs_oi_cool(nx), cs_oi_warm(nx)
  real cs_oii_cool(nx), cs_oii_warm(nx)
  real cs_oiii_cool(nx), cs_oiii_warm(nx)
```

```

real lambda_op_hi(nx)
real lambda_op_hei(nx)
real lambda_op_heii(nx)
real lambda_op_ci(nx)
real lambda_op_cii(nx)
real lambda_op_ciii(nx)
real lambda_op_ni(nx)
real lambda_op_nii(nx)
real lambda_op_niii(nx)
real lambda_op_oi(nx)
real lambda_op_oii(nx)
real lambda_op_oiii(nx)
real op_cs_lambda_hi_cool(nx), op_cs_lambda_hi_warm(nx)
real op_cs_lambda_hei_cool(nx), op_cs_lambda_hei_warm(nx)
real op_cs_lambda_heii_cool(nx), op_cs_lambda_heii_warm(nx)
real op_cs_lambda_ci_cool(nx), op_cs_lambda_ci_warm(nx)
real op_cs_lambda_cii_cool(nx), op_cs_lambda_cii_warm(nx)
real op_cs_lambda_ciii_cool(nx), op_cs_lambda_ciii_warm(nx)
real op_cs_lambda_ni_cool(nx), op_cs_lambda_ni_warm(nx)
real op_cs_lambda_nii_cool(nx), op_cs_lambda_nii_warm(nx)
real op_cs_lambda_niii_cool(nx), op_cs_lambda_niii_warm(nx)
real op_cs_lambda_oi_cool(nx), op_cs_lambda_oi_warm(nx)
real op_cs_lambda_oii_cool(nx), op_cs_lambda_oii_warm(nx)
real op_cs_lambda_oiii_cool(nx), op_cs_lambda_oiii_warm(nx)
real rvfactcool, rvfactwarm

data c/2.99792458E+10/

data filename_op/" /home/TEUV/ public_html/EUV/HI_OP ", &
"/home/TEUV/ public_html/EUV/Hel_OP ", &
"/home/TEUV/ public_html/EUV/Hell_OP ", &
"/home/TEUV/ public_html/EUV/CI_OP ", &
"/home/TEUV/ public_html/EUV/CII_OP ", &
"/home/TEUV/ public_html/EUV/CIII_OP ", &
"/home/TEUV/ public_html/EUV/NI_OP ", &
"/home/TEUV/ public_html/EUV/NII_OP ", &
"/home/TEUV/ public_html/EUV/NIII_OP ", &
"/home/TEUV/ public_html/EUV/OI_OP ", &
"/home/TEUV/ public_html/EUV/OII_OP ", &
"/home/TEUV/ public_html/EUV/OIII_OP "/"

data edge/911.75, &
504.26,227.32, &
1101.07,508.48,258.91, &
853.06,418.85,261.30, &
910.44,353.06,225.69/

! *** initialize variables

data tau_cool/nx*0.0/, tau_warm/nx*0.0/

data op_cs_lambda_hi_cool, op_cs_lambda_hi_warm, &
op_cs_lambda_hei_cool, op_cs_lambda_hei_warm, &
op_cs_lambda_heii_cool, op_cs_lambda_heii_warm, &
op_cs_lambda_ci_cool, op_cs_lambda_ci_warm, &
op_cs_lambda_cii_cool, op_cs_lambda_cii_warm, &
op_cs_lambda_ciii_cool, op_cs_lambda_ciii_warm, &
op_cs_lambda_ni_cool, op_cs_lambda_ni_warm, &
op_cs_lambda_nii_cool, op_cs_lambda_nii_warm, &
op_cs_lambda_niii_cool, op_cs_lambda_niii_warm, &
op_cs_lambda_oi_cool, op_cs_lambda_oi_warm, &
op_cs_lambda_oii_cool, op_cs_lambda_oii_warm, &
op_cs_lambda_oiii_cool, op_cs_lambda_oiii_warm/nx24*0.0/

data Hcol_warm, Helcol_warm, Hellcol_warm, &
Clcol_warm, CIIcol_warm, CIIIcol_warm, &
Nicol_warm, NIIcol_warm, NIIIcol_warm, &
Olcol_warm, OIIcol_warm, OIIIcol_warm, &
Hcol_cool, Helcol_cool, Hellcol_cool, &
Clcol_cool, CIIcol_cool, CIIIcol_cool, &
Nicol_cool, NIIcol_cool, NIIIcol_cool, &

```



```

        Olcol_cool, Ollcol_cool, Olllcol_cool/24*0.0/

! *** read input parameters

    read(*,'(A)') firstname
    read(*,'(A)') lastname
    read(*,'(A)') IP

! *** cool component

    read(*,*) v_turb_cool
    read(*,*) v_rad_cool
    read(*,*) T_cool
    read(*,*) resolution
    read(*,*) Hcol_cool
    read(*,*) Helcol_cool
    read(*,*) Hellcol_cool
    read(*,*) Clcol_cool
    read(*,*) Clicol_cool
    read(*,*) Clllcol_cool
    read(*,*) Nlcol_cool
    read(*,*) Nllcol_cool
    read(*,*) Nlllcol_cool
    read(*,*) Olcol_cool
    read(*,*) Ollcol_cool
    read(*,*) Olllcol_cool

! *** warm component

    read(*,*) v_turb_warm
    read(*,*) v_rad_warm
    read(*,*) T_warm
    read(*,*) Hcol_warm
    read(*,*) Helcol_warm
    read(*,*) Hellcol_warm
    read(*,*) Clcol_warm
    read(*,*) Clicol_warm
    read(*,*) Clllcol_warm
    read(*,*) Nlcol_warm
    read(*,*) Nllcol_warm
    read(*,*) Nlllcol_warm
    read(*,*) Olcol_warm
    read(*,*) Ollcol_warm
    read(*,*) Olllcol_warm

    read(*,*) date_e

! *** open the input and output files and read wavelengths and fluxes

    filename = '/home/TEUV/EUVreq/' // trim(date_e) // '/mod'
    filenameout = '/home/TEUV/EUVreq/TEUV.txt'

    open(2, file=filename, &
         form='formatted', &
         position='rewind', &
         status='old', &
         err=97)

    do i=1,70
        read(2,*) head
        if (head(1:1).ne."*") goto 2
    enddo
2    continue
    backspace 2

    do i=1,nx
        read(2,*,end=1000) lambda(i),flux(i)
    enddo

    write(*,('maximum number of wavelength points (" ,18,")exceeded')) nx
    stop '%PROG-TEUV, maximum number of wavelength points exceeded'

```

1000 continue

ilambda=i-1

*! \*\*\* read the Opacity Project RBF data files*  
*! \*\*\* apply Doppler broadening to account for Tcloud and vturb*  
*! \*\*\* interpolate between OP and input wavelengths*

```
if ((Hcol_warm+Hcol_cool).gt.0.0) then
  call readOP (filename_op(1),edge(1),T_cool,v_turb_cool,T_warm,v_turb_warm, &
              cs_hi_cool,cs_hi_warm,lambda_op_hi,imaxhi)
  call interpol (lambda,lambda_op_hi,edge(1),op_cs_lambda_hi_cool,cs_hi_cool, &
                ilambda,imaxhi)
  call interpol (lambda,lambda_op_hi,edge(1),op_cs_lambda_hi_warm,cs_hi_warm, &
                ilambda,imaxhi)
endif
```

```
if ((Helcol_warm+Helcol_cool).gt.0.0) then
  call readOP (filename_op(2),edge(2),T_cool,v_turb_cool,T_warm,v_turb_warm, &
              cs_hei_cool,cs_hei_warm,lambda_op_hei,imax)
  call interpol (lambda,lambda_op_hei,edge(2),op_cs_lambda_hei_cool,cs_hei_cool, &
                ilambda,imax)
  call interpol (lambda,lambda_op_hei,edge(2),op_cs_lambda_hei_warm,cs_hei_warm, &
                ilambda,imax)
endif
```

```
if ((Hellcol_warm+Hellcol_cool).gt.0.0) then
  call readOP (filename_op(3),edge(3),T_cool,v_turb_cool,T_warm,v_turb_warm, &
              cs_heii_cool,cs_heii_warm,lambda_op_heii,imaxheii)
  call interpol (lambda,lambda_op_heii,edge(3),op_cs_lambda_heii_cool,cs_heii_cool, &
                ilambda,imaxheii)
  call interpol (lambda,lambda_op_heii,edge(3),op_cs_lambda_heii_warm,cs_heii_warm, &
                ilambda,imaxheii)
endif
```

```
if ((Clcol_warm+Clcol_cool).gt.0.0) then
  call readOP (filename_op(4),edge(4),T_cool,v_turb_cool,T_warm,v_turb_warm, &
              cs_ci_cool,cs_ci_warm,lambda_op_ci,imaxci)
  call interpol (lambda,lambda_op_ci,edge(4),op_cs_lambda_ci_cool,cs_ci_cool,ilambda, &
                imaxci)
  call interpol (lambda,lambda_op_ci,edge(4),op_cs_lambda_ci_warm,cs_ci_warm,ilambda, &
                imaxci)
endif
```

```
if ((CIIcol_warm+CIIcol_cool).gt.0.0) then
  call readOP (filename_op(5),edge(5),T_cool,v_turb_cool,T_warm,v_turb_warm, &
              cs_cii_cool,cs_cii_warm,lambda_op_cii,imaxcii)
  call interpol (lambda,lambda_op_cii,edge(5),op_cs_lambda_cii_cool,cs_cii_cool, &
                ilambda,imaxcii)
  call interpol (lambda,lambda_op_cii,edge(5),op_cs_lambda_cii_warm,cs_cii_warm, &
                ilambda,imaxcii)
endif
```

```
if ((CIIIcol_warm+CIIIcol_cool).gt.0.0) then
  call readOP (filename_op(6),edge(6),T_cool,v_turb_cool,T_warm,v_turb_warm, &
              cs_ciii_cool,cs_ciii_warm,lambda_op_ciii,imaxciii)
  call interpol (lambda,lambda_op_ciii,edge(6),op_cs_lambda_ciii_cool,cs_ciii_cool, &
                ilambda,imaxciii)
  call interpol (lambda,lambda_op_ciii,edge(6),op_cs_lambda_ciii_warm,cs_ciii_warm, &
                ilambda,imaxciii)
endif
```

```
if ((NIcol_warm+NIcol_cool).gt.0.0) then
  call readOP (filename_op(7),edge(7),T_cool,v_turb_cool,T_warm,v_turb_warm, &
              cs_ni_cool,cs_ni_warm,lambda_op_ni,imaxni)
  call interpol (lambda,lambda_op_ni,edge(7),op_cs_lambda_ni_cool,cs_ni_cool,ilambda, &
                imaxni)
  call interpol (lambda,lambda_op_ni,edge(7),op_cs_lambda_ni_warm,cs_ni_warm,ilambda, &
                imaxni)
endif
```

```

if ((Nllcol_warm+Nllcol_cool).gt.0.0) then
  call readOP (filename_op(8),edge(8),T_cool,v_turb_cool,T_warm,v_turb_warm,      &
             cs_nii_cool,cs_nii_warm,lambda_op_nii,imaxnii)
  call interpol(lambda,lambda_op_nii,edge(8),op_cs_lambda_nii_cool,cs_nii_cool,      &
             ilambda,imaxnii)
  call interpol(lambda,lambda_op_nii,edge(8),op_cs_lambda_nii_warm,cs_nii_warm,      &
             ilambda,imaxnii)
endif

if ((Nlllcol_warm+Nlllcol_cool).gt.0.0) then
  call readOP (filename_op(9),edge(9),T_cool,v_turb_cool,T_warm,v_turb_warm,      &
             cs_nlii_cool,cs_nlii_warm,lambda_op_nlii,imaxnlii)
  call interpol(lambda,lambda_op_nlii,edge(9),op_cs_lambda_nlii_cool,cs_nlii_cool,      &
             ilambda,imaxnlii)
  call interpol(lambda,lambda_op_nlii,edge(9),op_cs_lambda_nlii_warm,cs_nlii_warm,      &
             ilambda,imaxnlii)
endif

if ((Ollcol_warm+Ollcol_cool).gt.0.0) then
  call readOP (filename_op(10),edge(10),T_cool,v_turb_cool,T_warm,v_turb_warm,      &
             cs_oi_cool,cs_oi_warm,lambda_op_oi,imaxoi)
  call interpol(lambda,lambda_op_oi,edge(10),op_cs_lambda_oi_cool,cs_oi_cool,ilambda,      &
             imaxoi)
  call interpol(lambda,lambda_op_oi,edge(10),op_cs_lambda_oi_warm,cs_oi_warm,ilambda,      &
             imaxoi)
endif

if ((Olllcol_warm+Olllcol_cool).gt.0.0) then
  call readOP (filename_op(11),edge(11),T_cool,v_turb_cool,T_warm,v_turb_warm,      &
             cs_oii_cool,cs_oii_warm,lambda_op_oii,imaxoii)
  call interpol(lambda,lambda_op_oii,edge(11),op_cs_lambda_oii_cool,cs_oii_cool,      &
             ilambda,imaxoii)
  call interpol(lambda,lambda_op_oii,edge(11),op_cs_lambda_oii_warm,cs_oii_warm,      &
             ilambda,imaxoii)
endif

if ((Ollllcol_warm+Ollllcol_cool).gt.0.0) then
  call readOP (filename_op(12),edge(12),T_cool,v_turb_cool,T_warm,v_turb_warm,      &
             cs_oiii_cool,cs_oiii_warm,lambda_op_oiii,imaxoiii)
  call interpol(lambda,lambda_op_oiii,edge(12),op_cs_lambda_oiii_cool,cs_oiii_cool,      &
             ilambda,imaxoiii)
  call interpol(lambda,lambda_op_oiii,edge(12),op_cs_lambda_oiii_warm,cs_oiii_warm,      &
             ilambda,imaxoiii)
endif

```

*! \*\*\* all tau values are calculated, added up, and shifted*

```

rvfactcool=(1.0+(v_rad_cool/c))
rvfactwarm=(1.0+(v_rad_warm/c))

```

```

do i=1,ilambda

```

```

  lambda_cool(i)=lambda(i)*rvfactcool

```

```

  tau_cool(i)=( \

```

```

    Hcol_cool * op_cs_lambda_Hi_cool(i) + \
    &
    Heicol_cool * op_cs_lambda_Hei_cool(i) +Heiicol_cool * op_cs_lambda_Heii_cool(i)\
    )+ &
    Cicol_cool * op_cs_lambda_Ci_cool(i) +Ciicol_cool * op_cs_lambda_Cii_cool(i)\
    + &
    Ciiicol_cool * op_cs_lambda_Ciii_cool(i) + \
    &
    Nicol_cool * op_cs_lambda_Ni_cool(i) +Niicol_cool * op_cs_lambda_Nii_cool(i)\
    + &
    Niiicol_cool * op_cs_lambda_Niii_cool(i) + \
    &
    Oicol_cool * op_cs_lambda_Oi_cool(i) +Oiicol_cool * op_cs_lambda_Oii_cool(i)\
    + &

```

```

        Oiiicol_cool * op_cs_lambda_Oiii_cool(i) \
    )

lambda_warm(i)=lambda(i)*rvfactwarm

tau_warm(i)=( \
    Hcol_warm * op_cs_lambda_Hi_warm(i) + \
    Heicol_warm * op_cs_lambda_Hei_warm(i) + Heiicol_warm * op_cs_lambda_Heii_warm(i) \
    + &
    Cicol_warm * op_cs_lambda_Ci_warm(i) + Ciicol_warm * op_cs_lambda_Cii_warm(i) \
    + &
    Ciiicol_warm * op_cs_lambda_Ciii_warm(i) + \
    Nicol_warm * op_cs_lambda_Ni_warm(i) + Niicol_warm * op_cs_lambda_Nii_warm(i) \
    + &
    Niiicol_warm * op_cs_lambda_Niii_warm(i) + \
    Oicol_warm * op_cs_lambda_Oi_warm(i) + Oiicol_warm * op_cs_lambda_Oii_warm(i) \
    + &
    Oiiicol_warm * op_cs_lambda_Oiii_warm(i) \
)

enddo

icool=1
iwarm=1

do i=1,ilambda

! *** interpolate to input wavelengths
do j=icool,ilambda
    if (lambda(i).gt.lambda_cool(j).and.lambda(i).le.lambda_cool(j+1)) then
        tau_int_cool(i)=tau_cool(j)+( tau_cool(j+1)- tau_cool(j))/ &
        (lambda_cool(j+1)-lambda_cool(j))* &
        ( lambda(i)-lambda_cool(j))
        icool=j
        goto 1008
    endif
enddo
1008 continue
do j=iwarm,ilambda
    if (lambda(i).gt.lambda_warm(j).and.lambda(i).le.lambda_warm(j+1)) then
        tau_int_warm(i)=tau_warm(j)+( tau_warm(j+1)- tau_warm(j))/ &
        (lambda_warm(j+1)-lambda_warm(j))* &
        ( lambda(i)-lambda_warm(j))
        iwarm=j
        goto 1009
    endif
enddo
1009 continue

enddo

! *** calculate new flux that accounts for ISM absorption

if (tau_int_cool(20).eq.0.0) then
    do i=1,ilambda
        ismflux(i)=flux(i)*exp(-(tau_int_warm(i)))
    enddo
elseif (tau_int_warm(20).eq.0.0) then
    do i=1,ilambda
        ismflux(i)=flux(i)*exp(-(tau_int_cool(i)))
    enddo
else
    do i=1,ilambda
        ismflux(i)=flux(i)*exp(-(tau_int_cool(i)+tau_int_warm(i))*0.5)
    enddo

```

```

endif

! *** apply instrumental broadening

if(resolution.gt.0.0) then
  call gaussf(lambda,ismflux,ilambda,resolution,ismfluxcorr)
else
  ismfluxcorr=ismflux
endif

! *** write output file

open(3,file=filenameout,&
  form='formatted',&
  position='rewind', &
  status='unknown', &
  err=98)

write(3,('* This file was created at ',8A)) date_e(1:10), ' at ',date_e(12:13),':',&
  date_e(15:16),':',date_e(18:19)
write(3,('* by: ',A,1X,A,A)) trim(firstname), trim(lastname)," with the TEUV tool."
write(3,('* '))
write(3,('* When publishing research making use of this tool, please acknowledge:'))
write(3,('* The TEUV tool (http://astro-uni-tuebingen.de/~TEUV) used to'))
write(3,('* apply interstellar corrections to theoretical spectra was constructed'))
write(3,('* as part of the activities of the German Astrophysical Virtual Observatory\
. '))
write(3,('* '))
write(3,('* The original file was uploaded from IP address ',A)) trim(IP)
write(3,('* '))
write(3,('* The following parameters are applied:'))
write(3,('* '))
write(3,(A,F4.1)) '* Spectral resolution:', resolution
write(3,('* '))

write(3,('* Cool Component:'))
write(3,('* Temperature / K :',F8.2)) T_cool
write(3,('* Turbulent velocity / km/sec :',F8.2)) v_turb_cool
write(3,('* Radial velocity / km/sec :',F8.2)) v_rad_cool
write(3,('* Column densities / cm*cm :'))
write(3,('* H ',1PE12.5)) Hcol_cool
write(3,('* He I ',1PE12.5)) HeIcol_cool
write(3,('* He II ',1PE12.5)) HeIIcol_cool
write(3,('* C I ',1PE12.5)) Ccol_cool
write(3,('* C II ',1PE12.5)) CIIcol_cool
write(3,('* C III ',1PE12.5)) CIIIcol_cool
write(3,('* N I ',1PE12.5)) Ncol_cool
write(3,('* N II ',1PE12.5)) NIIcol_cool
write(3,('* N III ',1PE12.5)) NIIIcol_cool
write(3,('* O I ',1PE12.5)) Ocol_cool
write(3,('* O II ',1PE12.5)) OIIcol_cool
write(3,('* O III ',1PE12.5)) OIIIcol_cool
write(3,('* '))

write(3,('* Hot Component:'))
write(3,('* Temperature / K :',F8.2)) T_warm
write(3,('* Turbulent velocity / km/sec :',F8.2)) v_turb_warm
write(3,('* Radial velocity / km/sec :',F8.2)) v_rad_warm
write(3,('* Column densities / cm*cm :'))
write(3,('* H ',1PE12.5)) Hcol_warm
write(3,('* He I ',1PE12.5)) HeIcol_warm
write(3,('* He II ',1PE12.5)) HeIIcol_warm
write(3,('* C I ',1PE12.5)) Ccol_warm
write(3,('* C II ',1PE12.5)) CIIcol_warm
write(3,('* C III ',1PE12.5)) CIIIcol_warm
write(3,('* N I ',1PE12.5)) Ncol_warm
write(3,('* N II ',1PE12.5)) NIIcol_warm
write(3,('* N III ',1PE12.5)) NIIIcol_warm
write(3,('* O I ',1PE12.5)) Ocol_warm
write(3,('* O II ',1PE12.5)) OIIcol_warm

```

```

write(3, ('* O III ',1PE12.5)') OIIIcol_warm
write(3, ('* '))
write(3, ('*****'))

do i=1,ilambda
  write(3, '(1PE11.5,E12.5)') lambda(i),ismfluxcorr(i)
enddo

return
97 continue
  stop '%PROG-TEUV; err in input model'
98 continue
  stop '%PROG-TEUV; err in output file'

end PROGRAM TEUV

```

Listing A.39: Die Subroutine `gaussd.f90` existierte bereits vor dieser Arbeit und wurde verändert.

```

subroutine gaussd (x,y,npoint,T,v,f)
! ***
! *** This subroutine performs a convolution with a Gaussian
! *** with a fixed FWHM.

implicit none

integer npoint,l,k,kmin,kmax

real kBOL,c,mass,masse,T,v,coldens
real x(npoint),f(npoint),y(npoint)
real p,s,fwhm,xa

data c/2.99792458E+10/
data kBOL/1.38046E-16/
data mass/1.6726216E-24/
data masse/9.1093819E-28/

! ***
! *** maximum distance from actual point (l) to calculate profile
! ***
p=3.0

do l=1,npoint
do k=l,1,-1
if (abs(x(k)-x(l)).ge.p) goto 1
end do
1 kmin=max(k,1)
do k=l,npoint
if (abs(x(k)-x(l)).ge.p) goto 2
end do
2 kmax=min(k,npoint)

s = 0.0
! *** fwhm is put together from single FWHMs (for v and for T)
fwhm=sqrt((x(l)*2.0*(v/(c*1.0E-5)))**2+(x(l)*2.0*(sqrt(2.0*kBOL*T/mass)/(c))**2)

do k=kmin,kmax
xa=(x(k)-x(l))/(fwhm*0.6006)
call convolg (x,y,f,npoint,xa,k,l,s)
end do
f(l)=f(l)/max(s,1.0e-08)

end do

return
end subroutine gaussd

```

Listing A.40: Die Subroutine `gaussf.f90` war ebenfalls bereits vor dieser Arbeit vorhanden und wurde verändert.

```

subroutine gaussf (x,y,npoint,fwhm,f)
! ***
! *** This subroutine performs a convolution with a Gaussian
! *** with a fixed FWHM.

implicit none

integer l,k, kmin, kmax,npoint

real f(npoint), x(npoint), y(npoint)
real p, s, xa, fwhm, d

! ***
! *** maximum distance from actual point (l) to calculate profile
! ***
    p=3.0

! *** three parts with different fwhm:
    d=fwhm*0.6006

    do l=1,npoint

        do k=l,1,-1
            if (abs(x(k)-x(l)).ge.p) goto 1
        end do
1      kmin=max(k, 1)
        do k=l,npoint
            if (abs(x(k)-x(l)).ge.p) goto 2
        end do
2      kmax=min(k,npoint)

        s = 0.0
        do k=kmin,kmax

            xa=(x(k)-x(l))/d

            call convolg (x,y,f,npoint,xa,k,l,s)
        end do
        f(l)=f(l)/max(s,1.0e-08)

    end do

return

end subroutine gaussf

```



Listing A.41: Auch die Subroutine `convolg.f90` wurde nicht neu erzeugt, sondern aus einer bestehenden Version abgeändert.

```

subroutine convolg (x,y,f,npoint,xa,k,l,s)
! ***
! *** Convolution with a Gaussian.
! ***
dimension f(npoint), x(npoint), y(npoint)

g=exp(-xa*xa)

if (k.eq.1) then
  z= x( 1)-x( 2)
elseif (k.ne.1.and.k.ne.npoint) then
  z=(x( k+1)-x( k-1))*0.5
elseif (k.eq.npoint) then
  z= x(npoint-1)-x(npoint)
endif

g=g*abs(z)
f(l)=f(l)+g*y(k)
s =s+g

return

end subroutine convolg

```

Listing A.42: Die Subroutine readOP.f90 wurde im Rahmen dieser Arbeit erstellt.

```

subroutine readOP(filename ,edge, T_cool ,v_turb_cool ,T_warm ,v_turb_warm ,cs_in_cool ,cs_in_warm\
,lambda_op ,imax)
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! This program is part of TEUV
! and reads the Opacity Project data.
!
! LAST_MODIFICATION="25-Jan-2013 09:40:06"
!
! Copyright 2012 – 2013 by E. Mueller-Ringat
!
! Institute for Astronomy und Astrophysics
! Kepler Center for Astro and Particle Physics
! Eberhard Karls University
! Sand 1
! D-72076 Tuebingen
! Germany
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

implicit none
integer nx ,imax ,i
parameter nx=200000
character (len=35) filename
real nue(nx) ,cs_help(nx) ,lambda_help(nx) ,cs_in(nx) ,lambda_op(nx) ,edge
real cs_help1(nx) ,lambda_help1(nx) ,zr ,en ,rbfhyd ,c ,sigma_op ,dist ,factor ,factor3
real cs_in_cool(nx) ,cs_in_warm(nx) ,T_warm ,T_cool ,v_turb_warm ,v_turb_cool

data c/2.99792458E+18/

open(7 ,file=filename , &
form='formatted' , &
position='rewind' , &
status='old' , &
err=99)

do i=1,nx
read(7 ,*,end=1007) nue(i) ,cs_help(i)
lambda_help(i)=c/(nue(i))
enddo

1007 continue
imax=i-1

! *** wavelengths in increasing order

do i=1,imax
lambda_op(i)=lambda_help(imax-i+1)
cs_in(i)=cs_help(imax-i+1)
enddo

! *** extrapolate to the edge wavelength

lambda_op(imax+1)=edge
cs_in(imax+1)=cs_in(imax)+( cs_in(imax) - cs_in(imax-1))/ &
(lambda_op(imax) -lambda_op(imax-1))* &
(lambda_op(imax+1)-lambda_op(imax))

close(7)

! *** extrapolate in the other direction (20 steps)

lambda_help(1)=0.01
dist=(lambda_op(1)-lambda_help(1))*0.05

do i=2,20
lambda_help(i)=lambda_help(1) + dist*float(i)
enddo

if (filename(28:28).eq. 'H' .and. &
filename(29:29).ne. 'e' ) then
zr=1.0

```

```

elseif (filename(28:29).eq. 'He') then
  zr=2.0
elseif (filename(28:28).eq. 'C') then
  zr=6.0
elseif (filename(28:28).eq. 'N') then
  zr=7.0
elseif (filename(28:28).eq. 'O') then
  zr=8.0
else
  stop '%PROG-TEUV; case not yet implemented'
endif

factor=c*1.0E5/edge
factor3=factor*factor*factor
en=zr*SQRT(109737.31534*c/(factor))
rbfhyd=2.815E+29*zr*zr*zr*zr/(en*en*en*en*factor3)
sigma_op=rbfhyd*factor3

do i=1,20
  cs_help(i)=cs_in(1)*(lambda_help(i)/lambda_op(1))**3
enddo

lambda_help1=lambda_op
cs_help1=cs_in

do i=1,imax+1
  lambda_op(i+20)=lambda_help1(i)
  cs_in(i+20)= cs_help1(i)
enddo

do i=1,20
  lambda_op(i)=lambda_help(i)
  cs_in(i)=cs_help(i)
enddo

imax=imax+21

! apply Doppler broadening

if (T_cool.gt.0.0) then
  call gaussd (lambda_op,cs_in ,imax ,T_cool ,v_turb_cool , cs_in_cool)
else
  cs_in_cool=0.0
endif

if (T_warm.gt.0.0) then
  call gaussd (lambda_op,cs_in ,imax ,T_warm ,v_turb_warm , cs_in_warm)
else
  cs_in_warm=0.0
endif

return

99 continue
write (*,'("Opacity Project data file not available:",A36)') filename
stop '%PROG-TEUV; Opacity Project data file missing'

end subroutine readOP

```

Listing A.43: Die Subroutine `interpol.f90` wurde im ebenfalls Rahmen dieser Arbeit erstellt.

```

subroutine interpol(lambda,lambda_op,edge,cs_out,cs_in,ilambda,imax)
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! This program is part of TEUV
! and interpolates to the user wavelength
!
! LAST_MODIFICATION="25-Jan-2013 09:41:37"
!
! Copyright 2012 – 2013 by E. Mueller-Ringat
!
! Institute for Astronomy und Astrophysics
! Kepler Center for Astro and Particle Physics
! Eberhard Karls University
! Sand 1
! D-72076 Tuebingen
! Germany
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

  implicit none
  integer i,j,imax,ianf,ilambda
  real lambda(ilambda),lambda_op(ilambda),edge,cs_out(ilambda),cs_in(ilambda)

  cs_out=0.0

  ianf=0
  do i=1,ilambda
    do j=ianf,imax
      if (lambda_op(j).gt.lambda(i)) then
        cs_out(i)=cs_in(j-1)+(cs_in(j)-cs_in(j-1))/ &
          (lambda_op(j)-lambda_op(j-1))* &
          (lambda(i)-lambda_op(j-1))
        goto 1004
      elseif (lambda_op(j).eq.lambda(i)) then
        cs_out(i)=cs_in(j)
        goto 1004
      endif
    enddo
  1004 continue
    ianf=j-1
    if (lambda(i).gt.lambda_op(imax).or.lambda(i).gt.edge) goto 1005

  enddo
  1005 continue

  return

end subroutine interpol

```

Listing A.44: Das Programm TEUV1.f90 ist eine modifizierte Version von TEUV.f90, die keine Instrumentenauflösung, Temperatur und Turbulenzgeschwindigkeit, dafür aber eine Normierung an die Modelle anbringt.

```

PROGRAM TEUV1
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! This program applies interstellar absorption to
! theoretical stellar fluxes.
!
! LAST_MODIFICATION="25-Jan-2013 09:43:38"
!
! Copyright 2012 - 2013 by E. Mueller-Ringat
!
! Institute for Astronomy and Astrophysics
! Kepler Center for Astro and Particle Physics
! Eberhard Karls University
! Sand 1
! D-72076 Tuebingen
! Germany
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

implicit none

integer nx, i, j, imax, imaxheii, imaxhi, ilambda, istart, istart1
integer imaxci, imaxcii, imaxciii, imaxni, imaxnii, imaxniii
integer imaxoi, imaxoii, imaxoiii, inorm, iobs, ivar, nx12, nx5

parameter nx=200000
parameter nx12=2400000
parameter nx5=1000000

character (len=190) filename, filenameobs, date_e, filenameout
character (len=90) head
character (len=35) filename_op(12)

real lambda(nx), lambdaobs(nx), lambda_cool(nx), lambda_warm(nx)
real flux(nx), fluxobs(nx), ismflux(nx,5), ismfluxcorr(nx)
real v_rad_cool, v_turb_cool, T_cool
real v_rad_warm, v_turb_warm, T_warm
real resolution, c, edge(12)
real tau_int_cool(nx,5), tau_int_warm(nx,5)
real tau_cool(nx,5), tau_warm(nx,5)
real Hcol_cool, HEIcol_cool, HEIIcol_cool
real Cicol_cool, CIIcol_cool, CIIIcol_cool
real Ncol_cool, NIcol_cool, NIIcol_cool
real Ocol_cool, OIcol_cool, OIIcol_cool
real Hcol_warm, HEIcol_warm, HEIIcol_warm
real Cicol_warm, CIIcol_warm, CIIIcol_warm
real Ncol_warm, NIcol_warm, NIIcol_warm
real Ocol_warm, OIcol_warm, OIIcol_warm
real lambda_op_hi(nx), cs_hi(nx), op_cs_lambda_hi(nx)
real lambda_op_hei(nx), cs_hei(nx), op_cs_lambda_hei(nx)
real lambda_op_heii(nx), cs_heii(nx), op_cs_lambda_heii(nx)
real lambda_op_ci(nx), cs_ci(nx), op_cs_lambda_ci(nx)
real lambda_op_cii(nx), cs_cii(nx), op_cs_lambda_cii(nx)
real lambda_op_ciii(nx), cs_ciii(nx), op_cs_lambda_ciii(nx)
real lambda_op_ni(nx), cs_ni(nx), op_cs_lambda_ni(nx)
real lambda_op_nii(nx), cs_nii(nx), op_cs_lambda_nii(nx)
real lambda_op_niii(nx), cs_niii(nx), op_cs_lambda_niii(nx)
real lambda_op_oi(nx), cs_oi(nx), op_cs_lambda_oi(nx)
real lambda_op_oii(nx), cs_oii(nx), op_cs_lambda_oii(nx)
real lambda_op_oiii(nx), cs_oiii(nx), op_cs_lambda_oiii(nx)
real norm, normwav, normval(5), normvalorig
real Hcol_coolvar(5), HEIcol_coolvar(5)
real Hcol_warmvar(5), HEIcol_warmvar(5)
real rvfactcool, rvfactwarm

data c/2.99792458E+10/

data filename_op/" /home/TEUV/public_html/EUV/Hi_OP ", &
"/home/TEUV/public_html/EUV/HeI_OP ", &

```

```

        "/home/TEUV/public_html/EUV/Hell_OP " , &
        "/home/TEUV/public_html/EUV/Cl_OP " , &
        "/home/TEUV/public_html/EUV/CII_OP " , &
        "/home/TEUV/public_html/EUV/CIII_OP " , &
        "/home/TEUV/public_html/EUV/NI_OP " , &
        "/home/TEUV/public_html/EUV/NII_OP " , &
        "/home/TEUV/public_html/EUV/NIII_OP " , &
        "/home/TEUV/public_html/EUV/OI_OP " , &
        "/home/TEUV/public_html/EUV/OII_OP " , &
        "/home/TEUV/public_html/EUV/OIII_OP " /

data edge/911.75 ,           &
        504.26,227.32,         &
        1101.07,508.48,258.91 , &
        853.06,418.85,261.30 , &
        910.44,353.06,225.69/

! *** initialize variables

data tau_cool/nx5*0.0/,tau_warm/nx5*0.0/
data tau_int_cool/nx5*0.0/,tau_int_warm/nx5*0.0/

data op_cs_lambda_hi,op_cs_lambda_hei,op_cs_lambda_heii, &
        op_cs_lambda_ci,op_cs_lambda_cii,op_cs_lambda_ciii, &
        op_cs_lambda_ni,op_cs_lambda_nii,op_cs_lambda_niii, &
        op_cs_lambda_oi,op_cs_lambda_oi,op_cs_lambda_oiii/nx12*0.0/

data Hcol_warm, Helcol_warm, Hellcol_warm, &
        Cicol_warm, CIIcol_warm, CIIIcol_warm, &
        Nicol_warm, NIIcol_warm, NIIIcol_warm, &
        OIcol_warm, OIIcol_warm, OIIIcol_warm, &
        Hcol_cool, Helcol_cool, Hellcol_cool, &
        Cicol_cool, CIIcol_cool, CIIIcol_cool, &
        Nicol_cool, NIIcol_cool, NIIIcol_cool, &
        OIcol_cool, OIIcol_cool, OIIIcol_cool/24*0.0/

! *** read input parameters

read(*,*) v_turb_cool
read(*,*) v_rad_cool
read(*,*) T_cool
read(*,*) resolution
read(*,*) Hcol_cool
read(*,*) Helcol_cool
read(*,*) Hellcol_cool
read(*,*) Cicol_cool
read(*,*) CIIcol_cool
read(*,*) CIIIcol_cool
read(*,*) Nicol_cool
read(*,*) NIIcol_cool
read(*,*) NIIIcol_cool
read(*,*) OIcol_cool
read(*,*) OIIcol_cool
read(*,*) OIIIcol_cool

! *** warm component
read(*,*) v_turb_warm
read(*,*) v_rad_warm
read(*,*) T_warm
read(*,*) Hcol_warm
read(*,*) Helcol_warm
read(*,*) Hellcol_warm
read(*,*) Cicol_warm
read(*,*) CIIcol_warm
read(*,*) CIIIcol_warm
read(*,*) Nicol_warm
read(*,*) NIIcol_warm
read(*,*) NIIIcol_warm
read(*,*) OIcol_warm
read(*,*) OIIcol_warm

```

```

read(*,*) OIIIcol_warm
read(*,*) date_e
read(*,*) norm
read(*,*) normwav

! *** open the input and observation file and read wavelengths and fluxes

filename = '/home/TEUV/EUVreq/' // trim(date_e) // '/mod'
filenameobs = '/home/TEUV/EUVreq/' // trim(date_e) // '/obs'
filenameout = trim(filename) // '.json'
print *, filename
print *, filenameout

open(2, file=filename, &
      form='formatted', &
      position='rewind', &
      status='old', &
      err=97)

do i=1,70
  read(2,*) head
  if(head(1:1).ne."*") goto 2
enddo
2 continue
backspace 2

do i=1,nx

  read(2,*,end=1000)lambda(i), flux(i)

  if(lambda(i).ge.928) goto 1002

enddo

write(*,'("maximum number of wavelength points (",l8,")exceeded")') nx
stop '%PROG-TEUV1, maximum number of wavelength points exceeded'

1002 continue
ilambda=i-1

close(2)

open(2, file=filenameobs, &
      form='formatted', &
      position='rewind', &
      status='old', &
      err=98)

do i=1,70
  read(2,*) head
  print *, head
  if(head(1:1).ne."*") goto 22
enddo
22 continue
backspace 2

do i=1,nx

  read(2,*,end=1000)lambdaobs(i), fluxobs(i)

  if(lambdaobs(i).ge.928) goto 1000

enddo

write(*,'("maximum number of wavelength points (",l8,")exceeded")') nx
stop '%PROG-TEUV1, maximum number of wavelength points exceeded'

1000 continue
iobs=i-1

! *** read the Opacity Project RBF data files

```

```

! *** interpolate between the OP and input wavelengths

  if ((Hcol_warm+Hcol_cool).gt.0.0)      then
    call readOP1(filename_op(1),edge(1),cs_hi,lambda_op_hi,imaxhi)
    call interpol(lambda,lambda_op_hi,911.75,op_cs_lambda_hi,cs_hi,ilambda,imaxhi)
  endif

  if ((Helcol_warm+Helcol_cool).gt.0.0)  then
    call readOP1(filename_op(2),edge(2),cs_hei,lambda_op_hei,imax)
    call interpol(lambda,lambda_op_hei,504.26,op_cs_lambda_hei,cs_hei,ilambda,imax)
  endif

  if ((Hellcol_warm+Hellcol_cool).gt.0.0) then
    call readOP1(filename_op(3),edge(3),cs_heii,lambda_op_heii,imaxheii)
    call interpol(lambda,lambda_op_heii,227.32,op_cs_lambda_heii,cs_heii,ilambda,imaxheii)
  endif

  if ((Clcol_warm+Clcol_cool).gt.0.0)    then
    call readOP1(filename_op(4),edge(4),cs_ci,lambda_op_ci,imaxci)
    call interpol(lambda,lambda_op_ci,1101.07,op_cs_lambda_ci,cs_ci,ilambda,imaxci)
  endif

  if ((Cllcol_warm+Cllcol_cool).gt.0.0)  then
    call readOP1(filename_op(5),edge(5),cs_cii,lambda_op_cii,imaxcii)
    call interpol(lambda,lambda_op_cii,508.48,op_cs_lambda_cii,cs_cii,ilambda,imaxcii)
  endif

  if ((Clllcol_warm+Clllcol_cool).gt.0.0) then
    call readOP1(filename_op(6),edge(6),cs_ciii,lambda_op_ciii,imaxciii)
    call interpol(lambda,lambda_op_ciii,258.91,op_cs_lambda_ciii,cs_ciii,ilambda,imaxciii)
  endif

  if ((Nlcol_warm+Nlcol_cool).gt.0.0)    then
    call readOP1(filename_op(7),edge(7),cs_ni,lambda_op_ni,imaxni)
    call interpol(lambda,lambda_op_ni,853.06,op_cs_lambda_ni,cs_ni,ilambda,imaxni)
  endif

  if ((Nllcol_warm+Nllcol_cool).gt.0.0)  then
    call readOP1(filename_op(8),edge(8),cs_nii,lambda_op_nii,imaxnii)
    call interpol(lambda,lambda_op_nii,418.85,op_cs_lambda_nii,cs_nii,ilambda,imaxnii)
  endif

  if ((Nlllcol_warm+Nlllcol_cool).gt.0.0) then
    call readOP1(filename_op(9),edge(9),cs_niii,lambda_op_niii,imaxniii)
    call interpol(lambda,lambda_op_niii,261.30,op_cs_lambda_niii,cs_niii,ilambda,imaxniii)
  endif

  if ((Olcol_warm+Olcol_cool).gt.0.0)    then
    call readOP1(filename_op(10),edge(10),cs_oi,lambda_op_oi,imaxoi)
    call interpol(lambda,lambda_op_oi,910.44,op_cs_lambda_oi,cs_oi,ilambda,imaxoi)
  endif

  if ((Ollcol_warm+Ollcol_cool).gt.0.0)  then
    call readOP1(filename_op(11),edge(11),cs_oi,lambda_op_oi,imaxoi)
    call interpol(lambda,lambda_op_oi,353.06,op_cs_lambda_oi,cs_oi,ilambda,imaxoi)
  endif

  if ((Olllcol_warm+Olllcol_cool).gt.0.0) then
    call readOP1(filename_op(12),edge(12),cs_oiii,lambda_op_oiii,imaxoiii)
    call interpol(lambda,lambda_op_oiii,225.69,op_cs_lambda_oiii,cs_oiii,ilambda,imaxoiii)
  endif

! *** all tau values are calculated, added up, and shifted (also for varied column densities)

Hcol_coolvar(1)=Hcol_cool
Hcol_coolvar(2)=10.0*Hcol_cool
Hcol_coolvar(3)=0.1*Hcol_cool
Hcol_coolvar(4)=Hcol_cool
Hcol_coolvar(5)=Hcol_cool
Helcol_coolvar(1)=Helcol_cool
Helcol_coolvar(2)=Helcol_cool

```



```

Helcol_coolvar(3)=Helcol_cool
Helcol_coolvar(4)=10.0*Helcol_cool
Helcol_coolvar(5)=0.1*Helcol_cool
Hcol_warmvar(1)=Hcol_warm
Hcol_warmvar(2)=10.0*Hcol_warm
Hcol_warmvar(3)=0.1*Hcol_warm
Hcol_warmvar(4)=Hcol_warm
Hcol_warmvar(5)=Hcol_warm
Helcol_warmvar(1)=Helcol_warm
Helcol_warmvar(2)=Helcol_warm
Helcol_warmvar(3)=Helcol_warm
Helcol_warmvar(4)=10.0*Helcol_warm
Helcol_warmvar(5)=0.1*Helcol_warm

rvfactcool=(1.0+(v_rad_cool/c))
rvfactwarm=(1.0+(v_rad_warm/c))

do i=1,ilambda

lambda_cool(i)=lambda(i)*rvfactcool

do ivar=1,5
tau_cool(i,ivar)=(
Hcol_coolvar(ivar)*op_cs_lambda_hi(i) + &
Helcol_coolvar(ivar)*op_cs_lambda_hei(i)+Heicol_cool*op_cs_lambda_heii(i) + &
Cicol_cool*op_cs_lambda_ci(i) +Ciicol_cool*op_cs_lambda_cii(i) + &
Ciicol_cool*op_cs_lambda_ciii(i) + &
Nicol_cool*op_cs_lambda_ni(i) +Niicol_cool*op_cs_lambda_nii(i) + &
Niicol_cool*op_cs_lambda_niii(i) + &
Oicol_cool*op_cs_lambda_oi(i) +Oiicol_cool*op_cs_lambda_oii(i) + &
Oiiicol_cool*op_cs_lambda_oiii(i) )
enddo

lambda_warm(i)=lambda(i)*rvfactwarm

do ivar=1,5
tau_warm(i,ivar)=(
Hcol_warmvar(ivar)*op_cs_lambda_hi(i) + &
Helcol_warmvar(ivar)*op_cs_lambda_hei(i)+Heicol_warm*op_cs_lambda_heii(i) + &
Cicol_warm*op_cs_lambda_ci(i) +Ciicol_warm*op_cs_lambda_cii(i) + &
Ciicol_warm*op_cs_lambda_ciii(i) + &
Nicol_warm*op_cs_lambda_ni(i) +Niicol_warm*op_cs_lambda_nii(i) + &
Niicol_warm*op_cs_lambda_niii(i) + &
Oicol_warm*op_cs_lambda_oi(i) +Oiicol_warm*op_cs_lambda_oii(i) + &
Oiiicol_warm*op_cs_lambda_oiii(i) )
enddo
enddo

! *** interpolate to input wavelengths

do ivar=1,5

istart=1
istart1=1

do i=1,ilambda

do j=istart,ilambda
if(lambda(i).gt.lambda_cool(j).and.lambda(i).le.lambda_cool(j+1)) then
tau_int_cool(i,ivar)=tau_cool(j,ivar)+(tau_cool(j+1,ivar)-tau_cool(j,ivar))/ &
( lambda_cool(j+1)- lambda_cool(j)) * &
( lambda(i)- lambda_cool(j))
istart=j
goto 1008
endif
enddo
1008 continue
do j=istart1,ilambda

```



```

do i=1,ilambda-1,ilambda/500.
  write(3,'(7X,A1,1PE12.5,A1,1PE12.5,A2)')'[',lambda(i),',',flux(i)*normvalorig,']',
enddo
write(3,'(7X,A1,1PE12.5,A1,1PE12.5,A5)')'[',lambda(ilambda),',',flux(ilambda)*normvalorig\
,']}]',

! *** ISM_choice
write(3,'(A)')'"ISM_choice":{'
write(3,'(A)')'"label": "ISM_choice",'
write(3,'(A)')'"data": ['
do i=1,ilambda-1,ilambda/400.
  write(3,'(7X,A1,1PE12.5,A1,1PE12.5,A2)')'[',lambda(i),',',ismflux(i,1)*normval(1),']',
enddo
write(3,'(7X,A1,1PE12.5,A1,1PE12.5,A3)')'[',lambda(ilambda),',',ismflux(ilambda,1)*\
normval(1),']}]'
write(3,'(A)')',

! *** 10xH
write(3,'(A)')'"10xH":{'
write(3,'(A)')'"label": "10xH",'
write(3,'(A)')'"data": ['
do i=1,ilambda-1,ilambda/400.
  write(3,'(7X,A1,1PE12.5,A1,1PE12.5,A2)')'[',lambda(i),',',ismflux(i,2)*normval(2),']',
enddo
write(3,'(7X,A1,1PE12.5,A1,1PE12.5,A3)')'[',lambda(ilambda),',',ismflux(ilambda,2)*\
normval(2),']}]'
write(3,'(A)')',

! *** 0.1xH
write(3,'(A)')'"0.1xH":{'
write(3,'(A)')'"label": "0.1xH",'
write(3,'(A)')'"data": ['
do i=1,ilambda-1,ilambda/400.
  write(3,'(7X,A1,1PE12.5,A1,1PE12.5,A2)')'[',lambda(i),',',ismflux(i,3)*normval(3),']',
enddo
write(3,'(7X,A1,1PE12.5,A1,1PE12.5,A3)')'[',lambda(ilambda),',',ismflux(ilambda,3)*\
normval(3),']}]'
write(3,'(A)')',

! *** 10xHel
write(3,'(A)')'"10xHel":{'
write(3,'(A)')'"label": "10xHel",'
write(3,'(A)')'"data": ['
do i=1,ilambda-1,ilambda/400.
  write(3,'(7X,A1,1PE12.5,A1,1PE12.5,A2)')'[',lambda(i),',',ismflux(i,4)*normval(4),']',
enddo
write(3,'(7X,A1,1PE12.5,A1,1PE12.5,A3)')'[',lambda(ilambda),',',ismflux(ilambda,4)*\
normval(4),']}]'
write(3,'(A)')',

! *** 0.1xHel
write(3,'(A)')'"0.1xHel":{'
write(3,'(A)')'"label": "0.1xHel",'
write(3,'(A)')'"data": ['
do i=1,ilambda-1,ilambda/400.
  write(3,'(7X,A1,1PE12.5,A1,1PE12.5,A2)')'[',lambda(i),',',ismflux(i,5)*normval(5),']',
enddo
write(3,'(7X,A1,1PE12.5,A1,1PE12.5,A3)')'[',lambda(ilambda),',',ismflux(ilambda,5)*\
normval(5),']}]'
write(3,'(A)')',

return
97 continue
stop '%PROG-TEUV; err in input model'
98 continue
stop '%PROG-TEUV; err in input observation'
99 continue
stop '%PROG-TEUV; err in output file'

end PROGRAM TEUV1

```

Listing A.45: Die Subroutine readOP1.f90 wurde von readOP.f90 abgewandelt.

```

subroutine readOP1( filename ,edge,cs_in ,lambda_op ,imax)
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! This program is part of TEUV
! and reads the Opacity Project data.
!
! LAST_MODIFICATION="25-Jan-2013 09:43:15"
!
! Copyright 2012 – 2013 by E. Mueller–Ringat
!
! Institute for Astronomy und Astrophysics
! Kepler Center for Astro and Particle Physics
! Eberhard Karls University
! Sand 1
! D–72076 Tuebingen
! Germany
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
implicit none

integer nx,imax,i
parameter nx=200000
character (len=35) filename
real nue(nx) ,cs_help(nx) ,lambda_help(nx) ,cs_in(nx) ,lambda_op(nx) ,edge
real cs_help1(nx) ,lambda_help1(nx) ,zr,en,rbfhyd ,c,sigma_op ,dist ,factor ,factor3

data c/2.99792458E+18/

open(7, file=filename , &
      form='formatted' ,&
      position='rewind' , &
      status='old' , &
      err=96
      )

do i=1,nx
  read(7,* ,end=1007) nue(i) ,cs_help(i)
  lambda_help(i)=c/(nue(i))
enddo
1007 continue
  imax=i-1

! *** wavelengths in increasing order

do i=1,imax
  lambda_op(i)=lambda_help(imax-i+1)
  cs_in(i)=cs_help(imax-i+1)
enddo

! *** extrapolate to the edge wavelength

lambda_op(imax+1)=edge
cs_in(imax+1)=cs_in(imax)+( cs_in(imax)– cs_in(imax-1))/ &
              ( lambda_op(imax)–lambda_op(imax-1))* &
              (lambda_op(imax+1)– lambda_op(imax))

close(7)

! *** extrapolate in the other direction (20 steps)
lambda_help(1)=0.01
dist=(lambda_op(1)–lambda_help(1))*0.05
do i=2,20
  lambda_help(i)=lambda_help(1) + dist*float(i)
enddo

if (filename(28:28).eq. 'H'.and. &
     filename(29:29).ne. 'e' ) then
  zr=1.0
elseif (filename(28:29).eq. 'He') then
  zr=2.
elseif (filename(28:28).eq. 'C') then
  zr=6.

```

```

elseif (filename(28:28).eq. 'N') then
    zr=7.
elseif (filename(28:28).eq. 'O') then
    zr=8.
else
    stop '%PROG-TEUV; case not yet implemented'
endif

factor=c*1.0E5/edge
factor3=factor*factor*factor
en=zr*SQR(109737.31534*c/(factor))
rbfhyd=2.815E+29*zr*zr*zr*zr/(en*en*en*en*en*factor3)
sigma_op=rbfhyd*factor3

do i=1,20
    cs_help(i)=cs_in(1)*(lambda_help(i)/lambda_op(1))**3
enddo

lambda_help1=lambda_op
cs_help1=cs_in

do i=1,imax+1
    lambda_op(i+20)=lambda_help1(i)
    cs_in(i+20)=cs_help1(i)
enddo

do i=1,20
    lambda_op(i)=lambda_help(i)
    cs_in(i)=cs_help(i)
enddo

imax=imax+21

return

96 continue
write (*,'("Opacity Project data file not available:",A36)') filename
stop '%PROG-TEUV; Opacity Project data file missing'

end subroutine readOP1

```

Listing A.46: Die TEUV . sh-Prozedur wird von der Webseite gestartet, um TEUV1 . Linux\_X86 anzuwenden, auf Fehler zu prüfen und die erzeugten Dateien dem Nutzer zugänglich zu machen.

```
#!/bin/sh
#####
# This script is used to apply interstellar corrections in EUV range to
# theoretical stellar fluxes
#
# LAST_MODIFICATION="25-Jan-2013 09:45:06"
#
# Copyright 2012 – 2013 by E. Mueller–Ringat
#
# Institute for Astronomy und Astrophysics
# Kepler Center for Astro and Particle Physics
# Eberhard Karls University
# Sand 1
# D-72076 Tuebingen
# Germany
#####
    lastname=${1}
    firstname=${2}
    email=${3}
    vturbcold=${4}
    vradcold=${5}
    Tcold=${6}
    resolution=${7}
    coldensHcold=${8}
    coldensHelcold=${9}
    coldensHellcold=${10}
    coldensClcold=${11}
    coldensCIIcold=${12}
    coldensCIIIcold=${13}
    coldensNlcold=${14}
    coldensNIIcold=${15}
    coldensNIIIcold=${16}
    coldensOlcold=${17}
    coldensOIIcold=${18}
    coldensOIIIcold=${19}
    vturbhot=${20}
    vradhot=${21}
    Thot=${22}
    coldensHhot=${23}
    coldensHelhot=${24}
    coldensHellhot=${25}
    coldensCIhot=${26}
    coldensCIIhot=${27}
    coldensCIIIhot=${28}
    coldensNIhot=${29}
    coldensNIIhot=${30}
    coldensNIIIhot=${31}
    coldensOIhot=${32}
    coldensOIIhot=${33}
    coldensOIIIhot=${34}
    upname=${35}
    beoname=${36}
    norm=${37}
    normwav=${38}
    date=${39}
#
echo "The following person just submitted a request:" > MAIL
echo $firstname >> MAIL
echo $lastname >> MAIL
echo "The email address is: " >> MAIL
echo $email >> MAIL
/usr/bin/mail -r "TEUV" -s "TEUV: Request submitted from ${lastname} ${firtsname} ${email}" \
ringat@astro.uni-tuebingen.de < MAIL
rm MAIL
#
mkdir /home/TEUV/EUVreq/${date}
chmod 777 /home/TEUV/EUVreq/${date}
```

```

cp /home/TEUV/uploads/${upname} /home/TEUV/EUVreq/${date}/mod
cp /home/TEUV/uploads/${beoname} /home/TEUV/EUVreq/${date}/obs
#
/home/TEUV/EUVreq/TEUV1.Linux_X86 2> /home/TEUV/EUVreq/ERROR_ > /home/TEUV/EUVreq/OUTPUT_ << \
EOs
${vturbcold}
${vradcold}
${Tcold}
${resolution}
${coldensHcold}
${coldensHelcold}
${coldensHellcold}
${coldensClcold}
${coldensCIIcold}
${coldensCIIIcold}
${coldensNlcold}
${coldensNIIcold}
${coldensNIIIIcold}
${coldensOlcold}
${coldensOIIcold}
${coldensOIIIIcold}
${vturbhot}
${vradhot}
${Thot}
${coldensHhot}
${coldensHelhot}
${coldensHellhot}
${coldensClhot}
${coldensCIIhot}
${coldensCIIIhot}
${coldensNIhot}
${coldensNIIhot}
${coldensNIIIIhot}
${coldensOIhot}
${coldensOIIhot}
${coldensOIIIIhot}
${date}
${norm}
${normwav}
EOs
# first check if there was an error calculating the new flux
#
err_count='wc -l /home/TEUV/EUVreq/ERROR_ | cut -c1-2'
if [ ${err_count} -gt 0 ]
then
    cat > MAIL <<EOF
Dear TEUV administrator ,

an error occured while calculating the new flux
corrected with interstellar absorption.

Please check that.

Yours
TEUV Team
EOF
#
/usr/bin/mail -r "TEUV" -s "TEUV: Error" ringat@astro.uni-tuebingen.de < MAIL
rm MAIL
fi
#
mv /home/TEUV/EUVreq/${date}/mod.json /home/TEUV/EUVreq/out.json
chmod 755 /home/TEUV/EUVreq/out.json
#
# check if the programm did produce an output file
if test ! -s /home/TEUV/EUVreq/out.json
then
    /usr/bin/mail -r "TEUV" -s "TEUV: Error , no output file" ringat@astro.uni-tuebingen.de < \
/home/TEUV/EUVreq/ERROR_
fi
#####

```

## Danksagung

An dieser Stelle möchte ich mich bei den Menschen bedanken, die mich während meiner Arbeit unterstützt haben. Besonderer Dank gilt den folgenden Personen bzw. Personenkreisen, jedoch gilt mein Dank auch all denen, die hier nicht genannt werden.

Zuerst möchte ich mich bei Klaus Werner und Thomas Rauch bedanken, die mir diese Arbeit ermöglicht und mich bei ihrer Anfertigung immer wieder unterstützt haben. Thomas Rauch möchte ich besonders für die Einarbeitung und Hilfestellung im Bezug auf die GAVO-Dienste sowie die vielen, geduldigen Antworten auf meine Fragen danken.

Vielen Dank an meine Freunde und Kollegen der Arbeitsgruppe „Optische und UV-Astronomie“, vor allem Stephan Hartmann, Thorsten Nagel, Felix Friederich, Kim Gehron, Ruth Kowalski, Nicole Reindl, Ulrich Müller, Marc Ziegler und Maren Mahserci für die vielen heiteren Gespräche und fachlichen wie außerfachlichen Diskussionen. Danke auch an die anderen Mitarbeiter des IAAT, besonders die Besucher der Espressorunde, für die freundliche Arbeitsatmosphäre und die Erweiterung des eigenen Blickwinkels.

Mein Dank gilt auch Steve White, Jochen Klar, Harry Enke und dem *AstroGrid-D*-Team sowie den Mitarbeitern von GAVO, die zusammen mit mir an den Diensten gearbeitet und mich bei ihrer Erweiterung unterstützt haben.

Danke an die DFG für die Förderung im Rahmen des Projekts WE 1312/41-1.

Danke an Sarah Bartram und den DAAD für die Unterstützung im Rahmen eines Research Internships.

Zu guter Letzt möchte ich mich bei meinem Mann Sebastian, meinen Eltern Günter und Andrea sowie bei meinen Geschwistern Benjamin und Stefanie bedanken, die immer ein offenes Ohr für mich hatten. Mein Dank gilt an dieser Stelle ebenfalls den Familien Wellerdiek und Müller. Ihr alle habt mich immer wieder motiviert, mich soweit wie möglich in allen Bereichen unterstützt, mir den Rücken frei gehalten und mir ein Zuhause geschaffen, in dem ich immer willkommen war.