

First version of the entry "Proof-Theoretic Semantics" in the Stanford Encyclopedia of Philosophy, which was submitted in 2011. For the version which finally appeared in 2012 it was reduced to one third of its length.

Cp. <https://plato.stanford.edu/entries/proof-theoretic-semantics/>

Proof-Theoretic Semantics

(SEP Entry)

Peter Schroeder-Heister*

Wilhelm-Schickard-Institut für Informatik

Universität Tübingen

Sand 13, 72076 Tübingen, Germany

psh@uni-tuebingen.de

Contents

1	Introduction	2
2	Background	4
2.1	General proof theory: consequence vs. proofs	4
2.2	Inferentialism	6
2.3	Constructivism and anti-realism	7
2.4	Intuitionistic logic and its semantics	9
2.5	Gentzen-style proof theory: Reduction, normalization, cut elimination .	11
2.6	Proofs, terms and their computability	14
2.7	‘Tonk’, the Belnap criteria and other constraints	15
3	Local versus global proof-theoretic semantics	19
4	Some versions of proof-theoretic semantics	20
4.1	The semantics of implications: Admissibility, derivability, rules	20
4.1.1	Lorenzen’s operative logic	21
4.1.2	Kutschera’s Gentzen semantics	24
4.1.3	Schroeder-Heister’s extension of natural deduction	26
4.1.4	Digression: Paul Hertz’s systems	28
4.1.5	Digression: Generalized elimination rules	28
4.2	The Semantics of derivations as based on introduction rules	29

*This paper was supported by the French-German DFG-ANR project “Hypothetical Reasoning — Logical and Semantical Perspectives (HYPOTHESES)” (DFG Schr 275/16-1).

4.2.1	Inversion principles and harmony	29
4.2.2	Prawitz's notion proof-theoretic validity	32
4.2.3	Examples of proof-theoretic validity	38
4.2.4	Martin-Löf type theory	41
4.3	Clausal definitions and definitional reasoning	44
4.3.1	The challenge from logic programming	44
4.3.2	Definitional Reflection	46
4.3.3	Examples, applications and problems of definitional reflection .	54
4.4	Characterization versus semantics	58
4.4.1	Structural characterization of logical constants	58
4.4.2	Categorical approaches to proof-theoretic semantics	60
5	Extensions and alternatives to standard proof-theoretic semantics	61
5.1	Elimination rules as basic, dual approaches	61
5.2	Negation and denial	68
5.3	Assertion-denial harmony	69
5.4	Harmony and reflection in the sequent calculus	71
5.5	Classical logic	75
5.6	A general perspective: The categorical and the hypothetical	79
6	Miscellaneous and outlook	81
A	Appendix: Gentzen-style natural deduction and sequent calculus	86
	References	90

1 Introduction

Proof-theoretic semantics is an alternative to truth-condition semantics. It is based on the fundamental assumption that the central notion in terms of which meanings are assigned to certain expressions of our language, in particular to logical constants, is that of *proof* rather than *truth*. In this sense proof-theoretic semantics is *semantics in terms of proof*. Proof-theoretic semantics also means the *semantics of proofs*, i.e., the semantics of entities which describe how we arrive at certain assertions given certain assumptions. Both aspects of proof-theoretic semantics can be intertwined, i.e. the semantics of proofs is itself often given in terms of proofs.

Proof-theoretic semantics has several roots, the most specific one being Gentzen's (1934/35) remarks that the introduction rules in his calculus of natural deduction define the meanings of logical constants, while the elimination rules can be obtained as a consequence of this definition. More broadly, it belongs to what Prawitz (1971,1972) called *general proof theory*, i.e., the study of proofs as entities in their own right (see

section 2.1). Even more broadly, it is part of the tradition according to which the meaning of a term should be explained by reference to the way it is *used* in our language.

Within philosophy, proof-theoretic semantics has mostly figured under the heading “theory of meaning”. This terminology follows Dummett, who claimed that the theory of meaning is the basis of theoretical philosophy, a view which he attributed to Frege. The term *proof-theoretic semantics* was proposed by Schroeder-Heister in order not to leave the term “semantics” to the denotationists alone — after all, “semantics” is the standard term for investigations dealing with the meaning of linguistic expressions¹. Furthermore, the term “proof-theoretic semantics” covers philosophical and technical aspects likewise. In 1999, the first conference with this title took place in Tübingen².

Although the “*meaning as use*” approach has been quite prominent for half a century now and has provided one of the cornerstones of the philosophy of language, in particular of ordinary language philosophy, it has never prevailed in the *formal* semantics of artificial and natural languages. In formal semantics, the *denotational* approach, which starts with interpretations of singular terms and predicates, then fixes the meaning of sentences in terms of truth conditions, and finally defines logical consequence as truth preservation under all interpretations, has always been dominant. The main reason for this, as I see it, is the fact that from the very beginning, denotational semantics received an authoritative rendering in Tarski’s (1935) theory of truth, which combined philosophical claims with a sophisticated technical exposition and, at the same time, laid the ground for model theory as a mathematical discipline. Compared to this development, the “meaning as use” idea was a slogan supported by strong philosophical arguments, but without much formal underpinning.

Proof-theoretic semantics attempts to develop a formal semantic theory. As one would expect, it uses ideas from *proof theory* as a mathematical discipline, similar to the way truth-condition semantics relies on model theory. However, just this is the basis of a fundamental misunderstanding of proof-theoretic semantics. To a great extent, the development of mathematical proof theory has been dominated by the formalist reading of Hilbert’s program as dealing with *formal* proofs exclusively, in contradistinction to model theory as concerned with the (denotational) *meaning* of expressions. This dichotomy has entered many textbooks of logic in which “semantics” means model-theoretic semantics and “proof theory” denotes the proof theory of formal systems. The result is that “proof-theoretic semantics” sounds like a contradiction in terms

¹I used this term in lectures in Stockholm in 1987. I am not aware of any earlier usage of this term, although, of course, the subject itself was already there, in particular in the Swedish school of proof theory established by Prawitz and Martin-Löf. The term appeared first in print in the abstract (Schroeder-Heister, 1991c).

²Jointly organized with Reinhard Kahle. See the special issue of *Synthese* (Kahle & Schroeder-Heister, 2006), which resulted from this conference.

even today, although the identification of “proof-theory” with “syntax” or “theory of syntax” has not even been true for proof theory in Hilbert’s sense (see section 2.1).

Strictly speaking, the formalist reading of proof theory is not any more foreign to the understanding of ‘real’ argumentation than model theory is to the interpretation of natural language. In order to apply proof-theoretic results, one has to consider formal proofs to be *representations* of proper arguments, just as, in order to apply model-theoretic methods, one has to consider formulas to be *representations* of proper sentences of a natural language like English. English is not *per se* a formal language, and arguments are not *per se* formal derivations. In this sense, the term “proof-theoretic semantics” is not any more provocative than Montague’s (1970) “English as a formal language”. Both proof-theoretic semantics and model-theoretic semantics are *indirect* in that they can only be applied via a *formal reading* of aspects of natural language. The basic difference lies in what these aspects are: proof-theoretic semantics starts with arguments and represents them by derivations, whereas model-theoretic semantics starts with names and sentences and represents them by individual terms and formulas.

It was the Swedish school of proof theory with its representatives Martin-Löf and Prawitz which paved the way for a non-formalist philosophical understanding of proofs. Although originally dealing with the mathematical proof-theory of formal systems, Prawitz and Martin-Löf soon realized that many of the concepts and methods developed there had a non-technical counterpart when looking at formal proofs as formal representations of “genuine” proofs or arguments. In taking Gentzen’s remarks on the definitional significance of introduction and elimination rules seriously, they developed the cornerstones of proof-theoretic semantics.

In order to keep the level of reasoning apart from its formal representation, one should terminologically distinguish between proofs as formal objects and proofs as acts which establish a proposition in the same sense in which one distinguishes between formulas as formal objects and propositions represented by them. From that one should even separate demonstrations as metalinguistic, i.e. mathematical proofs. However, as “proof” is a term commonly used in all these meanings, we will not always stick to this terminological distinction and often use “proof” in the formal sense of “derivation”. It should always be clear from the context what is meant.

2 Background

2.1 General proof theory: consequence vs. proofs

The term “general proof theory” was coined by Prawitz (1971, 1972). In general proof theory, “proofs are studied in their own right in the hope of understanding their nature” (Prawitz, 1972, p. 123). It is distinguished from “reductive proof theory”, which is the “attempt to analyze the proofs of mathematical theories with the intention

of reducing them to some more elementary part of mathematics such as finitistic or constructive mathematics” (*ibid.*). Whereas Hilbert-style proof theory is of the reductive kind, Gentzen-style proof theory belongs primarily to general proof theory (although it provides many techniques and results pertaining to reductive proof theory as well). In a similar way, Kreisel had asked for a re-orientation of proof theory, without proposing a specific term for it. He “explains recent work in proof theory from a neglected point of view. Proofs and their representations by formal derivations are treated as principal objects of study, not as mere tools for analyzing the consequence relation.” (Kreisel, 1971, p. 109) Whereas Kreisel focuses on the dichotomy between a theory of proofs and a theory of provability, Prawitz concentrates on the different goals proof theory may pursue. However, both stress the necessity of studying proofs as fundamental entities by means of which we acquire demonstrative (especially mathematical) knowledge. This means in particular that proofs are epistemic entities which should not be conflated with formal proofs or derivations. They are rather what derivations denote when they are considered as representations of arguments. In discussing Prawitz’s (1971) survey, Kreisel (1971, p. 111) explicitly speaks of a “mapping” between derivations and mental acts and considers it as a task of proof theory to elucidate this mapping, including the investigation of the identity of proofs, a topic that Prawitz and Martin-Löf had put on the agenda.

This means that in general proof theory we are not solely interested in whether B follows from A , but in the way by means of which we arrive at B starting from A . In this sense general proof theory is intensional and epistemological in character, whereas model theory, which is interested in the consequence relation and not in the way of establishing it, is extensional and non-epistemological. If we look at proof-theoretic semantics from this point of view and consider it as the semantics of proofs, it appears more like an addition or supplement to classical model-theoretic semantics rather than a rival of it. Proof-theoretic semantics would deal with the neglected epistemological aspect of mathematics, the way of achieving and securing mathematical knowledge, in addition to the truth of mathematical theorems and the consequence relation between mathematical propositions. Proof-theoretic semantics would deliver an intensional theory of consequence in addition to the classical extensional one. At the level of proof and consequence the intensional/extensional dichotomy would correspond to the one between general proof theory and model theory: General proof theory as the intensional theory of proofs, and model theory as the theory of consequence. Why is it then that proof-theoretic semantics competes with model theory and arrives at a notion of consequence, which is also extensionally different from the classical notion, being intuitionistic in spirit, or at least biased towards intuitionism?

To understand this fact one should have a look at the classical notion of intension. As remarked above, the theory of intensions normally assumes that intensions

are extensions depending on indices (worlds, situations, reference points). This indexical view of intensions goes back to Carnap’s (1947) “Meaning and Necessity” and is, e.g., the cornerstone of Montague’s formal grammar of natural languages. Roughly speaking, an intension of an expression is a function which associates with an index (world, situation, reference point) the extension of that expression at this index. In this sense there is no independent notion of intension in classical theories. Conceptually, i.e., with regard to the order of explanation, intensions are secondary to extensions. This is crucially different in proof-theoretic semantics. Proof as the intensional aspect of consequence is not derived from a pre-existing concept of consequence, but an independent concept. Proof determines consequence and justifies it. This means that proofs come first, and the result of a proof, viz. the consequence statement being proved, is, as its output, contingent on it. So in the case of proofs we have a genuine notion of intension which represents the way in which consequence is accessible to us³. But if consequence is determined by proofs and not an independent concept, then the properties of proofs have a strong bearing on the consequence relation, and if the concept of proof is intuitionistic, then the consequence relation as its output will be intuitionistic as well. Therefore proof-theoretic semantics is not merely the intensional supplement to the classical theory of consequence, but brings with it a novel theory of consequence, namely consequence as based on proofs. That most versions proof-theoretic semantics are rivals of model-theoretic semantics, is due to the fact that, at least in the area of proofs and consequence, intensions determine extensions and are not derived from them. The independent, and in particular non-indexical character of proofs-as-intensions is an absolutely crucial feature of proof-theoretic semantics.

2.2 Inferentialism

Proof-theoretic semantics is inherently inferential, as it is inferential activity which manifests itself in proofs. It thus belongs to *inferentialism*, according to which inferences and the rules of inference establish the meaning of expressions, in contradistinction to *denotationalism*, according to which denotations are the primary sort of meaning. In the last two decades inferentialist approaches to semantical issues have gained considerable ground, though they are still far from being mainstream. A prominent inferentialist approach is Brandom’s, who actually coined the term “inferentialism”⁴. More explicitly than ordinary language philosophy he attempts to derive denotational

³We do not speak here of consequence being “given” to us, as this might suggest a pre-existing notion of consequence. Frege speaks of intensions as “ways of being given” [“Art des Gegebenseins” (Frege, 1892, p. 26)]. This terminology indicates his view that extensions (his “Bedeutungen”) are independent of intensions (his “Sinne”), even if he does not necessarily suggest that intensions are dependent on extensions as in the later indexical view.

⁴I owe this terminological hint to O. Hjortland’s blog “Nothing of Consequence” (notofcon.blogspot.com, post of 27 December 2007.)

meaning from inferential meaning, using the idea that meaning is rooted in proofs as his starting point.⁵

I do not want to discuss the relationship between proof-theoretic semantics and inferentialism in detail, as “inferentialism” has received too many different meanings not all of which overlap with what proof-theoretic semantics aims at. Inferentialism is certainly wider than proof-theoretic semantics, as proof-theoretic semantics often has a very precise picture of what inferences look like and which criteria they have to satisfy, whereas inferentialism often just means that meaning is ‘somehow’ determined by inferences. Inferentialism is sometimes not much more specific than what is expressed by the ‘meaning-as-use’ slogan, except that ‘use’ is now understood as ‘inference’. So it seems to me to be futile to use ‘inferentialism’ to explain what proof-theoretic semantics means. On the contrary, proof-theoretic semantics gives examples of what ‘inferentialism’ could mean. (This also corresponds to how Brandom understood general inferentialism in relation to the more specific theories of Dummett and Gentzen.)

Proof-theoretic semantics even gives an answer to a fundamental problem of inferentialism, namely how to get from inferences to denotations. Proof-theoretic semantics’ idea here is that denotations are themselves the result of a form of inference or transformation, namely the computation of normal forms of certain expressions, at least in mathematics. In this way it evades the criticism that model theory knows how to proceed from denotation to inference (namely via truth and truth preservation), whereas proof theory has only inference without denotation. Nonetheless it must be emphasized that this position of proof-theoretic semantics is nominalistic in the sense that the normal forms of expressions are still expressions. These fundamental ontological problems of proof-theoretic semantics, which are particularly interesting in relation to Brandom’s approach, who has a different theory of how to obtain denotation from inference, cannot be discussed here.

2.3 Constructivism and anti-realism

Inferentialism and the ‘meaning-as-use’ view of semantics is the broad philosophical framework to which proof-theoretic semantics belongs. This general philosophical and semantical perspective merged with constructive views which originated in the philosophy of mathematics, especially in mathematical intuitionism. Mathematical intuitionism as founded by Brouwer has strong bearings on the philosophy of logic. It is a variant of mathematical constructivism, which claims that, when proving an existential statement, I have to prove a particular instance of it, and when proving a disjunctive statement, I have to prove a particular disjunct. This excludes certain in-

⁵See especially Brandom (2000), where the relationship to Dummett’s and Gentzen’s approaches is expressed very clearly, though no formal proof-theoretic semantics is developed.

direct modes of proof such as classical *reductio ad absurdum* and in particular leads to the rejection of the excluded middle as a universal principle which can be assumed in proofs, for example in the form of the classical dilemma: If C is entailed by both A and $\text{not}A$, then C holds outright, without knowledge of which alternative A and $\text{not}A$ is supposed to hold. Nearly all variants of proof-theoretic semantics are intuitionistic in spirit. This has to do with the fact that the main tool of proof-theoretic semantics, the calculus of natural deduction, is biased towards intuitionistic logic, in the sense that the straightforward formulation of its elimination rules is the intuitionistic ones, whereas classical logic is only available by means of some rule of indirect proof, which destroys the symmetry of the reasoning principles to some extent. If one adopts the standpoint of natural deduction, then intuitionistic logic is a natural logical system. But even from other starting points, for example the logic of rules such as developed by Lorenzen and others, intuitionistic systems are arrived at in a very natural way. More precisely, it is minimal logic which is distinguished in that way, whereas the genuine intuitionistic rules with their characteristic absurdity principle ('*ex falso quodlibet*') require a specific interpretation of negation in terms of implication and absurdity, and a particular interpretation of absurdity, which is not absolutely straightforward (van Atten).

The intuitionistic viewpoint has led Dummett to the metaphysical position of anti-realism. This is due to his view that the rejection of the excluded middle and therefore of bivalence⁶ brings with it the rejection of there being a reality independent of our ability of conceiving it. The view of such an independent reality is the metaphysical counterpart of the view that certain sentences (which purport to refer to this reality) are either true or false independent of our means of recognizing this semantical fact. Dummett is the central figure in claiming this link between semantics and metaphysics. He himself credits the insight into this link (not the rejection of bivalence, of course) to Frege.

Since Dummett the major part of proof-theoretic semantics is associated with the key term 'anti-realism'. This also brings with it the epistemological stance of proof-theoretic semantics. Anti-realism is based on the fact that we cannot access a transcendent reality described by sentences which are either true or false independent of our means of recognizing it. This leads to the view that classical truth-condition semantics, as biased by a realist epistemology, is ontologically naive, whereas the proposed alternative is based on the priority of epistemology over ontology. This is why in these approaches to semantics the notion of 'knowing' and 'knowledge' plays such a prominent role. Ultimately, the meaning of the logical constants has to be explained in terms of knowledge and our ways of achieving it rather than in terms of subject-independent truth.

⁶I cannot discuss here the differences between these principles

This gives this epistemology a particularly verificationist flavour. Knowing a fact is conceived as verifying a sentence describing this fact. This leads to replacing truth conditions by verification condition. In fact, proofs are considered as the epistemic vehicle which establishes a proposition by verifying it, which means that proofs are essentially viewed from a justificationist perspective. This is at least the prevailing view — there are alternative conceptions which do not view proofs in that way and which give room for alternative conceptions of rationality, for example the Popperian one (see sections 5.1, 5.2).

It should be mentioned that ‘anti-realism’ is a negative description. Adherents of this view hesitate to use the term ‘idealism’, which is normally considered the opposite to ‘realism’. This makes them differ from the founding father of intuitionism, Brouwer, who held idealist views. We do not discuss here in detail the sophisticated arguments supporting anti-realist positions. Tennant’s (1987, 1992, 1997) books are excellent introductions to this area in general, and at the same time present and defend a specific form of anti-realism which combines intuitionism with basic ideas of relevant logic. Another good source is Tranchini (2011b).

2.4 Intuitionistic logic and its semantics

As in most of its variants, proof-theoretic semantics is intuitionistic in spirit, it bears a significant relationship to other intuitionistic approaches to semantics. To that belong both (i) intuitionistic and (ii) classical treatments of the semantics of intuitionistic logic.

(i) Concerning the intuitionistic conceptions, the BHK (Brouwer-Heyting-Kolmogorov) interpretation of the logical signs plays a particular role. This interpretation is not a unique approach to semantics, but comprises various ideas which are often more informally than formally described. Proof-theoretic semantics such as Prawitz’s definition of validity (see section 4.2.2) can be subsumed under it. The BHK interpretation characterizes the logical constants in terms of a notion of ‘proof’, where a proof of a logically compound sentence is built up in a certain way from the proofs of its components, starting from atomic proofs, i.e., proofs of atomic sentences. In the propositional case, the fundamental constructions are pairing and constructive transformation. Following the presentation in Troelstra and van Dalen (1988), the clauses given for conjunction, disjunction and implication can be stated as follows⁷:

- A proof of $A \wedge B$ is a pair consisting of a proof of A and a proof of B
- A proof of $A \vee B$ is a pair $\langle a, A \vee B \rangle$, where a is a proof of A or a proof of B

⁷My wording. For a thorough discussion of the development of intuitionism and the BHK interpretation see Atten (2009). The pairing of a with $A \vee B$ in the clause for disjunction is necessary for a proof of A or of B to be considered as evidence for the compound proposition $A \vee B$.

- A proof of $A \rightarrow B$ is a constructive function which, when applied to a proof of A yields a proof of B

The notion of ‘proof’ is here understood in a very general way without reference to a specific formal system, so the term ‘proof’ could as well be replaced with the term ‘construction’. Apart from proof-theoretic semantics as investigated below, where proofs are represented by means of derivations in a natural deduction system, BHK semantics comprises Kleene’s (1945) realizability interpretation (see Troelstra & van Dalen, 1988 [section 4.4], Moschovakis, 2010), where constructions are represented as natural numbers (including constructive transformations, which are recursive functions represented by their indices), abstract interpretations in terms of typed λ -terms (see Prawitz Constructive validity) or in the sense of Kreisel (1962) and Goodman (1970), and even Gödel’s (1958) functional interpretation (although the latter brings with it, of course, additional ideas). In spite of its generality, the BHK interpretation conveys the basic tenets of intuitionistic semantics. For example, for a proof of a disjunction it is not only expected that there be a proof of one of the disjuncts, but that the information be given for which disjunct the proof is available. Moreover, and even more importantly, a proof of an implication is understood as a constructive transformation. It therefore transcends the combinatorial pairing operation by invoking a different concept, that of a (constructive) function, which links the (constructions of) the antecedent and the consequent of the conditional. This is entirely different from the classical reading of the conditional, which, as expressing a truth function, is not less ‘combinatorial’ than conjunction and disjunction. This functional reading of implication is a fundamental feature of most of the ‘standard’ conceptions of proof-theoretic semantics, in particular of the Dummett-Prawitz style. There the intention is to give the idea of a constructive procedure a more formalistic underpinning by understanding it as a reduction procedure for proofs which are represented by formal derivations. Although we do not discuss this issue here, it should be mentioned that in the quantified case, the functional view also applies to the universal quantifier, where the argument of the function is now an element from which a proof of a certain instance of a proposition is generated. (Incidentally, this view gives much credit to restricted quantification (as in “all P are Q ”), as the type of the function input needs to be indicated.)

(ii) Treatments of the semantics of intuitionistic logic within a classical framework are relevant to intuitionistic proof-theoretic semantics as well, even though they cannot be used for the foundational purposes often pursued by it. The most prominent approach here is Kripke semantics (see Troelstra & van Dalen, 1988, section 2.5, Moschovakis, 2010), which interprets propositions using partially ordered sets of ‘worlds’ or ‘reference points’, which may be viewed as representing our evolving knowledge. This approach is classical not only as it uses classical set theory and logic in the metalanguage, but also as it is built on the classical indexical view of intensions:

Propositions, which in intuitionism have to be understood intentionally, are evaluated in an indexed set of classical structures and in this sense are derived from classical extensions. Kripke uses the (since Gödel, 1933) well-known embedding of intuitionistic logic into the modal logic S4 to obtain from his indexical semantics of modal logic a corresponding semantics of intuitionistic logic. There is nevertheless room for inspiring proof-theoretic semantics to investigate some so far unexploited directions. In Prawitz-style semantics one starts from atomic proof systems and defines the semantics of complex sentences with respect to these systems. When giving the semantics of implication, one normally considers arbitrary extensions of such atomic systems (see below section 4.2.2). This is highly reminiscent of the clause for implication in Kripke semantics, where an implication is evaluated in all subsequent worlds. It is worthwhile to consider these extensions to be structured like a Kripke frame, which would make it possible to apply methods used in Kripke semantics to study extension structures. Although this idea has not been pursued (and, to my knowledge, not mentioned) by Prawitz, it is an interesting line of research. It would use the interpretation of implication in Kripke semantics, which is indexical rather than functional, by using atomic systems as indices. More speculative is the idea to gain novel insight into the modal necessity operator from properties of tree structures investigated in proof-theoretic semantics (see the remark in Kreisel, 1971, p. 164).

2.5 Gentzen-style proof theory: Reduction, normalization, cut elimination

Gentzen's calculus of natural deduction and its rendering by Prawitz is the background to most approaches to proof-theoretic semantics.⁸ Natural deduction is based on at least three major ideas:

Discharge of assumptions: Assumptions can be “discharged” or “eliminated” in the course of a derivation, so the central notion of natural deduction is that of a derivation *depending on assumptions*.

Separation: Each primitive rule schema contains only a single logical constant.

Introductions and eliminations: The rules for logical constants come in pairs. The introduction rule(s) allow(s) one to infer a formula with the constant in question as its main operator, the elimination rule(s) permit(s) to draw consequences from such a formula.

In Gentzen's natural deduction system for first-order logic derivations are written in tree form and based on the well-known rules (see appendix A). In the following, we mainly deal with propositional logic, and here essentially with the rules for implication,

⁸Besides Gentzen's (1934/35) original presentation and Prawitz's (1965) monograph, Tennant (1978), Troelstra and Schwichtenberg (1996), and Negri and von Plato (2001) can be recommended as references.

which represent the crucial case with the introduction and elimination rules

$$\frac{[A]}{\frac{B}{A \rightarrow B} \rightarrow \text{I}} \qquad \frac{A \rightarrow B}{B} \frac{A}{A} \rightarrow \text{E}$$

The system predominantly considered in proof-theoretic semantics is that for intuitionistic logic. For classical logic one would have to add some axiom or rule which does not fit very neatly within the pattern given above, such as *tertium non datur* $A \vee \neg A$ or double negation elimination

$$\frac{\neg \neg A}{A}$$

which is often written as the *classical reductio* rule

$$\frac{[\neg A]}{\perp} \frac{\perp}{A}$$

with $\neg A$ standing for $A \rightarrow \perp$. Some approaches to proof-theoretic semantics for classical natural deduction are described in section 5.5.

For the definition of validity, the notion of closed and open derivations and of reduction and normalization are crucial. The *open assumptions* of a derivation are the assumptions on which the end-formula depends. A derivation is called *closed*, if it has no open assumption, otherwise it is called *open*. If we deal with quantifiers, we have to consider open individual parameters, too.

Metalogical features crucial for proof-theoretic semantics and for the first time systematically investigated and published by Prawitz (1965) include:

Reduction: For every detour consisting of an introduction immediately followed by an elimination there is a reduction step removing this detour.

Normalization: By successive applications of reductions, derivations can be transformed into normal forms which contain no detours.

For implication the reduction step is as follows:

$$\frac{\frac{A}{\mathcal{D}} \quad \frac{B}{A \rightarrow B} \quad \mathcal{D}' \quad A}{B} \quad \text{reduces to} \quad \frac{A}{\mathcal{D}} \quad B$$

For other connectives and further details see appendix A. Here we just mention a simple, but very important corollary: *Every closed derivation in intuitionistic logic can be reduced to a derivation using an introduction rule in the last step.* We also say that intuitionistic natural deduction satisfies the “*introduction form property*”. In proof-theoretic semantics this result figures prominently under the heading “*fundamental assumption*” (Dummett, 1991, p. 254). The “*fundamental assumption*” is a typical example of a philosophical re-interpretation of a technical proof-theoretic result.

Sequent calculi as the second type of Gentzen-style systems are mostly, but not exclusively used in conceptions of proof-theoretic semantics, where a specific handling of assumptions is of particular significance (see section 4.3.2), and in proof-theoretic characterizations of logical constants which are not always intended as semantics (see section 4.4.1). Some of their central features are analogous to those for natural deduction, others differ considerably from them. In considering sequents $\Gamma \vdash A$ with a list (or multiset or set) Γ of formulas forming the antecedent and a formula A forming the succedent, and in allowing that formulas may disappear in the antecedent in the application of a rule such as that of \rightarrow introduction in the succedent:

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \rightarrow B},$$

the sequent calculus has a feature corresponding to the discharge of assumptions, the only difference being that assumptions are now notated as antecedents at every inference line and not as topmost formulas. Furthermore, primitive rules for the different logical operators are separated from one another as in natural deduction, so the separativity feature is shared with natural deduction as well. However, instead of introduction and elimination rules we have now rules for the introduction of a compound formula either in the succedent or in the antecedent rather than introductions and eliminations. In the case of implication, these rules are:

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \rightarrow B} \quad \frac{\Gamma \vdash A \quad B, \Delta \vdash C}{\Gamma, \Delta, A \rightarrow B \vdash C}$$

Introducing a connective in the antecedent is a fundamentally new idea, which is, e.g., exploited in proof-theoretic semantics based on definitional reflection (section 4.3.2). The role of normalization is now taken over by the elimination of cuts

$$\frac{\Gamma \vdash A \quad \Delta, A \vdash B}{\Gamma, \Delta \vdash B}$$

which again relies on specific reduction procedures. For further particulars see appendix A.

A special feature of sequent-style systems is that they make it easy to deal with substructural distinctions, i.e., ways of structuring assumptions. Such ways can be expressed by choosing an appropriate algebraic structure for the antecedent of a sequent and appropriate rules governing this structure. We might, for example, consider contraction-free systems, in which the structural rule

$$\frac{\Gamma, A, A \vdash B}{\Gamma, A \vdash B}$$

is lacking. Such features are difficult to handle in ordinary natural deduction, although they can be treated in ‘sequent-style’ natural deduction, in which one deals with sequents $\Gamma \vdash A$, but with introductions and eliminations rather than introductions in the antecedent and succedent (see appendix A).

Sequent calculi permit a smooth treatment of classical and other symmetric substructural logics when considering succedents with more than one formula (see section 5.4). This feature makes them more congenial to classical logic than natural deduction, which is biased towards intuitionism, if one does not want to introduce inference rules with multiple conclusions. However, this formal feature has not yet lead to philosophically fully convincing systems of proof-theoretic semantics for classical logic (see section 5.5).

2.6 Proofs, terms and their computability

According to the so-called “Curry-Howard correspondence” (see de Groote, 1995; Sørensen & Urzyczyn, 2006), derivations in the intuitionistic propositional calculus can be represented by terms, which have the sentence derived as their type. In extended form this leads to Martin-Löf type theory, where proofs as ontological objects are distinguished from demonstrations which prove that a certain object has a certain type (see section 4.2.4). Due to this correspondence, ideas and results concerning the typed λ -calculus have an immediate bearing on the handling of derivations. For proof-theoretic semantics, Tait’s (1967) method of proving normalization of terms is of particular relevance, as it has a strong semantic flavour and can be viewed as the main technical starting point of Prawitz definition of validity (see section 4.2.2). It consists of defining a semantics-like predicate of terms and then showing, in a kind of correctness proof, that every derivation satisfies this predicate, yielding as a corollary that every derivation is normalizable. Tait (1967) called this predicate “convertibility” and used it to demonstrate (weak) normalizability of terms. Martin-Löf (1971) carried Tait’s idea over from terms to derivations, defining a corresponding predicate which he called “computability” and proving (weak) normalization for the theory of iterated inductive definitions (an extension of first-order logic). At the same time, Girard (1971) used this method to prove (weak) normalization for second-order logic. Again at the same time Prawitz (1971) used it for proving *strong* normalization, calling it “strong validity”. Since then, it has served as the basis of proofs of strong normalization for a variety of systems (see also Joachimski & Matthes, 2003). We shall adopt Martin-Löf’s terminology and speak of “computability”. Prawitz’s (1971) term “valid” (which Prawitz himself uses both in proofs of normalization and in the semantical sense, see 1971, pp. 284, 290) will be reserved for genuinely semantical notions.

For implicational logic, computability can be defined as follows:

- (i) A derivation in introduction form
$$\frac{[A] \quad \mathcal{D}}{B} \quad A \rightarrow B$$
 is computable, if for every computable

derivation $\frac{\mathcal{D}'}{A}$ of A , the composed derivation, $\frac{A}{\frac{\mathcal{D}}{B}}$ is computable.

- (ii) If a derivation \mathcal{D} is not in introduction form and is normal, then it is computable.
- (iii) If a derivation is not in introduction form and is not normal, then \mathcal{D} is computable, if every derivation \mathcal{D}' to which \mathcal{D} immediately reduces, is computable.⁹

The semantical connotations of this predicate are obvious. By showing that every derivation is correct in the sense that it is computable, and then using the fact that every computable derivation is strongly normalizable (the latter being a nearly immediate consequence of the definition), one obtains strong normalization¹⁰. For the comparison with the definition of validity in Prawitz-style proof-theoretic semantics (section 4.2.2) it should be noted that in the definition of computability,

1. derivations in introduction form are considered primary,
2. for derivations not in introduction form, the result of applying a reduction step is significant,
3. assumptions which are discharged are interpreted as placeholders for computable derivations
4. the notion of normal form is presupposed
5. *closed* derivations do *not* play a distinguished role,

Points 1 to 3 are shared with the semantic notion of validity, whereas points 4 and 5 differ from it. Nevertheless, besides Prawitz's (1965) discussion of normalizability, Tait's (1967) contribution was the key technical concept for a precise semantical reading of natural deduction proofs.

2.7 ‘Tonk’, the Belnap criteria and other constraints

The most primitive form of the “meaning as use” paradigm with respect to logical operators is the claim that an arbitrary set of rules of whatever form implicitly determines the meaning of logical operators occurring in them. In this sense we would just have to stipulate certain rules to hold of an operator in order to equip it with meaning. In a holistic version several constants would then be defined together by one

⁹This is a generalized inductive definition. It uses induction on the degree of the end formula of the derivation (clause i), and, within each degree, induction on the reducibility relation (more precisely: induction given by the operator associating with a set of derivations X of a formula the set of those derivations which reduce in one step to a derivation in X) (clauses ii and iii). — Other definitions of computability rely on elimination rather than introduction rules as their base, see section 5.1.

¹⁰It should be noted that, as far as strong normalization as a technical result is concerned, there is also a ‘combinatorial’ proof, which does not rely on the computability predicate, see Zimmermann (2010).

and the same set of rules, whereas of a more constructive variant we would expect that such a set of meaning-giving rules must be separable into disjoint subsets which can be well-ordered in the following way: Every subset defines the meaning of one particular constant, where the meanings of other constants used in the definition are defined by rule sets preceding this set in the well-ordering. We call the latter condition the *well-foundedness requirement*.

However, for semantical purposes it is not sufficient just to lay down rules. As Prior (1960) has argued, admitting arbitrary rules as (allegedly) meaning determining can easily result in implausible, and in the worst case inconsistent, results, even though the constructive well-foundedness requirement is met. As an example Prior presents his operator *tonk* which combines an \vee -introduction with an \wedge -elimination rule:

$$\frac{A}{A \text{ tonk } B} \quad \frac{A \text{ tonk } B}{B}$$

allowing one to derive any B from any A . Prior considered this a decisive argument against inferentialism as such. However, one might instead impose certain restrictions on rules if they should qualify as meaning-giving rules in a reasonable way.

Belnap (1962) proposed to require *conservativeness* and *uniqueness* in the following sense.

- The meaning-giving rules for a constant α should not allow one to prove something which can be formulated in the vocabulary available prior to the definition of α (conservativeness). Obviously, this is violated in the case of *tonk*: If A and B stand for different atomic sentences, and no specific inference rules are presupposed for atoms, then the detour via $A \text{ tonk } B$ generates $A \vdash B$, which cannot be obtained without the *tonk*-rules.
- α should be characterized in such a way that, if we duplicate its rules by adding identical rules for a connective α^* , any formula $A(\alpha)$ containing α (but not α^*) should be interdeducible with the formula $A(\alpha^*)$, in which α is replaced with α^* (uniqueness). In particular, the interderivability statement $\alpha \dashv\vdash \alpha^*$ must hold. This is again violated in the case of *tonk*, since, if to the definition of *tonk* we add that of *tonk*^{*}:

$$\frac{A}{A \text{ tonk}^* B} \quad \frac{A \text{ tonk}^* B}{B}$$

we are not able to derive $A \text{ tonk } B \dashv\vdash A \text{ tonk}^* B$.¹¹

¹¹This claim holds if the schematic letters in the definition of *tonk* refer to expressions in the *tonk*-language and the schematic letters in the definition of *tonk*^{*} refer to expressions in the *tonk*^{*}-language. If we consider a joint language containing both *tonk*- and *tonk*^{*}-expressions, then uniqueness is trivially satisfied, as any expression is interdeducible with any other expression. For further investigations on the notions of conservativeness and uniqueness see Došen & Schroeder-Heister (1985, 1988).

Conservativeness is weaker than eliminability of definienda by definienda which, besides uniqueness, is normally expected to hold of explicit definitions. For example, it also covers inductive definitions. For each combination of conservativeness/non-conservativeness with uniqueness/non-uniqueness, connectives with appropriate rules can easily be designed. For the implicational calculus, there is an obvious close relationship between β -reduction and conservativeness and between η -reduction and uniqueness. β -reduction is related by the Curry-Howard correspondence to reduction procedures in derivations, thus to normalization, and therefore to conservativeness. η -reduction says that by application followed by abstraction, we can obtain back what we have started with, which is the sort of reasoning used to establish uniqueness.

It is tempting to base proof-theoretic semantics on the Belnap criteria of conservativeness and uniqueness. However, according to our understanding of it, these criteria are too wide in one respect and too narrow in another, if one wants to delineate semantical rules. Of course, meaning-giving rules should be conservative in any case. But uniqueness is too narrow a criterion. Later on we shall consider *relative* notions of uniqueness of the sort “The definiendum is uniquely defined *given* the definiens is unique, even if the definiens cannot be proven unique”. This makes it possible to also investigate partial notions of meaning (see section 4.3.2). On the other hand, conservativeness and uniqueness are also too wide to justify rules as meaning-giving, as they do not impose any restriction concerning the *form* of rules. In all variants of proof-theoretic semantics presented in the following, the constant whose meaning is to be explained, occurs in a special position, normally as the *conclusion* of a rule, which may be a production rule in an atomic system, an introduction rule in natural deduction, or a definitional clause in theories of definitional reflection. In a dual approach based on eliminations, the constant may occur in *premiss* position. But again, it occurs at a special place. We call this the *special form requirement*. Furthermore, the constant in question occurs as the major operator of a proposition, i.e. is not embedded within other connectives or quantifiers, not even with the defined operator itself (i.e., when defining conjunction \wedge , we would not permit rules containing iterated conjunctions of the form $A \wedge (B \wedge C)$ etc. This will be called the *explicitness requirement*. The *special form* and *explicitness* requirements make proof-theoretic semantics in our sense strongly differ from rule systems which just guarantee conservativeness and uniqueness.

As mentioned above, conservativeness should hold in any case. However, in natural deduction or sequent systems satisfying special form and explicitness constraints, conservativeness can be expressed by more specific properties which refer to the exact form of rules considered. For example, the features of normalization or of admissibility of cut may replace the conservativeness option in such systems. By considering different notions of cut elimination or normalization, more fine-grained notions can be considered. One possibility we shall deal with in section 4.3.2 is *relative cut*: Cut holds

for a definiendum given it holds for its definiens, without necessarily being provable for the definiens¹². So our general picture is the following: For meaning explanations we expect *special form* and *explicitness* requirements (constants at certain positions in defining rules, and not nested) to be satisfied. *Conservativeness* has to hold, but will be investigated in close relationship with notions of cut elimination and normalization. Constructivity (well-foundedness of sets of definitional rules) is not required, but may of course be studied as a useful property of particular systems. These are, of course adequacy conditions, which are at best necessary, but never sufficient.

A special remark applies to *separation*, which is very often considered to be mandatory for semantical rules. Separation means that every constant has rules of its own, so a rule for a specific constant must not refer to any other constant. It holds in intuitionistic natural deduction, but is violated in the classical system based on implication and absurdity, since the rule of classical reductio ad absurdum

$$\frac{[A \rightarrow \perp]}{\perp}$$

mixes implication with absurdity, something which is sometimes used as an argument against classical logic. Separation is often combined with particular special form requirements, for example the requirement that a constant must occur only in the conclusion of (the schema for) its introduction rule and only at a specific place in the premiss of (the schema for) its elimination rules. Insisting on full separation would be extremely restrictive. For example, a ternary logical constant α with the meaning $A \rightarrow (B \vee C)$ could never be defined, although

$$\frac{A \rightarrow (B \vee C)}{\alpha(A, B, C)} \quad \frac{\alpha(A, B, C)}{A \rightarrow (B \vee C)}$$

would be a perfectly legitimate set of introduction and elimination rules for α , with \rightarrow and \vee being defined by their common rules. Here the rules for α cannot be separated from those for \rightarrow and \vee (at least not in an intuitionistic framework). What we do have, of course, is the special form and explicitness requirements that α occurs only once in each rule schema, and only at a specified position, and in addition well-foundedness in that α is defined in terms of \rightarrow and \vee . Separation in its strict sense implies that any constant is definable directly in structural terms, without intermediate step involving other constants, i.e., the definitional chain starting from the defined expression has exactly one step. This goes way beyond well-foundedness, which says that after *finitely many* steps we reach the structural level. What is often intended instead by the

¹²This does not mean that the global meaning of conservativeness, viz. that nothing new is provable in the old vocabulary, is given up. In systems with cut, the eliminability of cut guarantees conservativeness. In systems without the cut rule, conservativeness holds anyway.

separation requirement, is the exclusion of a mutual dependency: It is not excluded that the definition of α depends on β , where β is independent of α , but only that α and β mutually depend on one another, so that none of them can be reduced to the other. This notion, however, is captured by our notion of well-foundedness. When considering conceptions of proof-theoretic semantics which give up well-foundedness (see section 4.3.2), we also consider those which give up separation.

3 Local versus global proof-theoretic semantics

There is a fundamental distinction, or better: one should make a fundamental distinction, as it is not well recognized yet, between *local* and *global* approaches to proof-theoretic semantics. This is the distinction of whether rules or proofs are more elementary with respect to one another. According to one approach, rules come first in the order of explanation or justification. Once we have established certain rules as valid, we can consider a proof to be valid if it consists of applications of valid rules. This approach is local as it is the individual rules of which a proof is composed which are valid in the primary sense. According to the alternative approach, proofs come first. Once we have established a class of proofs as valid, we can consider a rule to be valid, if it does not properly extend this class of valid proofs, that is, if it leads from valid proofs to valid proofs. This approach is global as it is the whole proof which is the primary bearer of validity.

Presented in this abstract way, one would presumably find the local approach more appropriate, as it proceeds from the simple entities (rules) to the more complex ones (proofs) out of which they are composed. However, many approaches to proof-theoretic semantics proceed the other way, i.e., are global approaches. This holds in particular of all approaches that can be subsumed under the BHK label. There an implication and therefore also a rule is considered to be valid, if it leads from a valid proof to a valid proof, where this “leading to” can be described by a constructive transformation. All ‘functional’ approaches to consequence and, in particular, notions of proof-theoretic validity and related theories are of that kind (see section 4.2.2, 4.2.4). Of course, in order to speak of a proof in the first place one needs certain initial rules which construct a proof in any case. These are rules which are sometimes called “self-justifying”, and for which in most cases the introduction rules are considered. However, the basic definition built on that base would then be the validity of derivations.

The advantage of the local approach is that it can dissociate the validity of the individual rules from the way they are composed into a complex derivation. Individual rules can be justified without the whole proof having certain distinctive features. This is important if one wants to deal with phenomena, e.g., of well-foundedness. Given non-wellfounded introduction rules, it will be possible, in the theory of definitional reflection (see section 4.3.2), to define corresponding elimination rules, which are in

perfect harmony with the introduction rules, without derivations as a whole satisfying certain normal form requirements. So the validity of individual rules does not presuppose anything about the proofs composed out of them.

In the following, it will always be mentioned to which branch (global versus local) the approach considered belongs. We have not classified them here according to this distinction, although that would be very appropriate, too, as this distinction is, in our view, absolutely fundamental. We do not want to conceal that we prefer the local approaches to the more standard global one as they are more versatile when it comes to semantics beyond logical constants in the narrower sense.

4 Some versions of proof-theoretic semantics

4.1 The semantics of implications: Admissibility, derivability, rules

The semantics of implication lies at the heart of proof-theoretic semantics, since, in contradistinction to classical truth-condition semantics, implication is a logical constant in its own right and genuinely different from constants like conjunction and disjunction. Already in intuitionistic semantics it played a special role (see section 2.4). Something similar holds for universal quantification, which behaves similarly to implication, and negation, which is intuitionistically understood as implication to absurdity. Implication has also the characteristic feature that it is tied to the concept of consequence. It can be viewed as expressing consequence at the sentential level due to modus ponens and to what in Hilbert-style systems is called the deduction theorem, i.e. the equivalence of $\Gamma, A \vdash B$ and $\Gamma \vdash A \rightarrow B$. Therefore, for every conception of proof-theoretic semantics the interpretation of implication is crucial.

A very natural understanding of an implication $A \rightarrow B$ is reading it as expressing the inference rule which allows one to pass over from A to B . Licensing the step from A to B on the basis of $A \rightarrow B$ is exactly, what modus ponens says. And the deduction theorem can be viewed as the means of establishing a rule: Having shown that B can be deduced from A justifies the rule that from A we may pass over to B . A rule-based semantics of implication along such lines underlies several conceptions of proof-theoretic semantics, notably those by Lorenzen, Kutschera and Schroeder-Heister. They all start with a fundamental semantics of implication and embed all further semantics into it. In Lorenzen this fundamental semantics is the admissibility interpretation of implication, in Kutschera the theory of iterated S -formulas, and in Schroeder-Heister the concept of rules of higher levels.

4.1.1 Lorenzen’s operative logic

Although Lorenzen’s *Introduction to Operative Logics and Mathematics* (1955)¹³ is formalistic in spirit, basing logical and mathematical reasoning on ‘operating’ with symbolic figures (in this sense being related to Curry’s approach), it contains a semantics for the logical constants even if not intended and designated as such. In dealing with logical constants, Lorenzen is concerned with the *justification* of the inference rules governing them, arriving at a formalism of intuitionistic first-order logic. This justification is a sort of proof-theoretic semantics in our sense as it establishes the *correct use* of those constants by *reflecting on proofs* and not, as one would expect from a formalist, by just laying down the set of intended inference rules. To be sure, the proofs Lorenzen is reflecting upon are formal proofs, i.e. derivations, and the logical rules being justified are formal rules as well. However, the way through which he arrives at these logical rules is semantical reasoning in our sense of the word, based on principles for the validation of rules. In fact, a crucial rule is played by an *inversion principle*, which similar to Prawitz’s one justifies elimination inferences from introduction inferences. It was actually Lorenzen who coined the term *inversion principle* for a more general claim not confined to logical constants, which was then later borrowed by Prawitz for use in the context of natural deduction (see section 4.2.2).

Lorenzen starts with logic-free (atomic) calculi, which correspond to production systems or grammars. He calls a rule *admissible* with respect to such a system if it can be added to it without enlarging the set of its derivable atoms¹⁴. Therefore, if \vdash_K denotes derivability in a calculus K , then the rule $a_1, \dots, a_n \rightarrow a$ is admissible in K , if $\vdash_K a_i$ for all i ($1 \leq i \leq n$) implies $\vdash_K a$, which must be distinguished from the derivability of a from the assumptions a_1, \dots, a_n , which is denoted by $a_1, \dots, a_n \vdash_K a$.

The implication arrow \rightarrow is identified with the rule arrow used in stating a production rule, and the derivability of an implication is interpreted as an admissibility statement. If A, B_1, \dots, B_n stand for lists of atoms and a, b_1, \dots, b_n are atoms, then $\vdash_K A \rightarrow a$ means that the rule $A \rightarrow a$ is admissible in K , and $B_1 \rightarrow b_1, \dots, B_n \rightarrow b_n \vdash_K A \rightarrow a$ means that the rule $A \rightarrow a$ is admissible in the calculus which results from K by adjoining $B_i \rightarrow b_i$ ($1 \leq i \leq n$) as additional rules. For iterated implications such as $(A \rightarrow b) \rightarrow a$ Lorenzen develops a theory of admissibility statements of higher levels, which cannot be presented here (see Schroeder-Heister, 2008a). Certain statements such as $\vdash_K a \rightarrow a$ or $a \rightarrow b, b \rightarrow c \vdash_K a \rightarrow c$ hold independently of the underlying system K . They are called *universally admissible* [“allge-

¹³The points relevant to our discussion of proof-theoretic semantics, including the inversion principle, are already presented in Lorenzen (1950).

¹⁴In contradistinction to Lorenzen, and following the terminology of logic programming, by *atoms* we denote the formulas of an atomic system, i.e., its *words* in the terminology of grammars. We make no assumption about the syntax of atoms. (In Lorenzen they are lists of symbols.)

meinzulässig”]), and constitute a system of positive implicational logic.

This represents a proof-theoretic semantics for implication. Implications express admissibility statements with respect to formal systems, and logically valid implications express admissibility statements which hold for *any* atomic system. In a related way, laws for universal quantification \forall are justified using admissibility statements for rules of the form $A \rightarrow_{x_1, \dots, x_n} a$ with schematic variables x_1, \dots, x_n as indices (with premiss-free rules $\rightarrow_{x_1, \dots, x_n} a$ as a limiting case). Obviously, the reading of implications as admissibility statements can be viewed as a variant of the BHK-interpretation, as it is a sort of functional view of implication: In order to establish the admissibility of the rule $A \rightarrow a$, we have to give a constructive procedure eliminating applications of $A \rightarrow a$, which is a constructive procedure transforming derivations of the premisses A into a derivation of the conclusion a .

For the justification of the laws for the logical constants \wedge , \vee , \exists and \perp , Lorenzen proceeds by first deducing a list of general principles for establishing admissibility, the crucial one being his *inversion principle*. In a very simplified form, without taking variables in rules into account, this inversion principle says the following. Let A, A_1, \dots, A_n be lists of atoms, and a, b atoms. Suppose

$$\begin{array}{l} A_1 \rightarrow a \\ \vdots \\ A_n \rightarrow a \end{array}$$

are the only rules by means of which a can be derived in a calculus K . Then the rule $a \rightarrow c$ is admissible in the calculus which results from K by adjoining

$$\begin{array}{l} A_1 \rightarrow c \\ \vdots \\ A_n \rightarrow c \end{array}$$

as primitive rules, i.e., we have $A_1 \rightarrow c, \dots, A_n \rightarrow c \vdash_K a \rightarrow c$. Roughly speaking: Everything that is implied by each condition of a is implied by a itself. This principle is used for the justification of logical inference rules as follows:

Let *disjunction* be defined by a pair of rules

$$\begin{array}{l} a \rightarrow a \vee b \\ b \rightarrow a \vee b \end{array}$$

Then the inversion principle says that $a \vee b \rightarrow c$ is admissible assuming $a \rightarrow c$ and $b \rightarrow c$, i.e., $a \rightarrow c, b \rightarrow c \vdash a \vee b \rightarrow c$, which justifies the elimination rule for disjunction.

Similarly, for *conjunction*, which is defined by

$$a, b \rightarrow a \wedge b,$$

we can argue that $a \wedge b \rightarrow c$ is admissible assuming $a, b \rightarrow c$, i.e., $a, b \rightarrow c \vdash_K a \wedge b \rightarrow c$. Since $a, b \rightarrow a$ and $a, b \rightarrow b$ are trivially admissible, this implies that both $a \wedge b \rightarrow a$ and $a \wedge b \rightarrow b$ are admissible, which justifies the common elimination rules for conjunction.

Using rules with bound variables, from the rule

$$a \rightarrow \exists x a$$

the elimination rule for *existential quantification* can be justified in the form of the principle

$$a \rightarrow_x c \vdash \exists x a \rightarrow c$$

(for c not containing x free) in an analogous way.

Finally, the intuitionistic *absurdity rule* is justified by reference to the admissibility of

$$\perp \rightarrow c$$

with respect to any calculus in which \perp is undefined, i.e., in which \perp is not the head of a primitive rule. This might be considered to be a limiting case of the inversion principle (with respect to an empty set of defining rules). Lorenzen formulates it as based on a principle of its own, called the *underivability principle* [“Unableitbarkeitsprinzip”].

It is obvious that Lorenzen’s justification of logical rules is related to Gentzen’s programme of taking introduction rules as definitions of constants and justifying elimination rules with respect to introduction rules by assuming, in each case, that the major premiss of the elimination rule has been derived using one of the introduction rules (see section 4.2.1). However, when basing his notion of implication on the concept of admissibility of rules, Lorenzen relies on a system crucially different from Gentzen’s calculus of natural deduction in that it does not employ the idea of discharging assumptions when introducing implication.

Furthermore, Lorenzen’s approach is much more general than Gentzen’s in that he considers logically compound sentences just as special cases of arbitrary atoms. This means that his inversion principle can be used as a justification of elimination rules from introduction rules for arbitrary atoms. It makes his approach capable of dealing with inductive definitions in general rather than just with introduction and elimination rules for logical constants. He thus anticipates the idea of introduction and elimination rules for atoms which can be found, e.g., in Martin-Löf’s (1971) theory of iterated inductive definitions¹⁵ and in general reflection principles for logic programming. In

¹⁵In contradistinction to Martin-Löf, Lorenzen considers the induction principle as a principle of its own rather than an application of the inversion principle. Martin-Löf subsumes the induction rules under the generalized notion of an elimination rule.

fact, his inversion principle in its general form is closely related to a certain form of the principle of definitional reflection (see section 4.3.2).

Concerning our local/global distinction (section 3), it is clear that Lorenzen's theory is global, as admissibility is a global concept. A rule is admissible if it transforms proofs of a certain class into proofs of the same class. Certain rules are primary which constitute these classes. In the case of disjunction and conjunction these correspond to the common introduction rules, otherwise they can be just arbitrary. But the rules justified out of them as admissible refer to derivations as a whole. That nonetheless Lorenzen's theory has the above-mentioned affinity to rule-based approaches, is due to the fact that one can ignore the admissibility interpretation of his rules.

4.1.2 Kutschera's Gentzen semantics

In what he calls "Gentzen semantics" ("Gentzensemantik") Kutschera gives, as Lorenzen, a semantics of logically complex implicational sentences with respect to calculi K which govern the reasoning with atomic sentences. These calculi are then extended with inferences governing the logical constants. For each calculus K , Kutschera first defines the derivability of implicational formulas, which are either atomic formulas or formulas of the form $(A_1, \dots, A_n \rightarrow B)$ for implicational formulas A_1, \dots, A_n, B . To distinguish them from propositional implications defined later on, he calls them "S-formulas". The derivability of S-formulas in K is defined via a hierarchy of calculi, where the derivability of an S-formula $A_1, \dots, A_n \rightarrow B$ in the $(n + 1)$ th-level calculus expresses the derivability of B from A_1, \dots, A_n in the n th-level calculus. The fundamental difference to Lorenzen is the fact that an S-formula $A_1, \dots, A_n \rightarrow B$ now expresses a *derivability* rather than an *admissibility* statement. In Kutschera, establishing an implication is an insight into the derivability rather than the admissibility of a rule¹⁶. What he has in common with Lorenzen is that implication is related to derivability in atomic calculi. Derivability in atomic systems is the basic notion to which implication is reduced. The idea of iterating this process by constructing higher-level calculi is similar in Lorenzen and Kutschera.

In order to turn this into a semantics of logical constants, Kutschera argues as follows: When giving up bivalence, we can no longer use truth-value assignments¹⁷ to atomic formulas. Instead we can use calculi which may confirm or refute atomic sentences. Refutation is here conceived as the derivation of absurdity from that sentence. Such calculi do not necessarily attach one of the two values to a given sentence. Moreover, since calculi not only generate confirmations or refutations but arbitrary

¹⁶By the derivability of a rule it is meant that its conclusion can be derived *from* its premisses as assumptions. We repeat this here, as in many textbooks the derivability of an inference rule means its admissibility, i.e. the fact that from proofs of its premisses a proof of its conclusion can be generated.

¹⁷We are here dealing with propositional logic only.

derivability relations, the idea is to start directly with derivability in an atomic system, use its extended form (S -formulas of any level), and further extend this system with rules that characterize the logical connectives. For that Kutschera gives general rules for the introduction of n -ary propositional connectives on the right and left side of \rightarrow , yielding a sequent calculus for arbitrary propositional connectives. Kutschera can then show that the connectives so defined can all be expressed by the standard connectives of intuitionistic logic ($\wedge, \vee, \supset, \perp$).

Kutschera's schema for right and left introduction inferences is as follows:

$$\frac{\Gamma \rightarrow \Delta_1}{\Gamma \rightarrow \alpha(A_1, \dots, A_n)} \quad \cdots \quad \frac{\Gamma \rightarrow \Delta_m}{\Gamma \rightarrow \alpha(A_1, \dots, A_n)} \quad \frac{\Gamma, \Delta_1 \rightarrow C \quad \dots \quad \Gamma, \Delta_m \rightarrow C}{\Gamma, \alpha(A_1, \dots, A_n) \rightarrow C}$$

Here α is an n -ary propositional operator, and the Δ_i are lists of S -formulas built up from the schematic letters A_1, \dots, A_n as the only atomic formulas. The left introduction rules are justified by using a non-creativity requirement¹⁸. The Δ_i on the right side of \rightarrow are understood conjunctively, i.e., $\Gamma \rightarrow R_1, \dots, R_k$ is to be read as $\Gamma \rightarrow R_1, \dots, \Gamma \rightarrow R_k$. Kutschera does not consider multiple-formulae succedents.

Apart from the fact that Kutschera's approach is based on derivability from assumptions rather than admissibility, a crucial difference to Lorenzen's approach is that Kutschera defines implication as a propositional connective in terms of the implication arrow \rightarrow , which is available still without the rules for connectives. So his idea is that as our base we have some sort of 'structural' implication (my term), which expresses iterated derivability in a system, on top of which all 'logical' connectives are defined. This means that the structural component of Gentzen's sequent calculus is extended with additional structural means of expression, namely structural implication (in addition to structural conjunction, which is the comma). This structural extension of atomic systems then serves as the definitional basis of logical constants to be defined. This idea goes beyond Gentzen, although in a sense it was already present in Hertz's work (see section 4.1.4). Comparing it with Lorenzen's way of proceeding, it should be noted that one could extend Lorenzen's meta-calculi with explicit introduction rules for a logical connective \supset for implication as well. Obviously, Lorenzen did not see a conceptual necessity to introduce such a notation, as his approach gives him already the implicational calculus he wanted to justify. An attempt to close this gap and to relate Lorenzen's approach to Prawitz's validity concept can be found in (Schroeder-Heister, 2008a).

Overall, Kutschera's theory is local, as it explicitly dissociates itself from the idea of admissibility when justifying individual rules.

¹⁸In Kutschera (1968) this is not specified in detail, but from Kutschera's remarks it is clear that the existence of main reductions is meant which allow one to eliminate cuts.

4.1.3 Schroeder-Heister's extension of natural deduction

One possibility to give a proof-theoretic semantics for logical constants is to present a general schema for inference rules governing a connective and to motivate this schema as semantically significant. Kutschera's approach is of this kind. Another one was developed by Schroeder-Heister (1981, 1984b)¹⁹. Within a programme of developing a general schema for rules for arbitrary logical constants he proposed that a proposition of the form $\alpha(A_1, \dots, A_n)$, where α is an n -ary logical connective, should express what was called the *common content of systems of rules*. Here a system of rules is a list of expressions R_1, \dots, R_m , where each R_i is called a "rule". A rule R is either a formula A or has the form $R_1, \dots, R_n \Rightarrow A$, where R_1, \dots, R_n are themselves rules. In a sense, systems of rules are expressions of a conjunction-implication-calculus (with the comma expressing conjunction and the rule arrow \Rightarrow expressing implication), where implication is iterated only to the left. This left-iteration distinguishes them from Kutschera's S -formulas, which are iterated both to the left and to the right. Instead of systems of rules, of which a single rule is a limiting case, we shall also speak of *conditions*²⁰. They are denoted by capital Greek letters Γ, Δ , with and without indices. The derivability of formulas from conditions is explained in the following way: A can be derived from itself considered as a condition (consisting just of A as a premiss-free rule). If A_1, \dots, A_n have been derived from $\Gamma_1, \dots, \Gamma_n$, respectively, then B can be derived from $\Gamma_1, \dots, \Gamma_n$ together with the rule $A_1, \dots, A_n \Rightarrow B$. This motivates the "rule"-terminology: B can be derived using the rule $A_1, \dots, A_n \Rightarrow B$ (and perhaps further conditions) as an assumption. This gives rise to even more complicated rules, permitting to discharge assumptions. For example, the rule $((A \Rightarrow B) \Rightarrow C) \Rightarrow D$ allows one to pass over from C to D , provided C has been derived using $A \Rightarrow B$ as an assumption. This again means that the assumption $A \Rightarrow B$ may be discharged when passing from C to D . As rules can be discharged at the application of other rules who have an additional level of nesting, we also speak of *rules of higher levels*. This explanation can be extended to rules with individual variables and quantifiers (see Schroeder-Heister, 1984a).²¹

Now the semantical idea of the common content of conditions is defined as follows.

The formula A expresses the common content of conditions $\Delta_1, \dots, \Delta_n$, iff for every condition Γ and every formula C , it holds that

$$\Gamma, A \vdash C \quad \text{iff} \quad \text{for every } i \ (1 \leq i \leq n), \ \Gamma, \Delta_i \vdash C,$$

¹⁹Besides Kutschera (1968) it was stimulated by Prawitz (1979). See also Schroeder-Heister (1981, 1987).

²⁰This is the terminology proposed by Hallnäs in the context of definitional reflection (see below section 4.3.2).

²¹For various systems for the handling of rules see Schroeder-Heister (1987).

i.e., iff from A , together with possible assumptions Γ , everything follows which follows from each of the conditions Δ_i .

Obviously, *common content* means the same as *content*, if there is just one condition Δ available, where “content” is understood as “set of consequences”. The idea that formulas built up by a generalized connective should express the common content of systems of rules is a semantical idea, as it endows such formulas and the corresponding connective with a deductive meaning in terms of rules.

For the standard logical constants this means the following, where conditions, which are lists of rules, are included in braces:

$A \wedge B$ expresses the common content of $\langle A, B \rangle$
 $A \rightarrow B$ expresses the common content of $\langle A \Rightarrow B \rangle$
 $A \vee B$ expresses the common content of $\langle A \rangle$ and $\langle B \rangle$
 \perp expresses the common content of the empty list of conditions (case $n = 0$)

It can then easily be shown that the standard introduction and elimination inferences just express this semantical condition, *i.e.*, they hold iff this condition is fulfilled.

In general, we assume conditions $\Delta_1(p_1, \dots, p_n), \dots, \Delta_m(p_1, \dots, p_n)$, in short $\Delta_1, \dots, \Delta_m$, to be associated with an n -ary logical constant α , where every Δ_i contains no schematic letters beyond the indicated A_1, \dots, A_n . Then we suppose that for all A_1, \dots, A_n ,

$\alpha(A_1, \dots, A_n)$ expresses the common content of the conditions $\Delta_1, \dots, \Delta_m$.

The corresponding introduction and elimination inferences in the general case are

$$\frac{\Delta_1}{\alpha(A_1, \dots, A_n)} \quad \cdots \quad \frac{\Delta_m}{\alpha(A_1, \dots, A_n)} \quad \frac{\alpha(A_1, \dots, A_n) \quad \frac{[\Delta_1]}{C} \quad \cdots \quad \frac{[\Delta_m]}{C}}{C} .$$

As a corollary we obtain operator completeness in the sense that any $\alpha(A_1, \dots, A_n)$ which expresses the common content of certain conditions, can be explicitly defined using the standard connectives \wedge , \vee , \rightarrow and \perp . For example, if $\alpha_1(A_1, \dots, A_4)$ expresses the common content of $\langle A_1 \Rightarrow A_2 \rangle$ and $\langle A_3 \Rightarrow A_4 \rangle$, which means that it can be characterized by the following introduction and elimination rules

$$\frac{\frac{[A_1]}{A_2}}{\alpha_1(A_1, A_2, A_3, A_4)} \quad \frac{\frac{[A_3]}{A_4}}{\alpha_1(A_1, A_2, A_3, A_4)} \quad \frac{\alpha_1(A_1, A_2, A_3, A_4) \quad \frac{[A_1 \Rightarrow A_2]}{C} \quad \frac{[A_3 \Rightarrow A_4]}{C}}{C} ,$$

then $\alpha_1(A_1, A_2, A_3, A_4)$ can be explicitly defined as $(A_1 \rightarrow A_2) \vee (A_3 \rightarrow A_4)$.

It is, of course, easily possible to define other sorts of proof-theoretic semantics for operators with generalized introduction and elimination rules, for example, validity notions in Prawitz's sense (see section 4.2.2). However, this is a different issue and would be missing our point here, which is the semantical justification of both introduction and elimination rules via the notion of common content of conditions. Therefore, although a global interpretation is possible, the approach is intended as a justification of individual rules at a local level, without taking the structure of whole derivations into account.

4.1.4 Digression: Paul Hertz's systems

Hertz was the precursor of Gentzen. Gentzen's sequent calculus is the result of a close study of Hertz's system. In fact, Gentzen's first publication (Gentzen, 1933) deals exclusively with Hertz's formalism. Hertz developed some system of implicational logic that can be viewed as a purely structural system with an iterated implication arrow, without any further logical constant (see Schroeder-Heister, 2002). The approaches discussed here resemble Hertz in so far as they all consider some 'structural' implicational system as basic, on which all the rest is built. The essential conceptual difference between Hertz and Gentzen is that Hertz did not see the necessity to build a system of propositional and quantifier logic on top of his implicational system, whereas Gentzen did exactly that with reducing the basic system to just his structural rules, without any iteration of the sequent arrow (which would correspond to Hertz's implication). The approaches by Lorenzen, Kutschera and Schroeder-Heister discussed in the previous subsections are conceptually 'in between' Hertz and Gentzen, as, in contradistinction to Gentzen, they rely on an elaborated system of structural implication which can be iterated, but, in contradistinction to Hertz, also build a logical system on top of the structural one. Their basic idea is to have a structural system strong enough to characterize logical constants by reference to structural ones. The natural idea that conjunction is structurally expressed by the comma, is extended to that (the connective of) implication is structurally expressed by structural implication. An attempt to deal with generalized structural notions in the spirit of Hertz is Arndt's (2008) logical tomography. Both Hertz's and Arndt's theories are local approaches based on rules, in Arndt's case even in the stricter sense that the rules themselves have a particular local character, dealing with only one occurrence of a connective at a time.

4.1.5 Digression: Generalized elimination rules

Schroeder-Heister's (1984b) generalized elimination rules are related to the general rules which Kutschera gives in his proof-theoretic semantics of connectives (see sections 4.1.3, 4.1.2). The fundamental difference is that Kutschera uses a sequent-style framework and also that Kutschera allows *S*-formulas, which correspond to Schroeder-Heister's

higher-level rules, to be iterated both to the left and right. For a natural deduction style framework, such generalized elimination rules have been proposed by Prawitz (1979). Prawitz's treatment of implication was erroneous, as it did not use any 'structural' sort of implication, to which the connective of implication could be reduced. The general elimination rules proposed by Dyckhoff (1988), Tennant (1992, 2002), Lopez-Escobar (1999) and von Plato (2001) are different from those proposed here. They avoid the higher-level feature in the case of implication, but are not suitable for a general schema for arbitrary connectives. Therefore they have only a limited scope in proof-theoretic semantics. Their essential feature is to model elimination rules according to the left introduction rules of the sequent calculus. For a detailed exposition see Schroeder-Heister (2010c).

4.2 The Semantics of derivations as based on introduction rules

The most influential and also most elaborated approaches to proof-theoretic semantics are those by Prawitz and Martin-Löf. Both are primarily semantics of derivations and only secondarily of rules. They rest on the idea that introduction inferences are fundamental and give meaning to logical connectives, whereas all other inferences are justified by certain procedures. This idea goes back to certain programmatic remarks by Gentzen. It has been made precise in inversion principles and principles of harmony of various sorts.

4.2.1 Inversion principles and harmony

In his *Investigations into Natural Deduction*, Gentzen makes some, nowadays very frequently quoted, programmatic remarks on the semantic relationship between introduction and elimination inferences in natural deduction.

The introductions represent, as it were, the 'definitions' of the symbols concerned, and the eliminations are no more, in the final analysis, than the consequences of these definitions. This fact may be expressed as follows: In eliminating a symbol, we may use the formula with whose terminal symbol we are dealing only 'in the sense afforded it by the introduction of that symbol'. (Gentzen, 1934/35, p. 80)

This cannot mean, of course, that the elimination rules are *deducible* from the introduction rules in the literal sense of the word; in fact, they are not. It can only mean that they can be *justified* by them in some way.

By making these ideas more precise it should be possible to display the E-inferences as unique functions of their corresponding I-inferences, on the basis of certain requirements. (ibid., p. 81)

So the idea underlying Gentzen's programme is that we have "definitions" in the form of introduction rules and some sort of semantic reasoning which, by using "certain requirements", validate the elimination rules.

We cannot discuss in detail the philosophical reasons which might support Gentzen's programme. For that we would have to refer to Dummett's work and in particular to his claim that there are two different aspects of language use: one connected with 'directly' or 'canonically' asserting a sentence, and another one with drawing consequences from such an assertion (see especially Dummett, 1991). The first is the primary or 'self-justifying' way corresponding to reasoning by introduction rules, whereas the second one, which corresponds to reasoning by elimination rules, is in need of justification. This justification relies on the *harmony* which is required to hold between both aspects: The possible *consequences* to be drawn from an assertion are determined by the *premisses* from which the assertion can possibly be inferred by direct means.

By adopting Lorenzen's term and accommodating its idea to the context of natural deduction, Prawitz (1965) formulated an "inversion principle" to make Gentzen's remarks more precise:

Let α be an application of an elimination rule that has B as consequence. Then, deductions that satisfy the sufficient condition [...] for deriving the major premiss of α , when combined with deductions of the minor premisses of α (if any), already "contain" a deduction of B ; the deduction of B is thus obtainable directly from the given deductions without the addition of α . (p. 33)

Here the sufficient conditions are given by the premisses of the corresponding introduction rules. Thus the inversion principle says that a derivation of the conclusion of an elimination rule can be obtained without an application of the elimination rule if its major premiss has been derived using an introduction rule in the last step, which means that a combination

$$\frac{\frac{\mathcal{D}}{A} \text{ I-inference} \quad \{\mathcal{D}_i\}}{B} \text{ E-inference}$$

of steps, where $\{\mathcal{D}_i\}$ stands for a (possibly empty) list of deductions of minor premisses, can be avoided.

At first glance, this simply states the fact that maximum formulas, i.e. formulas being conclusions of an I-inference and at the same time major premiss of an E-inference (in the example: A), can be removed by means of certain reductions, which leads to the idea of a normal derivation. However, it also represents a *semantical* interpretation of elimination inferences by saying that nothing is gained by an application of an elimination rule if its major premiss has been derived *according to its meaning* (i.e. by

means of an introduction rule). So the reductions proposed by Prawitz for the purpose of normalization are at the same time semantic justifications of elimination rules with respect to introduction rules. His inversion principle elaborates Gentzen’s idea of “special requirements” needed for this justification, by demanding that elimination rules invert introduction rules in a precise sense.

That it corresponds indeed to what Gentzen had in mind can be seen by taking a closer look at the example Gentzen gives:

We were able to introduce the formula $A \rightarrow B$ when there existed a derivation of B from the assumption formula A . If we then wished to use that formula by eliminating the \rightarrow -symbol (we could, of course, also use it to form longer formulae, e.g., $(A \rightarrow B) \vee C$, \vee -I), we could do this precisely by inferring B directly, once A has been proved, for what $A \rightarrow B$ attests is just the existence of a derivation of B from A . (Gentzen, 1934/35, pp. 80–81)

This may be read as follows: Given the situation

$$\frac{\frac{\frac{A}{\mathcal{D}}}{B} \quad \mathcal{D}'}{A \rightarrow B} \quad A}{B}$$

where \mathcal{D} is “a derivation of B from the assumption formula A ”, and \mathcal{D}' is the derivation showing that “ A has been proved”, so that we can use $A \rightarrow B$ to obtain B “by eliminating the \rightarrow -symbol”. Then by means of

$$\frac{\mathcal{D}'}{A} \quad \mathcal{D}}{B}$$

we can infer “ B directly, once A has been proved [by means of \mathcal{D}']”, as “ $A \rightarrow B$ attests [...] the existence of a derivation [viz. \mathcal{D}] of B from A ”. According to this reading, Gentzen describes the standard reduction for implication later made explicit by Prawitz and used in his normalization proof (see appendix A).

However, although Gentzen’s remarks are correctly read as outlining a semantic programme, he himself takes a more formalistic stance, which is clear from his writings in general and from the continuation of the passage quoted above:

Note that in saying this we need not go into the “informal sense” [“inhaltlicher Sinn”]²² of the \rightarrow -symbol. (Gentzen, 1934/35, p. 81)

²²Quotes by Gentzen.

Prawitz (1965) deserves the credit to have drawn our attention to the genuine semantic content of Gentzen’s remarks, though this is not spelled out in detail in his monograph. Only later in Prawitz (1971) and in particular in Prawitz (1973, 1974) is it turned into a full-fledged semantic theory.

As remarked in section 4.1.1, Lorenzen’s (1955) inversion principle has a much more general function. It is an admissibility principle allowing one to establish the admissibility of certain rules in a calculus. Its general reading is that everything that can be obtained from each defining atom of an object can be obtained from this atom itself. In this generality it is quite near to the principle of definitional reflection (see section 4.3.2). It can, however, be shown that Lorenzen’s approach can be put in close relationship with Prawitz-style proof-theoretic semantics (see Schroeder-Heister, 2008a).

What is common both to Lorenzen’s original inversion principle and Prawitz’s adaptation to natural deduction is that it is a principle for the justification of rules on the basis of the consideration of derivations. Because of the latter it is a global principle according to our classification (section 3). Dummett-Prawitz-style proof-theoretic semantics as described in the next section continues this global proof-based perspective, whereas proof-theoretic semantics based on definitional reflection (see section 4.3.2), though also related to Lorenzen’s conception, emphasizes the rule-based perspective.

The relationship between introduction and elimination rules is often described as “harmony”, or as governed by a “principle of harmony” (see, e.g. Tennant, 1978, p. 74²³). This terminology is not uniform and sometimes not even fully clear. It essentially expresses what is also meant by “inversion”. Even if “harmony” is a term which suggests a symmetric relationship, it is frequently understood as expressing the conception based on introduction rules (although occasionally one includes an elimination based conception as well). In most cases authors understand by harmony the fact that a pair of introductions and eliminations is conservative (together with explicitness and separation). Only very rarely there is some requirement of uniqueness, although this is crucial as well. Sometimes harmony is supposed to mean that connectives are strongest or weakest in a certain sense given their introduction or their elimination rules. This idea underlies Tennant’s (1978) harmony principle, and also Popper’s and Koslow’s structural characterizations (see section 4.4.1).

4.2.2 Prawitz’s notion proof-theoretic validity

Proof-theoretic validity is the dominating approach to proof-theoretic semantics. As a technical concept it was developed by Prawitz (1971, 1973, 1974), by turning a proof-theoretic validity notion based on Tait’s ideas (see section 2.6), and originally used to prove strong normalization, into a semantical concept. Dummett provided much

²³According to my knowledge, this is the first use of this term in this sense in the literature.

philosophical underpinning to this notion (see Dummett, 1991). Unlike the concepts discussed in the previous sections 4.1 and 4.2.1 it is a notion that applies to whole proofs rather than rules, so the objects which can be valid are proofs as representations of arguments. In this sense it is a global rather than a local notion. It applies to arbitrary derivations over a given atomic system, which defines derivability for atoms. The definition of validity is based on Gentzen’s idea that introduction rules are ‘self-justifying’ and give the logical constants their meaning. A *canonical derivation* is a derivation, which uses an introduction rule in the last step. Noncanonical derivations are justified by reducing them to canonical ones. Thus reduction procedures as used in normalization proofs (see appendix A) play a crucial role. As they justify arguments, they are also called “justifications” by Prawitz. This definition only applies to closed derivations, turning the proof-theoretic result that every closed derivation reduces to one which uses an introduction rule in the last step, into a philosophical principle (called by Dummett (1991) the ‘fundamental assumption’ of the theory of meaning). Open derivations are justified by considering their closed instances, which are obtained by replacing open assumptions with closed proofs of them.

Prawitz definition of validity, of which there are several variants, can be reconstructed as follows. We consider only the constants of positive propositional logic (conjunction, disjunction, implication). We assume that an atomic system S is given determining the derivability of atomic formulas, which is the same as their validity. A formula over S is a formula built up by means of logical connectives starting with atoms from S . We want to define the validity of a derivation which proceeds from formulas over S as assumptions to a formula over S as conclusion. Such a derivation is not necessarily a derivation in a given formal system: We want to tell of an *arbitrary* derivation whether it is valid or not. We propose the term “derivation structure” for such an arbitrary derivation. (Prawitz uses various terminologies, such as “[argument or proof] schema” or “[argument or proof] skeleton”.) Derivation structures are candidates for valid derivations. More precisely, a derivation structure is a formula tree which resembles a natural deduction tree with the difference that it is composed of arbitrary rules. Such rules can have arbitrary and arbitrarily many premisses, and each premiss may depend on assumptions which are discharged at this step. So the general form of an inference rule is the following, where the square brackets indicate assumptions which can be discharged at the application of the rule:

$$\frac{[C_{11}, \dots, C_{1m_1}] \quad \dots \quad [C_{n1}, \dots, C_{nm_n}]}{A_1 \quad \dots \quad A_n} \quad , \quad \text{in short:} \quad \frac{\Gamma_1 \quad \dots \quad \Gamma_n}{A_1 \quad \dots \quad A_n} \quad .$$

Obviously, the standard introduction and elimination rules are particular cases of such rules. As a generalization of the standard reductions of maximal formulas it is supposed that certain reduction procedures are given. A reduction procedure transforms a given

derivation structure into another one. A set of reduction procedures is called a *derivation reduction system* and denoted by \mathcal{J} . Reductions serve as justifying procedures for non-canonical steps, i.e. for all steps, which are not self-justifying, i.e., which are not introduction steps. Therefore a reduction system \mathcal{J} is also called a *justification*. Reduction procedures must satisfy certain constraints such as closure under substitution. As the validity of a derivation not only depends on the atomic system S but also on the derivation reduction system used, we define the validity of a derivation structure with respect to the underlying atomic basis S and with respect to the justification \mathcal{J} :

- (i) *Every closed derivation in S is S -valid with respect to \mathcal{J} (for every \mathcal{J}).*
- (ii) *A closed canonical derivation structure is S -valid with respect to \mathcal{J} , if its immediate substructure \mathcal{D} is S -valid with respect to \mathcal{J} .*

$$\begin{array}{c} A \\ \mathcal{D} \\ B \end{array}$$
- (iii) *A closed non-canonical derivation structure is S -valid with respect to \mathcal{J} , if it reduces, with respect to \mathcal{J} , to a canonical derivation structure, which is S -valid with respect to \mathcal{J} .*
- (iv) *An open derivation structure $\begin{array}{c} A_1 \dots A_n \\ \mathcal{D} \\ B \end{array}$, where all open assumptions of \mathcal{D} are among A_1, \dots, A_n , is S -valid with respect to \mathcal{J} , if for every extension S' of S and every extension \mathcal{J}' of \mathcal{J} , and for every list of closed derivation structures $\begin{array}{c} \mathcal{D}_i \\ A_i \end{array}$ ($1 \leq i \leq n$), which are S' -valid with respect to \mathcal{J}' , $\begin{array}{c} \mathcal{D}_1 \quad \mathcal{D}_n \\ A_1 \dots A_n \\ \mathcal{D} \\ B \end{array}$ is S' -valid with respect to \mathcal{J}' .²⁴*

In clause (iv), the reason for considering extensions \mathcal{J}' of \mathcal{J} and of extensions S' of S , is, a monotonicity constraint. Derivation should remain valid if one's knowledge incorporated in the atomic system and in the reduction procedures is extended.

Examples demonstrating validity are given in the next subsection (4.2.3). Here we continue with the abstract presentation and discussion of the general concept. A corresponding concept of *universal validity* can be defined as follows: Let S_0 be the atomic system with only propositional variables as atoms and with no inference rules. Let $\mathcal{L}(S_0)$ be a set of derivation structures over S_0 together with a justification \mathcal{J} . Let v be an assignment of S -formulas to propositional variables. Let \mathcal{D}^v be obtained from \mathcal{D} by substituting in \mathcal{D} propositional variables p with $v(p)$. Let \mathcal{J}^v be the set of reductions which acts on derivations \mathcal{D}^v in the same way as \mathcal{J} acts on \mathcal{D} (i.e., \mathcal{J}^v is

²⁴See Prawitz (1973, p. 236; 1974, p. 73; 2006).

the homomorphic image of \mathcal{J} under v). Then a derivation structure \mathcal{D} in $\mathcal{L}(S_0)$ (i.e. a derivation structure containing only propositional variables as atoms) is defined to be *universally valid* with respect to \mathcal{J} iff for every S and every v , \mathcal{D}^v is S -valid with respect to \mathcal{J}^v . It is easy to see that \mathcal{D} is universally valid with respect to \mathcal{J} iff \mathcal{D} is S_0 -valid with respect to \mathcal{J} . This means that we can use the term “valid” (with respect to some \mathcal{J}) interchangeably for both universal and S_0 -validity.

Validity with respect to some \mathcal{J} can be viewed as a generalized notion of logical validity. In fact, if we specialize \mathcal{J} to the standard reductions of intuitionistic logic (see appendix A), then all derivations in intuitionistic logic are valid with respect to \mathcal{J} (see the next subsection 4.2.3). This is semantical *correctness*. We may ask if the converse holds, viz. whether, given that a derivation \mathcal{D} over S_0 is valid with respect to some \mathcal{J} , there is a derivation in intuitionistic logic with the same end-formula and without any open assumption beyond those already open in \mathcal{D} . That intuitionistic logic is complete with respect to this semantics is natural to assume. However, no satisfactory proof of this fact has been given so far. This problem, also known as Prawitz’s conjecture (see Prawitz, 1973, 2010), is still open.

The S -validity of a generalized inference rule

$$\frac{\Gamma_1 \quad \dots \quad \Gamma_n}{A_1 \quad \dots \quad A_n} B$$

with respect to a justification \mathcal{J} means that for all derivations $\mathcal{D}_1, \dots, \mathcal{D}_n$, which are S' -valid with respect to \mathcal{J}' for extensions S' and \mathcal{J}' of S and \mathcal{J} , respectively, the derivation

$$\frac{\Gamma_1 \quad \dots \quad \Gamma_n}{\mathcal{D}_1 \quad \dots \quad \mathcal{D}_n} \cdot \frac{A_1 \quad \dots \quad A_n}{B}$$

is S' -valid with respect to \mathcal{J}' . For a simple inference rule

$$\frac{A_1 \quad \dots \quad A_n}{A}$$

this means that it is valid with respect to \mathcal{J} , if the one-step derivation structure of the same form is S -valid with respect to \mathcal{J} .

This gives rise to a corresponding notion of consequence.²⁵ Instead of saying that the rule

$$\frac{A_1 \quad \dots \quad A_n}{A}$$

²⁵See also Prawitz (1985).

is S -valid with respect to \mathcal{J} , we may say that A is a *consequence* of A_1, \dots, A_n with respect to S and \mathcal{J} ($A_1, \dots, A_n \models_{S, \mathcal{J}} A$); if we consider universal validity with respect to \mathcal{J} , we may speak of *consequence with respect to \mathcal{J}* ($A_1, \dots, A_n \models_{\mathcal{J}} A$); and finally, if there is some \mathcal{J} such that universal validity holds for \mathcal{J} , then we may speak of *logical consequence* ($A_1, \dots, A_n \models A$).

This makes *proof-theoretic* consequence differ from *constructive* consequence according to which

$$\frac{A_1 \quad \dots \quad A_n}{A}$$

would be defined as valid with respect to a *constructive function* f , if f transforms valid arguments of the premisses A_1, \dots, A_n into a valid argument of the conclusion A . Actually, it is not always possible to extract such a constructive function from our derivation reduction system, as a reduction system \mathcal{J} serving as a justification need not be deterministic, which means that it merely generates a constructive relation on arguments. However, constructive consequence may be viewed as a limiting case of proof-theoretic consequence (see below point (3)).

I list some characteristic features and problems of the notion of proof-theoretic validity.

(1) Closed derivations are primary as compared to open ones. The definition of validity starts in clause (ii) with the validity of closed derivation structures. The validity of open derivation structures is defined in clause (iv) by considering their closed instances which are obtained by substituting open assumptions with closed derivation structures. In this respect proof-theoretic validity crucially differs from computability which is defined for open derivations (see section 2.6). The distinction between canonical and non-canonical derivations is crucial for validity, but is considered a subdivision of closed derivation as expressed by clauses (ii) and (iii) of the definition. This is another difference to computability. To represent this comparison graphically, the definition of proof-theoretic validity proceeds according to the concept tree

$$\left| \begin{array}{l} \text{closed} \left| \begin{array}{l} \text{canonical} \\ \text{non - canonical} \end{array} \right. \\ \text{open} \end{array} \right.$$

in contradistinction to the notion of computability which is based on the concept tree

$$\left| \begin{array}{l} \text{canonical} \\ \text{non - canonical} \left| \begin{array}{l} \text{reducible} \\ \text{irreducible} \end{array} \right. \end{array} \right.$$

This also shows that the reducibility-irreducibility distinction, which is crucial for computability (as it is a tool for proving normalization), does not play any role in proof-

theoretic validity (but, of course, the notion of reduction in clause (iii) of the definition). The fact that the validity of open derivations is reduced to that of closed ones, may be considered a critical point. It contains a well-foundedness assumption, which is trivially satisfied in the semantics of logical constants, but may be questioned in a proof-theoretic semantics for a wider class of expressions, as, e.g., studied in definitional reflection (see section 4.3.2). Independently of such considerations one may also ask whether the evaluation of open expressions, might open up novel perspectives, as recently considered by Martin-Löf (2009).

(2) The validity of a derivation structure is defined not only with respect to atomic systems, which represent, so to speak, the ‘material base’, but also with respect to a justification \mathcal{J} . It is this justification which, via clause (iii) of the definition, is crucial for validity. Without a justification \mathcal{J} , only derivations based on introduction rules could be rendered as valid. Now justifications operate solely on derivation structures. It is the derivation tree which is being reduced or transformed. However, if a justification provides information crucial for validity, one might consider justifications which not just rewrite a given derivation structure but also transform the justification associated with it. One would then consider derivation structures *together with justifications* as primary objects of study and allow for justifications to operate on these pairs and not exclusively on derivation structures alone. This idea has been put forward by Prawitz (2010) but not worked out in detail yet.

(3) Proof-theoretic validity lives on the idea that, by means of justifications \mathcal{J} , derivations are transformed or reduced in the sense that other derivations are constructed from them. Models for this are the standard reductions of intuitionistic logic used in proofs of normalization (see section A), where one takes out parts of given proofs and recombines them in a new way. It is not normally meant that a new derivation is generated ‘out of the blue’, even if this generation can be described by a constructive function. This makes proof-theoretic semantics differ from constructive semantics in the sense of the BHK interpretation. A derivation structure \mathcal{D} of B from A together with a justification \mathcal{J} should give more information than just the fact that a derivation of B can be constructed from one of A . It should rather tell us something about the way to obtain B from A . However, it cannot be excluded in principle that \mathcal{J} is nothing but a (constructive) function generating a derivation of B from one of A , quite independent of \mathcal{D} . In this case constructive validity is a limiting case of proof-theoretic validity. Proof-theoretic semantics covers a wide spectrum of describing the way from A to B . Its intensional notion of consequence is more fine-grained than that of constructive semantics, where one only has constructive transformations. Further developments may even attempt to distinguish different grades of information for different \mathcal{J} contained in a pair $\langle \mathcal{D}, \mathcal{J} \rangle$. The examples given in the next subsection (4.2.3) provide some intuitive arguments in favour of such distinctions.

(4) It is sometimes said that in intuitionistic logic, and especially in the BHK interpretation, the definition of implication is impredicative in the sense that one quantifies over all proofs of A when one defines a proof of $A \rightarrow B$, where a proof of A may itself contain implications of any complexity. If correct, this argument would also apply to the definition of validity due to the quantification in clause (iv). However, a careful inspection shows that the definition of validity is an inductive definition in a precise sense, which, apart from considering reductions in clause (iii), proceeds by induction over the complexity of end formulas of closed derivation structures, and makes no validity assumption about any internal steps used. In fact, validity is a global definition that applies to derivations as a whole and not to any single inference step. The validity of single inference steps is defined by considering them to be one-step derivations. What might only be objected to the definition of validity is that the generalized inductive definition, which contains a quantifier in clause (iv) ranging over objects in antecedent (i.e., negative) position, is too complex to be acceptable as a semantic definition. The borderline between predicative and non-predicative generalized inductive definitions is not sharp, and some might already regard the definition used here as impredicative. But this is then not a sense of impredicativity that is unacceptable for ‘obvious’ reasons, and definitely not one that is not normally accepted in constructive mathematics. Usberti (2006) used the notion of “epistemic transparency” as a requirement for an epistemologically adequate definition of validity. One might, for example, call into question that the standard definition of validity satisfies this requirement, as in the case of implication it might be impossible to tell what it means to be in possession of a valid derivation, even if an idealized agent is considered.

(5) In contradistinction to the notions discussed in section 4.1, validity is a global notion. Candidates of validity are whole derivations structures, and reductions apply to whole derivation structures. A rule is valid if it leads from valid derivations to valid derivations, so derivations are conceptually prior over rules. This assumption is called into question in definitional reflection, where, again, rules are primary, and where it is possible to consider derivations which are only locally valid.

For further details concerning Prawitz-style proof-theoretic validity see Schroeder-Heister (2006).

4.2.3 Examples of proof-theoretic validity

For simplicity, we disregard atomic systems S and speak of \mathcal{J} -validity for validity with respect to \mathcal{J} . First we observe that any derivation that results from the composition of \mathcal{J} -valid rules and/or \mathcal{J} -valid derivations is itself \mathcal{J} -valid. For example, the derivation

$$\frac{\frac{A \quad B}{C}}{\frac{D_1 \quad D_2}{\frac{D \quad E}{F}}}$$

is \mathcal{J} -valid, if the rules $\frac{A \quad B}{C}$ and $\frac{D \quad E}{F}$ as well as the derivations \mathcal{D}_1 and \mathcal{D}_2 are \mathcal{J} -valid.

As our first example, we show that the rule of \rightarrow elimination (modus ponens) is valid with respect to $\{sr(\rightarrow)\}$, i.e., with respect to the justification consisting just of the standard reduction for implication (see appendix A). For that we have to show that for any $\mathcal{J} \supseteq \{sr(\rightarrow)\}$, and for all closed \mathcal{J} -valid derivations $\frac{\mathcal{D}_1}{A \rightarrow B}$ and $\frac{\mathcal{D}_2}{A}$, the derivation

$$\frac{\frac{\mathcal{D}_1}{A \rightarrow B} \quad \frac{\mathcal{D}_2}{A}}{B}$$

is \mathcal{J} -valid. Since \mathcal{D}_1 is closed \mathcal{J} -valid, it is of the form, or reduces with respect to \mathcal{J} to the form

$$\frac{\frac{A}{\mathcal{D}'_1}}{B},$$

where \mathcal{D}'_1 is \mathcal{J} -valid. Applying $sr(\rightarrow)$, which is part of \mathcal{J} , to

$$\frac{\frac{\frac{A}{\mathcal{D}'_1}}{B} \quad \mathcal{D}_2}{A \rightarrow B} \quad \frac{\mathcal{D}_2}{A}}{B}$$

yields the derivation

$$\frac{\mathcal{D}_2}{\frac{A}{\mathcal{D}'_1}} \quad \frac{\mathcal{D}_2}{B}.$$

This derivation is \mathcal{J} -valid, as it is the result of a composition of the \mathcal{J} -valid derivations \mathcal{D}'_1 and \mathcal{D}_2 . In a similar way we can demonstrate the validity of \wedge and \vee elimination with respect to the standard reductions $sr(\wedge)$ and $sr(\vee)$ as justifications.

As our second example, we show that the rule of importation

$$(R_{imp}) \quad \frac{A \rightarrow (B \rightarrow C)}{A \wedge B \rightarrow C}$$

is valid with respect to the justification $\mathcal{J}_{imp} = \{sr(\rightarrow), sr(\wedge), r_1, r_2\}$, where $sr(\rightarrow)$ and $sr(\wedge)$ are, as before, the standard reductions for implication and conjunction, and r_1 and r_2 are the following reductions:

$$r_1 : \quad \frac{\frac{(1) \quad [A]}{\mathcal{D}}}{B \rightarrow C} (1)}{A \rightarrow (B \rightarrow C)} \quad \text{reduces to} \quad \frac{\frac{(2) \quad [A]}{\mathcal{D}} (1)}{B \rightarrow C} [B]}{\frac{C}{B \rightarrow C} (1)} (2)}{A \rightarrow (B \rightarrow C)}$$

$$r_2 : \frac{\frac{\frac{\mathcal{D}}{C} \text{ (1)}}{B \rightarrow C} \text{ (1)}}{A \rightarrow (B \rightarrow C)} \text{ (2)}}{A \wedge B \rightarrow C} \text{ (2)} \quad \text{reduces to} \quad \frac{\frac{\frac{\mathcal{D}}{C} \text{ (1)}}{A \wedge B \rightarrow C} \text{ (1)}}{A \wedge B} \text{ (1)}}{A \wedge B} \text{ (1)}$$

We have to show that for every $\mathcal{J} \supseteq \mathcal{J}_{imp}$ and for every closed \mathcal{J} -valid derivation

$\frac{\mathcal{D}}{A \rightarrow (B \rightarrow C)}$ the derivation

$$(\mathcal{D}_1) : \frac{\mathcal{D}}{A \rightarrow (B \rightarrow C)} \frac{\mathcal{D}}{A \wedge B \rightarrow C}$$

is \mathcal{J} -valid. Since \mathcal{D} is closed \mathcal{J} -valid, is is of the form, or reduces with respect to \mathcal{J} to the form

$$\frac{\frac{\mathcal{D}'}{B \rightarrow C} \text{ (1)}}{A \rightarrow (B \rightarrow C)} \text{ (1)},$$

where \mathcal{D}' is \mathcal{J} -valid. Applying r_1 to this derivation yields

$$(\mathcal{D}_2) : \frac{\frac{\frac{\mathcal{D}'}{B \rightarrow C} \text{ (1)}}{B \rightarrow C} \text{ (1)}}{A \rightarrow (B \rightarrow C)} \text{ (2)}$$

which is \mathcal{J} -valid, as it is composed of the \mathcal{J} -valid derivation \mathcal{D}' and \mathcal{J} -valid rules (note that \rightarrow elimination is \mathcal{J} -valid since $sr(\rightarrow)$ belongs to \mathcal{J} , and introduction rules are

trivially valid). This means that \mathcal{D}_1 reduces with respect to \mathcal{J} to $\frac{\mathcal{D}_2}{A \wedge B \rightarrow C}$, which, by means of r_2 , reduces to

$$\frac{\frac{\frac{\frac{\mathcal{D}'}{B \rightarrow C} \text{ (1)}}{A \wedge B} \text{ (1)}}{A \wedge B} \text{ (1)}}{A \wedge B \rightarrow C} \text{ (1)}$$

The latter derivation structure is \mathcal{J} -valid as being composed of the \mathcal{J} -valid derivation structure \mathcal{D}' and \mathcal{J} -valid rules (\wedge elimination and \rightarrow elimination are \mathcal{J} -valid, because $sr(\rightarrow)$ and $sr(\wedge)$ are in \mathcal{J}).

Alternatively, R_{imp} can be shown to be valid with respect to $\mathcal{J}'_{imp} = \{sr(\rightarrow), sr(\wedge), r_3\}$, where r_3 is defined as:

$$r_3 : \frac{\mathcal{D} \quad A \rightarrow (B \rightarrow C)}{A \wedge B \rightarrow C} \text{ reduces to } \frac{\mathcal{D} \quad \frac{[A \wedge B] \quad A}{B \rightarrow C} \quad (1) \quad [A \wedge B] \quad B}{\frac{C}{A \wedge B \rightarrow C}} \quad (1)$$

The comparison of the standard reductions ($sr(\rightarrow)$, $sr(\wedge)$, $sr(\vee)$) with the reductions r_1, r_2 and r_3 shows that the former are more elementary than the latter in that they just compose given subderivations, whereas r_1, r_2 and r_3 use additional steps to generate their output. r_1 uses $\rightarrow E$ and introduction rules, r_2 uses $\wedge E$ and introduction rules, and r_3 uses both $\rightarrow E$ and $\wedge E$, and introduction rules. In using standard elimination inferences, both \mathcal{J}_{imp} and \mathcal{J}'_{imp} have to rely on the standard reductions for the connectives involved. In fact, r_3 can be viewed as a derivation of R_{imp} within natural deduction. \mathcal{J}_{imp} can be viewed somewhat more elementary than \mathcal{J}'_{imp} in that it not just straightforwardly gives a natural deduction derivation, but requires first a reduction of the premiss derivation of R_{imp} in order to be able to apply r_1 . In a sense, \mathcal{J}_{imp} just generates a derivation of the conclusion of R_{imp} from its premiss, so it comes nearest to constructive semantics, where just a transformation from derivations into derivations is required. However, the specific form of the proof of the conclusion of R_{imp} and the fact that \mathcal{D} is used at all in the derivation of the conclusion of R_{imp} gives at least a certain amount of information. It should be emphasized that by far not every valid rule has a justification consisting of elementary reductions only. Importation is a counterexample. As soon as one has a right-iterated implication in the premiss of a rule, we have to rely on non-elementary reduction to establish its validity.

4.2.4 Martin-Löf type theory

The method of proof terms is a technical device according to which the fact that a formula A has a certain proof can be codified as the fact that a certain term t is of type A , whereby the formula A is identified with the type A . By means of this method, which was introduced by Curry and Howard (see de Groote, 1995; Sørensen & Urzyczyn, 2006), formulas can be considered as the types of their proofs. This can again be put into a calculus for type assignment, whose statements are of the form $t : A$. A proof of $t : A$ in this system can be read as showing that t codifies a natural deduction proof of A . If t contains variables, $t : A$ may depend on declarations of the form $x_1 : A_1, \dots, x_n : A_n$, where the A_1, \dots, A_n correspond to the open assumptions on which the natural deduction derivation of A depends. This idea is exploited in type-theoretical systems such as Martin-Löf's (Martin-Löf, 1984; Nordström, Petersson, & Smith, 1990; Sommaruga, 2000) which especially use the idea of *dependent* types, i.e.,

types containing variables for terms, etc.

Martin-Löf (1995, 1998) has put this into a philosophical perspective by distinguishing a two-fold sense of proof. First we have proofs of statements of the form

$$t : A .$$

These statements are called *judgements*, their proofs are called *demonstrations*. Within such judgements the term t represents a *proof* of the *proposition* A . A proof in this sense is also called a *proof object*. So when demonstrating a judgement $t : A$, we demonstrate that a proposition has a certain proof²⁶. Within this two-layer system the *demonstration* layer is the layer of argumentation. Unlike proof objects, demonstrations have epistemic significance; their judgements carry assertoric force. The proof layer is the layer at which meanings are explained: The meaning of a proposition A is explained by telling what counts as a proof (object) for A . The distinction made between canonical and non-canonical proofs etc. is a distinction at the propositional and not at the judgemental layer.

On the background of Prawitz's definition of validity, one could expect that Martin-Löf gives a formal definition of validity for proof (objects). However, this is not the case, at least not in the sense of a metalinguistic inductive definition of what is a valid proof (object). Rather, he gives a justification of demonstration steps which refers to the meanings of propositions and to the forms of proof (objects) referred to in its judgements. For example, for the case of implication, the rules for judgements of the form $t : A$ are the following (the standard typing rules of the typed *lambda*-calculus):

$$\frac{[x : A] \quad t(x) : B}{\lambda x.t(x) : A \rightarrow B} (\rightarrow I) \quad \frac{t : A \rightarrow B \quad t' : A}{tt' : B} (\rightarrow E)$$

where $t(x)$ denotes a term in which x may occur free, and tt' denotes the term application of t to t' .

According to Martin-Löf the justification runs roughly as follows:

($\rightarrow I$) The proof (object) $\lambda x.t(x)$ is in canonical form, so it is a proof of $A \rightarrow B$, provided for every proof (object) t' of A , $t(t')$ is a proof of B . The latter holds, as the demonstration of $t(x) : B$ from $x : A$ convinces us exactly of this fact, namely that $t(t')$ proves B if t' proves A .

($\rightarrow E$) Suppose demonstrations of $t : A \rightarrow B$ and $t' : A$ are given. Then they convince us that t is a proof (object) of $A \rightarrow B$ and t' one of A . As a proof of $A \rightarrow B$, t is already

²⁶I do not discuss here other forms of judgements which occur in type theory.

in canonical form or reduces to a proof in canonical form. In each of the two cases, tt' reduces to a proof in canonical form, i.e., tt' is a proof of B .²⁷

By justifying (making evident) demonstration steps, Martin-Löf establishes the means for validating proofs. If I have demonstrated $a : A$, I have shown that a is in fact a proof of A . So in a sense Martin-Löf also defines what it means for a proof to be valid (i.e., to be a “real” proof). But this definition is given in the form of the explanation and justification of a system for the demonstration of validity. The crucial difference to Prawitz’s procedure is that it is not *metalinguistic* in character, where metalinguistic means that candidates of proofs are specified first and then, by means of a definition in the metalanguage, it is fixed which of them are valid and which are not. Rather, proofs come into play only in the context of demonstrations. I give a proof of A by presenting an object a , of which I demonstrate that it is a proof of A .

Presenting and validating a proof takes place at the same level. Not the proof itself has epistemic force, but its validation in form of a demonstration endows it with epistemic force. Conversely this means that making an assertion, i.e. using epistemic force, includes presentation of the proof as a proof object. So when asserting something, I am not just relying on a proof in the sense that I *can* justify it, if I am asked to do so, but I am presenting it as something which by my very reasoning turns out to be a proof (and is as such justified). Proving something and demonstrating the validity of this proof cannot be separated.

This implies a certain explicitness requirement. When I prove something, I not only have to have a justification for my proof *at my disposal* as in Prawitz, but at the same time have to be *certain* that this justification fulfils its purpose, which is much more. This certainty is guaranteed by a demonstration.

In Martin-Löf’s theory, proof-theoretic semantics receives a strongly ontological component. A recent debate deals with the question of whether proof objects have a purely ontological status (which Martin-Löf claims now, see also Sundholm, 2000) or whether they codify knowledge (even if they are not epistemic acts, see Prawitz, 2009). According to our global-local classification (section 3), Martin-Löf’s proposal is primarily global, as the meaning explanations operate at the level of terms, which stand for proofs. However, as these proofs are understood as proof objects, the explanations are not global in considering demonstration structures. The meaning explanations for proof terms rather proceed by laying down demonstration rules which are supposed to be evident. Therefore, concerning the level of demonstrations, Martin-Löf’s theory can be considered local, as he proposes rules which contain meaning explanations.

²⁷To make this fully precise, we would have to take equality judgements into consideration, which govern the reductions of proof objects. In a sense, the application operation (indicated here by concatenation) is a kind of justifying operation which, when applied to a canonical proof, yields a proof, as the equality rules for the evaluation of applications (which correspond to β -reduction) show.

4.3 Clausal definitions and definitional reasoning

Proof-theoretic semantics normally focuses on logical constants. This focus is practically never questioned, apparently because it is considered so obvious. This may have to do with the fact that usually, proof theory is the proof theory of logical systems and systems extending logic (such as arithmetic). In proof theory, little attention has been paid to atomic systems, although there has been Lorenzen's early work (see section 4.1.1), where the justification of logical rules is embedded in a theory of arbitrary rules, and Martin-Löf's (1971) theory of iterated inductive definitions where introduction and elimination rules for atomic formulas are proposed. The rise of logic programming has widened this perspective. From the proof-theoretic point of view, logic programming is a theory of atomic reasoning with respect to clausal definitions of atoms. Definitional reflection is an approach to proof-theoretic semantics that takes up this challenge and attempts to build a theory whose range of application goes beyond logical constants. In particular it can deal with phenomena whose associated definitions are not well-founded. It is a local approach throughout based on the primacy of rules. Even though these rules obey certain adequacy conditions, they do not expect that whole derivation behave in a certain 'nice' way. Otherwise we would not have the option to give up well-foundedness (and with it separation).

4.3.1 The challenge from logic programming

In logic programming we are dealing with program clauses of the form

$$A \Leftarrow B_1, \dots, B_n$$

which *define* atomic formulas. Such clauses can naturally be interpreted as describing introduction rules for atoms. This is quite natural for a PROLOG programmer who reads clauses as rules, although this reading is blurred by the understanding of clauses as disjunctions of the form $\neg A \vee B_1 \vee \dots \vee B_n$, which is common in treatments of logic programming within the framework of classical logic.

If one takes the "rule"-interpretation of clauses seriously, one is inevitably led to a proof-theoretic treatment, which reads programs as collections of introduction rules. Such an approach has been carried out in detail in Hallnäs and Schroeder-Heister (1990/91)²⁸. It has especially led to extensions of definite Horn clause programming by considering iterations of the rule arrow in bodies of clauses, and, correspondingly, to a natural treatment of negation.

From the point of view of proof-theoretic semantics the following two points are essential:

(1) Introduction rules (clauses) for logically compound formulas are not distinguished in principle from introduction rules (clauses) for atoms. The introduction rules for

²⁸See also Schroeder-Heister (1991a).

conjunction and disjunction would, for example be handled by means of clauses for a truth predicate with conjunction and disjunction as term-forming operators:

$$\mathbb{D}_{log} \left\{ \begin{array}{l} T(p \wedge q) \Leftarrow T(p), T(q) \\ T(p \vee q) \Leftarrow T(p) \\ T(p \vee q) \Leftarrow T(q) \end{array} \right.$$

In order to define implication, we need a rule arrow in the body, which, for the whole close, corresponds to using a higher-level rule:

$$\left\{ T(p \rightarrow q) \Leftarrow T(p) \Rightarrow T(q) \right.$$

We need, of course, some sort of “background logic”. This is the structural logic governing the comma and the rule arrow, which determine the way the bodies of clauses are handled. In standard logic we have just to the comma, which is handled implicitly. In extended versions of logic programming we would have the (iterated) rule arrow, i.e., structural implication and associated principles governing it, and perhaps even structural disjunction (this is present in disjunctive logic programming, but not needed for the applications considered here). What is important here is that we are throughout dealing with clauses for atoms, perhaps with some structuring of their defining conditions. An example for a clause which defines an atom by using structural implication, which does not represent a logical constant, would be a clause for a disposition predicate:

$$\left\{ \text{water-soluble}(x) \Leftarrow (\text{put_into_water}(x) \Rightarrow \text{dissolves}(x)) \right.$$

(2) The rules one is dealing with are not necessarily well-founded. It is not even required that all possible instances of an atom are defined. For example, the clauses

$$\left\{ \begin{array}{l} p(a, x) \Leftarrow q(a), r(x), p(a, f(x)) \\ p(x, b) \Leftarrow s(b), r(x) \end{array} \right.$$

are appropriate as program clauses, although they give only a partial definition of p (namely for instances of the forms $p(a, \cdot)$ and $p(\cdot, b)$), and although they include a non-terminating recursion. According to the approach followed in logic programming there is no point in asking whether syntactically correct program clauses are well-formed with respect to semantic considerations. So logic programming proclaims a great deal of *definitional freedom* in generating programs.

From these observations we learn the following:

(Ad 1) Interpreting logic programming proof-theoretically motivates an extension of proof-theoretic semantics to arbitrary atoms, which yields a semantics with a much wider realm of applications. This is the topic of the next subsection. Conversely, such a proof-theoretic semantics leads itself to interesting extensions of logic programming,

as such a semantics generates elimination rules corresponding to clausal introduction rules, which can be successfully exploited in logic programming. This idea was put into practice in the extended logic programming languages GCLA and Gisela (Aronsson, Erkksson, Gäredal, Hallnäs, & Olin, 1990; Torgersson, 2000).

(Ad 2) The use of arbitrary clauses without further requirements in logic programming is a motivation to pursue the same idea in proof-theoretic semantics, admitting just any sort of introduction rules and not just those of a special form, and in particular not necessarily ones which are well-founded. This idea, which takes definitional freedom over to semantics, is a key concept of definitional reflection as discussed in the next subsection.

Some final remarks concerning Lorenzen's (1955) and Martin-Löf's (1971) works are appropriate, as they both deal with introduction rules for atoms. Lorenzen's approach is very near to what is done in logic programming as far as the declarative aspects are concerned, in particular his start from production rules for atoms. His inversion principle is closely related to the more general principle of definitional reflection dealt with in the next section. Although Martin-Löf shares with ideas in logic programming and definitional reflection the fundamental idea that atomic formulas can be treated similarly to logical constants, his elimination rules for atoms fundamentally differ from those considered in the following. This is particularly clear from his treatment of mathematical induction which for him is an elimination rule corresponding to introduction rules for a natural number predicate.

The idea of considering introduction rules as meaning-giving rules for atoms is closely related to the theory of inductive definitions in its general form, where inductive definitions are nothing but systems of production rules (see Aczel, 1977). Since a definition is the classic way of endowing something with meaning, it is very natural to include such rules in proof-theoretic semantics. The relationship between inductive definitions and ideas in logic programming has been pointed out from a proof-theoretic perspective by Denecker, Bruynooghe, and Marek (2001).

4.3.2 Definitional Reflection

The logic of definitional reflection as developed by Hallnäs and Schroeder-Heister is a decidedly local rather than global approach (in the sense of section 3). It takes up the challenge from logic programming and gives a proof-theoretic semantics not just for logical constants but for arbitrary expressions, for which a clausal definition can be given. The proof-theoretic semantics of logical constants is then nothing but a special case. It is the non-logical cases which are particularly interesting, as the logical ones have many features which make definitional reflection undistinguishable in many respects from more standard approaches. So it is crucial to stress the *general* character of this sort of proof-theoretic semantics, covering others as special cases.

With this general character: a definition consists of arbitrary clauses, which may

even contain structural implications embedded in their bodies, we give up principles that are characteristic for the global, proof-based approaches. The first such principle is *well-foundedness*. We do not assume that a set of definitional clauses is stratified in the sense that, when going back the definitional chain from definiendum to definiens we reach an empty defining condition after finitely many steps. So definitional chains may be circular or infinitely descending. With that the even stronger principle characteristic of logical constants falls according to which an expression is defined in structural terms alone, referring only to its subterms. This principle is *separation*. We do not assume that we can separate the defining clauses for two expressions a and b , i.e., it may well be that a is defined in terms of b and b in terms of a . So a definition with clauses for a and b cannot necessarily be split into two definitions, one with clauses for a and another one with clauses for b . Further possible principles are discarded as well. A definition (not necessarily of a single expression) is a list of clauses, that is all. So we allow for full definitional freedom. Actually, the failure of separation is already something that one finds in a simultaneous inductive definition of two predicates, a trivial example being the definition

$$\left\{ \begin{array}{l} \text{even}(0) \Leftarrow \\ \text{even}(S(x)) \Leftarrow \text{odd}(x) \\ \text{odd}(S(x)) \Leftarrow \text{even}(x) . \end{array} \right.$$

Although in this this simple case, the definition can, be reformulated to obtain separation, one might wonder if separation is a desirable feature at all, or whether it is just something that accidentally holds in the case of logical constants. The example given at the end of section 2.7 show that it breaks down already in the definition of logical constants indirectly defined in terms of other operators. In contradistinction to separation, the feature of well-foundedness hits basic opinions about definitions. However, in the case of the revision theory of truth, the investigations that have started with the works of Kripke, Gupta, Herzberger and Belnap have shown that the exclusion of non-wellfoundedness obstructs the view on interesting phenomena (for an overview see Kremer, 2009).

As a consequence of our general approach which takes arbitrary clausal definitions as its base, without any further constraint beyond the clausal syntactic form, we cannot expect that the derivations we obtain from such definitions have certain ‘desirable’ properties. In particular, we cannot expect that they satisfy certain normal form conditions. This does not mean that the global structure of derivations obtained is irrelevant. It is simply not a part of the definition. Features of the global structures of the derivations obtained are very important, but as *consequences* of the clauses which we consider as meaning giving, not as features restricting possible definitions. For example, whether our formal system of proofs enjoys cut elimination or normalization is not something built into the syntax of a definition (as is essentially the case in Prawitz-

style validity), but a matter of the particular choice of definitional clauses. In the ‘well-behaved’ case of logical constants we do have normalization and cut elimination, but in other cases, which are not so well-behaved, but which do occur, we do not have this property. That is why we, following Hallnäs, also speak of “partial inductive definitions”, which should express that definitional clauses determine meaning, i.e., have a semantical function, but in some cases only a ‘partial’ one, where the ‘total’ ones are the cases where we obtain ‘nice’ global properties such as cut elimination. The term “partial” is here chosen as a reminiscence of partial recursive functions in recursive function theory. There the fact that a function is total is a global feature which cannot be decided from the recursive definition as a local entity. The fact that we presuppose neither well-foundedness nor separation, means that our clausal definitions are holistic in a sense: The definitions of various expressions may be mixed together and be mutually interdependent. Unlike Dummett, we do not think this is a negative feature of a semantics which must be avoided at any price. To be sure, it can and is avoided in the case of logical constants. But it cannot be avoided in other cases, and these cases do not deprive the defining clauses of any sense. Even in a properly partial definition we have the particular syntax of clauses: The head serves as the definiendum, and the body as the definiens (or as one of the definientia, if there are multiple clauses with the same head) of an expression. So it is not just any set of sentences which determines meaning, but clauses of a particular form.

It turns out that the definitional approach can be better expressed using a sequent-style representation of derivations rather than natural deduction format (see appendix A). It can even be claimed that this is not a matter of convenience but that, in a sense, sequent-style systems are more “natural” than natural deduction. The decisive feature of the sequent calculus, by which we here mean the ‘genuine’ sequent calculus, not sequent-style natural deduction, are the introduction rules for formulas in the antecedent. These are rules which introduce a formula according to its meaning. Unlike natural deduction, where there is only one rule for introducing an assumption, namely just posing it as the starting point of a derivation, the left-introductions in the sequent calculus are different for every connective. This technical feature has not been given sufficient philosophical attendance. Translated into natural deduction it means that we can not only assert but also assume propositions according to their meaning (i.e., dependent on their form). Technically, this would correspond to a bidirectional system of natural deduction whose derivations can be extended both to the top and to the bottom²⁹. Assumptions would be introduced in a specific way, just like assertions.

Formally, our system is such that we have a list of clauses which is the *definition*

²⁹And where one would have to use generalized elimination rules with major premisses in top position. See Schroeder-Heister (2009).

we are considering. Each clause has the form

$$a \Leftarrow B$$

where the head a is an atomic formula (atom). In the simplest case, the body B is a list of atoms b_1, \dots, b_m , in which case a definition looks like a definite logic program. We often consider an extended case where B may also contain some structural implication ‘ \Rightarrow ’, and sometimes even some structural universal implication, which essentially is handled by restricting substitution. Given a definition \mathbb{D} , the list of clauses whose head starts with the predicate P is called the *definition of P* . In the propositional case where atoms are just propositional letters, we speak of the *definition of a* having the form

$$\mathbb{D}_a \left\{ \begin{array}{l} a \Leftarrow B_1 \\ \vdots \\ a \Leftarrow B_n . \end{array} \right.$$

However, it should be clear that normally the definition of P or of a is just a particular part of a definition \mathbb{D} , which contains clauses for other expressions as well, and that this definition \mathbb{D} cannot not always be split up into separate definitions of its predicates or propositional letters. So ‘definition of a ’ or ‘of P ’ is a *façon de parler*. What is always meant is the list of clauses for a predicate or propositional letter within a definition \mathbb{D} .

Syntactically, a clause looks like an introduction rule, especially if one looks at the definition \mathbb{D}_{log} of the propositional constants in section 4.3.1. However, in the theory of definitional reflection we separate the definition, which is incorporated in the set of clauses, from the inference rules, which put it into practice. So instead of different introduction rules which define different expressions we have a general schema which applies a given definition. Separating the specific definition from the inference schema using arbitrary definitions gives us wider flexibility. We need not consider introduction rules to be basic and other rules to be derived from them. Instead we can speak of certain inference principles which determine the inferential meaning of a clausal definition, and which are of equal stance. This is in fact what we are claiming here. There is a pair of inference principles putting a definition into action, which are in harmony with each other, without one of them being preferential. As we are working in a sequent-style framework, we have inferential principles for introducing the defined constant on the right and on the left of the turnstile, i.e. in assertion and in assumption position. For simplicity we consider the case of a propositional definition \mathbb{D} , which has no predicates, functions, individual variables or constants, and in which the bodies of clauses are just lists of propositional letters. Suppose \mathbb{D}_a (as above) is the definition of a (within \mathbb{D}), and the B_i have the form ‘ b_{i1}, \dots, b_{ik_i} ’ as in propositional logic programming. Then the right-introduction rules for a are

$$(\vdash a) \frac{\Gamma \vdash b_{i1} \quad \dots \quad \Gamma \vdash b_{ik_i}}{\Gamma \vdash a}, \quad \text{in short} \quad \frac{\Gamma \vdash B_i}{\Gamma \vdash a} \quad (1 \leq i \leq n) .$$

and the left-introduction rule for a is

$$(a \vdash) \frac{\Gamma, B_1 \vdash C \quad \dots \quad \Gamma, B_n \vdash C}{\Gamma, a \vdash C}$$

If we talk generically about these rules, i.e., without mentioning a specific a , but just the definition \mathbb{D} , we also write $(\vdash \mathbb{D})$ and $(\mathbb{D} \vdash)$. The right introduction rule expresses reasoning ‘along’ the clauses. It is also called *definitional closure*, by which is meant ‘closure under the definition’. The intuitive meaning of the left introduction rule is the following: Everything that follows from every possible definiens of a , follows from a itself. It is called the *principle of definitional reflection*, as it reflects upon the definition as a whole. If B_1, \dots, B_n exhaust *all possible conditions* to generate a according to the given definition, and if each of these conditions entails the very same conclusion, then a itself entails this conclusion.

The crucial principle, which gives the whole theory its name, is definitional reflection. It extracts deductive *consequences* of a from a definition, in which only defining *conditions* of a are given. If the clausal definition \mathbb{D} is viewed as an inductive definition, this principle can be viewed as expressing the extremal clause in inductive definitions: Nothing else beyond the clauses given defines a . To give a very simple example, consider the following definition:

$$\left\{ \begin{array}{l} \text{child_of_tom} \Leftarrow \text{anna} \\ \text{child_of_tom} \Leftarrow \text{robert} \end{array} \right.$$

Then one instance of the principle of definitional reflection with respect to this definition is

$$\frac{\text{anna} \vdash \text{tall} \quad \text{robert} \vdash \text{tall}}{\text{child_of_tom} \vdash \text{tall}}$$

Therefore, if on the basis of other information we know $\text{anna} \vdash \text{tall}$ and $\text{robert} \vdash \text{tall}$, we can infer $\text{child_of_tom} \vdash \text{tall}$.

Since definitional reflection depends on the definition as a whole, taking *all* definiens of a into account, it is non-monotonic with respect to \mathbb{D} . If \mathbb{D} is extended with an additional clause

$$a \Leftarrow B_{n+1}$$

for a , then previous applications of the $(\mathbb{D} \vdash)$ rule may fail to remain valid. In the present example, if we add the clause

$$\text{child_of_tom} \Leftarrow \text{john}$$

we can no longer infer $\text{child_of_tom} \vdash \text{tall}$, except we also know $\text{john} \vdash \text{tall}$. Note that due to the definitional reading of clauses, which gives rise to inversion, the sign “ \Leftarrow ”

expresses more than just implication, in contradistinction to structural implication “ \Rightarrow ” that may occur in the body of a clause. To do justice to this fact, one might instead use “ \vdash ” as in PROLOG, or “ $\vdash=$ ” to express that we are dealing with some sort of definitional equality.

In standard logic programming one has, on the declarative side, only what corresponds to definitional closure. Definitional reflection leads to powerful extensions of logic programming, which hinge, of course on the computation procedures to be associated with definitional reflection. These computational procedures, though very interesting in themselves, are not relevant in the present context, where we deal only with the declarative aspects.

The principles of definitional closure and definitional reflection are considered as local principles which do not presuppose anything about the structure of the derivation they are embedded into. Although they much resemble certain admissibility principles considered by Lorenzen (see section 4.1.1) and can, in a different context, even interpreted as such (see Schroeder-Heister, 2007), they are not understood in this sense here. The harmony between definitional closure and definitional reflection is here just considered locally at the level of rules. The idea that by means of definitional reflection we get from a defined object a exactly what we have put into it by means of definitional closure is understood locally as something that pertains to a single step and not necessarily to the derivation as a whole.

For example, if we combine definitional closure and definitional reflection, we can reduce cut with the definiendum to cut with the definiens by reducing

$$\frac{\frac{\Gamma \vdash B_i \quad \Gamma, B_1 \vdash C \quad \dots \quad \Gamma, B_n \vdash C}{\Gamma \vdash a} \quad \Gamma, a \vdash C}{\Gamma \vdash C}$$

to

$$\frac{\Gamma \vdash B_i \quad \Gamma, B_i \vdash C}{\Gamma \vdash C}$$

However, this reduction only says that, if cut is valid for the definiens, it is valid for the definiendum. It does not say anything about the global eliminability of cut from an arbitrary derivation. (We do not assume cut as a primitive rule of inference!) So if cut elimination expresses conservativeness, then we have just *relative* or *one-step conservativeness*: If we have it for the definiendum, we have it for the definiens. In the case of non-wellfounded definitions this does not imply a global property. The same holds for the second Belnap condition (see section 2.7). If we consider a duplicate of our definition with all propositional letters starred, then in the joint system we have the derivation

$$\frac{\frac{B_1 \vdash B_1^*}{B_1 \vdash a^*} \quad \dots \quad \frac{B_n \vdash B_n^*}{B_n \vdash a^*}}{a \vdash a^*}$$

This shows that if we have uniqueness for each defining condition B_i of a , we do have it for a . This is *relative* uniqueness and does not necessarily mean that we have absolute uniqueness meaning that $a \vdash a^*$ is derivable outright.³⁰ So we do have harmony, but only in a relative and local sense. Whether certain global principles hold, is a different question and not a matter of whether a definition can be admitted.

In our reasoning system, we would, in addition to definitional closure and definitional reflection also have general principles governing consequence ‘ \vdash ’, in particular the initial sequent

$$(Ini) \quad A \vdash A$$

and structural rules like thinning and contraction, depending on which substructural framework we are using. There will also be rules governing structural implication ‘ \Rightarrow ’ and structural universal quantification, if one needs them. However, cut is under no circumstances a primitive rule (and in some cases not even an admissible rule).

The case with variables cannot be described in any detail. We just mention a fundamental point by means of an example. Suppose we have the following definition, in which the atoms have a predicate-argument-structure:

$$\left\{ \begin{array}{l} \text{child_of_tom(anna)} \Leftarrow \text{daughter_of_tom(anna)} \\ \text{child_of_tom(robert)} \Leftarrow \text{son_of_tom(robert)} \\ \qquad \text{tall(anna)} \Leftarrow \text{daughter_of_tom(anna)} \\ \qquad \text{tall(robert)} \Leftarrow \text{daughter_of_tom(robert)} \end{array} \right.$$

Given our propositional rule of definitional reflection, we could just infer propositional results such as $\text{child_of_tom(anna)} \vdash \text{tall(anna)}$ or $\text{child_of_tom(robert)} \vdash \text{tall(robert)}$. However, what we would like to infer is

$$\text{child_of_tom}(x) \vdash \text{tall}(x)$$

with free variable x , since *anna* and *robert* are the only objects, for which the predicate *child_of_tom* is defined, and since for these instances the desired principle holds. This leads to a principle of definitional reflection according to which for the introduction of an atom a on the left side of the turnstile the most general unifier of a with the heads of all definitional clauses is considered. For further details see Schroeder-Heister (1993).

As an example of a non-wellfounded definition consider

$$\mathbb{D}_r \quad \{r \Leftarrow (r \Rightarrow \perp)\}$$

³⁰I have used a somewhat sloppy notation. B_i^* means the conditions obtained by replacing a with a^* , and the B_i occurring on the right side of the turnstile means, of course, multiple sequents, and in the case of cut, multiple cuts, if B_i consists of more than one formula. (B_i is here never understood disjunctively, as in the classical sequent calculus.)

where \perp is a constant not defined in the definition considered. If we (quite naturally) assume that the structural background logic gives us the sequent $r, r \Rightarrow \perp \vdash \perp$, then by means of definitional reflection we can, on the left side, replace $r \Rightarrow \perp$ with r , yielding $r \vdash \perp$ (here we have also used contraction). If we (again quite naturally) assume that the structural background logic gives us $\vdash r \Rightarrow \perp$ from $r \vdash \perp$, we obtain $\vdash r$ by definitional closure. This means that we have derived both $r \vdash \perp$ and $\vdash r$, which would give us $\vdash \perp$, if cut were eliminable. However, since \perp is undefined, $\vdash \perp$ cannot not be derivable, which means that cut is not admissible.³¹

This example shows that local cut reduction, which we have in our system, does not entail global admissibility of cut. This is not considered a defect, but an advantage. Otherwise we would have to rule out definitions like that of r from the very beginning. In Prawitz-style validity this is implicitly done in the definition of validity, as it proceeds on the complexity of the end-formula of a closed derivation structure, making it impossible to consider such a case. This deprives us from dealing with certain phenomena, which do occur, even if they are unwanted. We can only classify local definitions according to their global consequences, but not rule out such definitions, as we do not normally know these consequences. This makes our local approach strongly differ from proof-theoretic semantics in the Dummett-Prawitz-sense. We retain basic definitional features at the local level without sacrificing the freedom of formulating definitions.

It is an easy task to give a natural deduction version of definitional reflection, and even a corresponding type system. The left-introduction rule ($a \vdash$) for an atom a would then be turned into an a -elimination rule of the form

$$(aE) \frac{a \quad \frac{[B_1] \quad C \quad \dots \quad C}{C}}{C} \cdot$$

The failure of cut elimination in the non-wellfounded case appears here in the form of derivations which cannot be normalized.³² If we do not consider non-well-founded cases, but only the ‘nice’ ones, definitional reflection can indeed be viewed as carrying over

³¹Therefore, as a derivation of absurdity, assuming that cut is available, this derivation reads as follows:

$$\frac{\frac{\frac{r \vdash r}{r, r \Rightarrow \perp \vdash \perp}}{r \vdash \perp}}{\vdash r \Rightarrow \perp} \quad \frac{\frac{r \vdash r}{r, r \Rightarrow \perp \vdash \perp}}{r \vdash \perp}}{\vdash r \quad r \vdash \perp} \vdash \perp$$

³²The analysis of Russell’s paradox in naive set theory in terms of non-normalizing derivations has been carried out first by Prawitz (1965, Appendix B, p. 95). Prawitz explicitly remarks that this is a

proof-theoretic semantics to the case of clauses, i.e., as including non-atomic definitions. However, it would be missing the point if this were considered to be its central feature. By insisting on local rules rather than global proof reductions it develops its power only in the non-standard cases.

4.3.3 Examples, applications and problems of definitional reflection

Logical Constants. That the standard laws for the positive logical constants of intuitionistic propositional logic can be derived using the definition \mathbb{D}_{log} of section 4.3.1 is obvious. As we have a local approach based on rules, nothing comparable to the reduction procedures in Prawitz-style validity semantics is needed. Formally, we have to apply the reflection rule for definitions with variables, as the propositional variables in \mathbb{D}_{log} are individual variables in the sense of the definition. Using the natural deduction formulation (*aE*) of definitional reflection, we obtain the generalized elimination rules which are also considered in the rule-based semantics of implication (section 4.1.3).³³

Absurdity. If \perp is an atom for which there is no defining clause in a definition \mathbb{D} , then we obtain as a limiting case of definitional reflection, with the empty list of premisses, the axiom $\Gamma, \perp \vdash C$. This gives us intuitionistic logic in a natural way. However, the problems with the handling of limiting cases is essentially the same as in validity-based semantics. We may instead discard the metalinguistic consideration of limiting cases and state explicit principles for \perp . This would then lead away from the intuitionistic framework, but is definitely a viable option (see section 5.5). Intuitionistic logic lives on dealing with absurdity as a limiting case.

Generalized definitional reflection: Negation and interaction. The idea of definitional reflection (and therefore of inversion and the inversion principle) can be generalized. In the form presented here we considered hypothetical derivations expressed as derivations of hypothetical judgements in the sequent calculus, where definitional reflection said that everything that can be derived from each defining condition of A can be derived from A itself. Carrying over this to the case of negation and denial, it might be formulated as: Everything that contradicts each defining condition of A , contradicts A itself, yielding introductions of the denial of A (see section 5.3). Or even more generally, if we have defined some sort of interaction between conditions: Everything

system where any given maximum formula can be removed, but not all of them. In our terminology, we have local, but not global cut reduction.

³³For the sequent version, we obtain left introduction rules corresponding to the generalized elimination rules in natural deduction, which, except in the case of implication, are identical to the standard left introduction rules. Implication \rightarrow is reduced to structural implication \Rightarrow , for which in the structural background logic we would have appropriate principles equivalent to the standard implication laws.

that interacts with the defining conditions of A in a certain way, interacts with A itself in that way. What “interaction” here means, is determined by the specific content. For an application with respect to the treatment of symmetry in the sequent calculus (including that for linear logic) see section 5.4.

Inversion and expert systems. The sort of inversion inherent in definitional reflection is frequently needed in knowledge-based systems, where one wants to extract information from sentences which are defined in terms of clauses. Medical expert systems are a particular application (see Falkman, 2003). Though not relevant to proof-theoretic semantics, which is primarily a philosophical and foundational enterprise, it is illuminating to see that inversion as a theoretical concept is used in practical applications. This also gives further weight to a form of proof-theoretic semantics which goes beyond logical constants.

Structured expressions as atoms. Following the usage in logic programming, we have interpreted the definition arrow ‘ \Leftarrow ’ in definitions as relating (atomic) formulas. However, this is not mandatory. We are defining expressions which are structured in a certain way, and it does not matter in principle whether this is a term structure or a formula structure. Only minor adjustments when formulating the reflection rule are necessary. The definition \mathbb{D}_{log} of propositional logical constants then becomes more natural, as we do not need to embed it into a truth predicate and can just write:

$$\mathbb{D}'_{log} \left\{ \begin{array}{l} p \wedge q \Leftarrow p, q \\ p \vee q \Leftarrow p \\ p \vee q \Leftarrow q \\ p \rightarrow q \Leftarrow p \Rightarrow q \end{array} \right.$$

Function definition. Definitional reflection is a broad framework, which covers also notions in which the turnstile ‘ \vdash ’ is not interpreted as expressing consequence in the narrower sense. One such interpretation is its interpretation as ‘computes to’, if we extend the notion of a definition in such a way as to define functions and not only predicates. For example, the following definition of the function *plus*:

$$\left\{ \begin{array}{l} plus(0, y) \Leftarrow y \\ plus(s(x), y) \Leftarrow s(plus(x, y)) \\ s(x) \Leftarrow (x \Rightarrow y) \Rightarrow s(y) \end{array} \right.$$

Here the third clause serves to make sure that, if x computes to y , then $s(x)$ computes to $s(y)$. This idea has been put forward by Hallnäs (1991) and Fredholm (1995). Computation then proceeds by definitional reflection. For example, we can prove $plus(s(0), s(0)) \vdash s(s(0))$, and the right hand side can be viewed as a computed answer

substitution. For other mathematical applications such as the definition of functionals see Hallnäs (2006). This again demonstrates that inversion and reflection is by far not confined to logic.

Free equality and completion. An instructive example for the power of definitional reflection is free equality. We obtain all axioms of free equality from the definition

$$\left\{ x = x \leftarrow$$

For example, we can derive symmetry $x = y \vdash y = x$ by observing that the substitution of y with x is the only way to obtain $x = x$ from $x = y$, and that for this substitution we have $\vdash y = x$. Thus by means of definitional reflection we obtain the theory of free equality which is needed in the theory of completion in logic programming, which itself can be viewed as expressing a sort of inversion in logic programming. For the relationship between definitional reflection and completion see Schroeder-Heister (1994).

The interaction of cut, contraction and initial sequents. We have seen that for the circular definition

$$\mathbb{D}_r \{ r \leftarrow (r \Rightarrow \perp) \}$$

cut is not admissible. Now cut elimination is a global feature which depends not only on definitional closure and reflection. So we may distinguish different conditions for the admissibility of cut: (1) Those depending on the particular form of the definition \mathbb{D} and (2) those which depend on features of the deduction system which are independent of the form of \mathbb{D} .

Ad (1) Cut is admissible if the definition \mathbb{D} is well-founded, i.e., if for any ground instance of a defined expression a , every chain of definitional successors starting with a terminates. Here e_2 is a definitional successor of e_1 , if e_1 is defined in terms of e_2 . More precisely, a ground expression e_2 is a definitional successor of a ground expression e_1 , if e_2 occurs in the body B of a clause $e_1 \leftarrow B$, which is a ground instance of a clause in \mathbb{D} . This result is what one naturally expects, and what holds in the case of the standard logical connectives. Cut is also admissible, if no structural implication ‘ \Rightarrow ’ occurs in the body of a definitional clause. So if the definition \mathbb{D} is a definite program, we do have cut elimination. This reflects the view that in the definition \mathbb{D}_r the complexity goes up when passing from definiendum to definiens. However, it should be emphasized that only the complexity in terms of structural implication counts. Structural conjunction in terms of the comma has no effect. This means that in the presence of logical constants, conjunction and disjunction have no effect either, since conjunction-disjunction formulas in bodies can be resolved by (recursively)

replacing conjunction with the comma and disjunction with two alternative clauses. Implication and negation can have an impact on the admissibility of cut, as implication is definitionally reduced to structural implication, and negation to implication and absurdity.

Ad (2). Even with circular or otherwise non-wellfounded definitions, cut is admissible, if the structural rule of contraction is dropped. This is not very surprising, as since the work of Curry and Fitch it is well-known that choosing a logic without contraction prevents many paradoxes. In the derivation of absurdity with respect to \mathbb{D}_r , using cut sketched in section 4.3.2, we identify two occurrences of r in the antecedent, thus using contraction. Therefore, if we considered a contraction-free framework to be convincing, we could continue to work with cut. It would actually be sufficient to disallow those versions of contraction, where the two expressions being contracted have been introduced in semantically different ways. By this we mean the following: In the definitional framework an expression a can be introduced either by means of an initial sequent $a \vdash a$ or by one of the two definitional principles $(\vdash a)$ or $(a \vdash)$. In the first case, a is introduced in an *unspecific* way, i.e., independent of its definition, whereas in the second case, a is introduced in a *specific* way, namely according to its meaning as given by the definition of a . It can be shown that contraction is only critical (with respect to cut elimination) if a specific occurrence of a is identified with an unspecific one. This situation obtains in the considered derivation with respect to \mathbb{D}_r . In order to retain cut, we would have to block contraction only in these critical cases. Another way of saving cut elimination, which corresponds to the result just mentioned, is to restrict initial sequents $a \vdash a$ to those cases, where there is no definitional clause for a available, i.e. where a cannot be introduced by definitional closure and reflection. The rationale behind this proposal is that, if a has a definitional meaning, then a should be introduced according to its meaning and not in an unspecific way. This corresponds to the idea of the logical sequent calculus, in which one often restricts initial sequents to the atomic case, i.e., to the case, where no meaning-determining right and left introduction rules are available. This way of proceeding was proposed by Kreuger (Kreuger, 1994) and is related to using certain four-valued semantics in logic programming (see Jäger & Stärk, 1998). See Schroeder-Heister (1992, 2004) for overviews of cut elimination in relation to definitional reflection.

The form of definitions. Why do we choose the particular form of clausal definitions as the basis of our semantic theory? There are at least three reasons for it. (1) This form of definitions has proved powerful, as the development of logic programming shows. Together with definitional reflection it becomes even more powerful. Due to the logic programming applications it is directly connected to computational matters and opens up a wide range of applications. (2) Clausal definitions can be read as

inductive definitions, which means that the most important definitional tool in mathematics can be framed in it, at least in principle. Explicit definitions, which are also natural candidates, can be viewed as a special case of inductive definitions. Inductive definitions come in various strengths, and strong ones go beyond what is included in elementary definitional reflection as presented here. However, what can be expressed by clauses in our sense is already considerable, and more than what is normally taken into account in proof-theoretic semantics.³⁴ (3) It is a generalization of the definition in terms of introduction and/or elimination rules that one is used to in the semantics of logical constants. So clausal definition are a natural extension of the standard way of proceeding in proof-theoretic semantics. Leaving an older theory, which has proved successful in its range of application, as a specific case, is always a good strategy when developing a new theory.

4.4 Characterization versus semantics

4.4.1 Structural characterization of logical constants

There is a large field of ideas and results concerning what might be called the “structural characterization” of logical constants, where “structural” is here meant both in the proof-theoretic sense of “structural rules” and in the sense of a framework that bears a certain structure, where this framework is again proof-theoretically described. Some of its authors use a semantical vocabulary and at least implicitly suggest that their topic belongs to proof-theoretic semantics. Others explicitly deny these connotations, emphasizing that they are interested in a characterization which establishes the logicity of a constant. The question “What is a logical constant?” can be answered in proof-theoretic terms, even if the semantics of the constants themselves is truth-conditional: Namely by requiring that the (perhaps truth-conditionally defined) constants show a certain inferential behaviour that can be described in proof-theoretic terms. However, as some of the authors consider their characterization at the same time as a semantics, it is appropriate that we mention some of these approaches here.

The most outspoken structuralist with respect to logical constants, who explicitly understands himself as such, is Koslow. In his *Structuralist Theory of Logic* (1992) he develops a theory of logical constants, in which he characterizes them by certain “implication relations”, where an implication relation roughly corresponds to a finite consequence relation in Tarski’s sense (which again can be described by certain structural rules of a sequent-style system). However, Koslow does not give sequent-style rules for these constants, at least not in the first place, but uses a metalinguistic characterization telling the reader what a conjunction, a disjunction, an implication etc. is, even when it is not designated or conceived as such in a language. For example, a

³⁴A treatment of inductive definitions within sequent calculi systems, which is closely related to definitional reflection, has been carried out by Brotherston and Simpson (2007).

conjunction C of A and B must satisfy the conditions that (i) C implies both A and B , (ii) C is the weakest object (with respect to the given implication relation) such that (i) is fulfilled (i.e., for any C' , if C' implies both A and B , then C' implies C). Koslow develops a structural theory in the precise metamathematical sense, which does not specify the domain of objects in any sense beyond the axioms given. Even if the domain is supposed to be a language, the structural axioms do not tell what a conjunction of A and B looks like (if there is one at all). Rather, if a language or any other domain of objects equipped with an implication relation is given, the structural approach may be used to single out logical compounds by checking their implicational properties. It does not postulate axioms and inference rules for a formal object language. Whether and how implication structures are realized as object languages, is entirely left open. In particular, nothing is being said about the inferential format used in such a realization (e.g., whether it takes the form of a Hilbert-style or a Gentzen-style system).

In his early papers on the foundations of logic, Popper gave inferential characterizations of logical constants in proof-theoretic terms. He used a calculus of sequents and characterized logical constants by certain derivability conditions of such sequents. His terminology clearly suggests that he intends a proof-theoretic semantics of logical constants, as he speaks of “inferential definitions” and the “trivialization of mathematical logic” achieved by defining constants in the way described. Although his presentation is not free from conceptual imprecision and errors, he was the first to consider the sequent-style inferential behaviour of logical constants to characterize them. This is all the more remarkable as he was probably not at all, and definitely not fully aware of Gentzen’s sequent calculus and Gentzen’s further achievements (he was in correspondence with Bernays, though). However, against his own opinion, his work can better be understood as an attempt to define the logicity of constants and to structurally characterize them, than as a proof-theoretic semantics in the genuine sense. He nevertheless anticipated many ideas now common in proof-theoretic semantics, such as the characterization of logical constants by means of certain minimality of maximality conditions with respect to introduction or elimination rules. For detailed expositions and reconstructions of Popper’s approach see Schroeder-Heister (1984c, 2005).

Important contributions to the logicity debate that characterize logical constants inferentially in terms of sequent calculus rules that they obey are those by Kneale (1956) and Hacking (1979). Hacking claims, on the one hand, that the rules of Gentzen’s sequent calculus should not be regarded as definitions, but as characterizations (Hacking, 1977, p. 377). On the other hand he considers right- and left-introduction rules of a particular form as definitions within a so-called “do-it-yourself semantics” of logical constants (Hacking, 1977, p. 385; 1979, pp. 312–314). His approach can perhaps best be understood in terms of approaches making the symmetry in the sequent calculus explicit (see 5.4). A thorough theory of logicity was proposed by Došen (1980, 1989) in

his theory of logical constants as “punctuation marks”, expressing structural features at the logical level. He understands logical constants as being characterized by certain double-line rules for sequents which can be read in both directions. For example, conjunction and disjunction are (in classical logic, with multiple-formulae succedents) characterized by the double-line rules

$$\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B} \qquad \frac{\Gamma, A \vdash \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A \vee B \vdash \Delta}$$

Došen is able to give characterizations which even include systems of modal logic. He explicitly considers his work as a contribution to the logicity debate and not to any conception of proof-theoretic semantics. However, Sambin et al., in their *Basic Logic* (Sambin, Battilotti, & Faggian, 2000), explicitly understand what Došen calls double-line rules as fundamental meaning giving rules. The double-line rules for conjunction and disjunction are read as implicit definitions of these constants, which by some procedure can be turned into the explicit sequent-style rules we are used to. So Sambin et al. use the same starting point as Došen, but interpret it not as a structural description of the behaviour of constants, but semantically as their implicit definition. Their approach will be dealt with in section 5.4, in connection with generalized definitional reflection and other approaches to deal semantically with the symmetry of the sequent calculus (including the classical one). For the problem of demarcation of logical constants in general, not just from the inferential point of view, see MacFarlane (2009).

There are several other approaches to a uniform proof-theoretic characterization of logical constants, all of whom at least touch upon issues of proof-theoretic semantics. Such theories are Belnap’s *Display Logic* (Belnap, 1982), Wansing’s *Logic of Information Structures* (Wansing, 1993), generic proof editing systems and their implementations such as the *Edinburgh logical framework* (Harper, Honsell, & Plotkin, 1987) and many successors which allow the specification of a variety of logical systems. Since the rise of linear and, more generally, substructural logics (Di Cosmo & Miller, 2010; Restall, 2009) there are various approaches dealing with logics that differ with respect to restrictions on their structural rules. The recent movement away from singling out a particular logic as the true one towards a more pluralist stance (see, e.g., Beall & Restall, 2006) which is interested in what different logics have in common without any preference for a particular logic can be seen as a shift away from semantical justification towards structural characterization.

4.4.2 **Categorical approaches to proof-theoretic semantics**

There is an abundant literature on category theory in relation to proof theory, and, following seminal work by Lawvere, Lambek and others (see Lambek & Scott, 1986, and the references therein), category itself can be viewed as a kind of abstract proof

theory. If one looks at an arrow $A \rightarrow B$ in a category as a kind of abstract proof of B from A , we have a representation which goes beyond pure derivability of B from A (as the arrow has its individuality), but does not deal with the particular syntactic structure of this proof. As many systems, in particular intuitionistic ones have a categorial semantics, this may be considered as an abstract proof-theoretic semantics of such systems³⁵. For intuitionistic systems, proof-theoretic semantics in categorial form comes probably closest to what denotational semantics is in the classical case. A more detailed consideration of this approach would need separate consideration. A comprehensive comparison of proof theory and category theory from a philosophical perspective is a much needed desideratum.

5 Extensions and alternatives to standard proof-theoretic semantics

5.1 Elimination rules as basic, dual approaches

Following Gentzen's dictum, many approaches to proof-theoretic semantics consider introduction rules as basic, meaning giving, or self-justifying, whereas the elimination inferences are justified as valid with respect to the given introduction rules. The roots of this conception are threefold: First there is a verificationist theory of meaning according to which assertibility conditions of a sentence constitute its meaning. This seems to underly not only Dummett's philosophy, which is the most developed one in this respect, but the whole movement of intuitionism. Even if it is not directly connected to verificationism in early Wittgenstein and the Vienna circle, there are strong reminiscences of their position in verificationist proof-theoretic semantics. So there is a justificationist and verificationist bias. The second is the idea that we must distinguish between what gives the meaning and what are the consequences of this meaning, in order to cope with the 'paradox of consequence' (see Dummett, 1978). For inference to be informative, not every inference can be definitional. The informative inferences are established by reflection on the meaning of the expressions involved, without being meaning-constituting themselves. Whereas introduction steps are meaning giving, the remaining valid inferences give novel insight beyond what is 'definitionally' already contained in the premisses. The third one is the primacy of assertion over other speech acts, such as assuming or denying, which is implicit in all approaches considered so far. In Prawitz's definition of validity, and in intuitionistic semantics in general, assumptions are placeholders for proofs or constructions, and negation is reduced to implying absurdity, so there is a general bias towards positive forward reasoning, which is reflected in the primacy of forward-directed introductions. To some extent this view is

³⁵Also conversely, categories can be approached by proof-theoretic methods such as cut elimination (see Došen, 2000).

also implicit in the clause based theory of definitional reflection, as clauses are directed from bodies to heads, i.e. from defining conditions towards defined atoms. The non-determinism in clauses, i.e. the fact that several clauses may define the same atom (which in logic we have, e.g., with the introduction rules for disjunction) emphasizes this directedness. Whereas definitional closure just applies single clauses, definitional reflection extracts a certain meaning from an expression with respect to a whole definition, which can be viewed as generating the properly informative inferences.

One might consider how far one gets by considering elimination rules rather than introduction rules as a basis of proof-theoretic semantics. Such an approach would be nearer to a falsificationist methodology in Popper's sense. The philosophical problems and shortcomings of verificationism, which cannot be discussed here, would be strong arguments in favour of this alternative. The second point mentioned in the previous paragraph is indifferent with respect to the primacy of introduction or elimination rules, as it only says that there must be *one part* of the rules which is meaning giving and another one informative, so one may as well choose the elimination rules as meaning giving. The third point, the primacy of assertion, would be replaced with the Popperian claim that conjectures and therefore assumptions are primary to assertions.

Some ideas towards a proof-theoretic semantics based on elimination rather than introduction rules have been given by Dummett (1991, Ch. 13), albeit in a very rudimentary form. A more precise definition of validity based on elimination inferences is due to Prawitz (1971). In improved form, it can be presented as follows. We consider derivation structures, justifications and atomic systems as in section 4.2.2. The difference is now that the elimination inferences are considered 'self-justifying', and that the introduction rules are justified. The reductions need not to be changed for that purpose. For example, the standard reductions for the logical constants can serve for the justification of the introductions from the eliminations as well. The idea behind the definition is that, if all applications of elimination rules to the complex end-formula A of a derivation structure \mathcal{D} yield S -valid derivation structures or reduce to such (with respect to a justification \mathcal{J}), then \mathcal{D} is itself S -valid (with respect to \mathcal{J}). This suggests the following definition for positive propositional logic:

- (i) *Every closed derivation in S is S -valid_E with respect to \mathcal{J} (for every \mathcal{J}).*
- (ii \wedge) *A closed derivation structure $\frac{\mathcal{D}}{A \wedge B}$ is S -valid_E with respect to \mathcal{J} , if the closed derivation structures $\frac{\mathcal{D}}{A}$ and $\frac{\mathcal{D}}{A \wedge B}$ are S -valid_E with respect to \mathcal{J} , or reduce to derivation structures, which are S -valid_E with respect to \mathcal{J} .*
- (ii \rightarrow) *A closed derivation structure $\frac{\mathcal{D}}{A \rightarrow B}$ is S -valid_E with respect to \mathcal{J} , if for every extension S' of S and for every extension \mathcal{J}' of \mathcal{J} , and for every closed deriva-*

- tion structure $\frac{\mathcal{D}'}{A}$, which is S' -valid_E with respect to \mathcal{J}' , the (closed) derivation structure $\frac{\frac{\mathcal{D}}{A \rightarrow B} \quad \mathcal{D}'}{A}$ is S' -valid_E with respect to \mathcal{J}' , or reduces to a derivation structure, which is S' -valid_E with respect to \mathcal{J}' .
- (iiV) A closed derivation structure $\frac{\mathcal{D}}{A \vee B}$ is S -valid_E with respect to \mathcal{J} , if for every extension S' of S and every extension \mathcal{J}' of \mathcal{J} , and for all derivation structures $\frac{\mathcal{D}_1}{A}$ and $\frac{\mathcal{D}_2}{B}$ with atomic C , which are S' -valid_E with respect to \mathcal{J}' and which depend on no assumptions beyond A and B , respectively, the (closed) derivation structure $\frac{\frac{\mathcal{D}}{A \vee B} \quad \frac{\mathcal{D}_1}{C} \quad \frac{\mathcal{D}_2}{C}}{C}$ (1) is S' -valid_E with respect to \mathcal{J}' , or reduces to a derivation structure, which is S' -valid_E with respect to \mathcal{J}' .
- (iii) A closed derivation structure $\frac{\mathcal{D}}{A}$ of an atomic formula A , which is not a derivation in S , is S -valid_E with respect to \mathcal{J} , if it reduces with respect to \mathcal{J} to a derivation in S .
- (iv) An open derivation structure $\frac{A_1 \dots A_n}{\mathcal{D}} \quad B$, where all open assumptions of \mathcal{D} are among A_1, \dots, A_n , is S -valid_E with respect to \mathcal{J} , if for every extension S' of S and every extension \mathcal{J}' of \mathcal{J} , and for every list of closed derivation structures $\frac{\mathcal{D}_1}{A_i}$ ($1 \leq i \leq n$), which are S' -valid_E with respect to \mathcal{J}' , $\frac{\frac{\mathcal{D}_1 \dots \mathcal{D}_n}{A_1 \dots A_n} \quad \mathcal{D}}{B}$ is S' -valid_E with respect to \mathcal{J}' .

Clause (iv) is identical with clause (iv) in the definitions of S -validity in section 4.2.2, i.e., open assumptions in derivations are interpreted in the same way as before, namely as placeholders for closed valid derivations. Note that clause (iii) is needed, as we do not have here a notion of a canonical derivation. In the definition of validity based on introduction rules, the case considered in clause (iii) was a special case of non-canonical derivations. Clauses (i) and (iii) can be conjoined to form the single clause

- (i/iii) A closed derivation structure $\frac{\mathcal{D}}{A}$ of an atomic formula A is S -valid_E with respect to \mathcal{J} , if it reduces with respect to \mathcal{J} to a derivation in S .

We leave out the index “E” if it is clear whether validity based on introduction or on elimination rules is meant.

It is crucial that the minor premisses C in the application of $\forall E$ (and similarly for $\exists E$, if we deal with quantifiers) are atomic, otherwise the induction over the end-formulas of derivations, on which this definition is based, would break down. Prawitz's (1971) definition was without clauses for disjunction (and existential quantification), as he had not been aware at the time that for the purpose of defining validity the restriction to atomic C is sufficient³⁶. The revised proposal with atomic C was published in Prawitz (2007). There he refers to the fact that also Dummett (1991, Ch. 13) in his sketchy remarks on a "pragmatist" theory of meaning with an inverse justification based on elimination rules uses an atomic C . The fact that one can do without complex C is closely related to the fact that the definability of first-order logical constants in second-order propositional $\forall \rightarrow$ -logic, which was first observed by Prawitz (1965), can already be obtained in *predicative* second-order $\forall \rightarrow$ -logic in the sense that the latter proves the introduction and elimination rules for the defined connectives as shown by Ferreira (2006).³⁷ The addition "or reduces to a derivation structure ..." at the end of clauses (ii \rightarrow), (ii \wedge), (ii \vee) is due to me. It is called the 'reduction condition'. In the original notion of validity_E envisaged by Dummett and defined by Prawitz (and also in corresponding notions of computability) the notion of reduction does not come in until the atomic stage is reached. However, we do not see any problem with the reduction condition. It corresponds to the basic intuition of validity semantics that a derivation is valid, if it is of a certain form or reduces to such a form. Without the reduction condition, not even the justification of the standard introduction rules would be straightforward. One would have to establish first as a theorem that the reduction condition holds, i.e., that a closed derivation structure of a non-atomic formula is valid, if it reduces to a closed valid derivation structure.

Using the standard reductions, it can be shown that all introduction and elimination rules are valid. Due to the restriction of C to be atomic, we now have to justify the standard $\forall E$ rule for nonatomic C . For that we have to present appropriate reductions. For example, in order to show that

$$\begin{array}{ccc}
 (1) & & (1) \\
 [A] & & [B] \\
 \mathcal{D} & \mathcal{D}_1 & \mathcal{D}_2 \\
 \hline
 A \vee B & C_1 \wedge C_2 & C_1 \wedge C_2 \quad (1) \\
 \hline
 & C_1 \wedge C_2 &
 \end{array}$$

is valid, given that \mathcal{D} , \mathcal{D}_1 and \mathcal{D}_2 are valid, we have to use reductions according to

³⁶Repeated in Schroeder-Heister, 2006.

³⁷It was independently discovered by Sandqvist at around the same time.

which

$$\begin{array}{c}
 \begin{array}{ccc}
 (1) & (1) & \\
 [A] & [B] & \\
 \mathcal{D} & \mathcal{D}_1 & \mathcal{D}_2 \\
 \hline
 A \vee B & C_1 \wedge C_2 & C_1 \wedge C_2
 \end{array} \\
 \hline
 \frac{C_1 \wedge C_2}{C_i} \quad (1)
 \end{array}
 \text{ reduces to }
 \begin{array}{c}
 \begin{array}{ccc}
 (1) & (1) & \\
 [A] & [B] & \\
 \mathcal{D}_1 & \mathcal{D}_2 & \\
 \hline
 \mathcal{D} & C_1 \wedge C_2 & C_1 \wedge C_2 \\
 A \vee B & \hline C_i & \hline C_i
 \end{array} \\
 \hline
 \frac{C_i}{C_i} \quad (1)
 \end{array}$$

To establish validity of $\vee E$, the reduction condition in (ii \vee) is essential. For the example

$$(R_{imp}) \quad \frac{A \rightarrow (B \rightarrow C)}{A \wedge B \rightarrow C}$$

discussed in section 4.2.3 for validity based on introduction rules, we would now need the following reduction as a justification.

$$\begin{array}{c}
 \begin{array}{ccc}
 \mathcal{D} & & \\
 \hline
 A \rightarrow (B \rightarrow C) & & \mathcal{D}' \\
 \hline
 A \wedge B \rightarrow C & & A \wedge B \\
 C & & \hline
 C
 \end{array} \\
 \text{ reduces to } \\
 \begin{array}{ccc}
 \mathcal{D}' & & \\
 \hline
 A \wedge B & & \mathcal{D}' \\
 \hline
 A & & A \wedge B \\
 \hline
 B \rightarrow C & & \hline
 C & & B \\
 \hline
 C
 \end{array}
 \end{array}$$

Note that now we do not need any of the standard reductions, i.e., importation is valid with respect to the justification consisting of this reduction alone. The reduction condition in clause (ii \wedge) is again essential.

There is also a corresponding notion of computability based on elimination rules for the purpose of strong normalization proofs. Actually, this notion is more common in today's presentations than computability based on introduction rules. It is used, for example, in Troelstra and Schwichtenberg (1996).

The intuition behind this approach based on elimination rules is that a derivation is valid, if the result of an application of an elimination rule to its end-formula is valid. This means that it is not valid due to its very form (as in the introduction rule approach), but due to its application. Its validity depends on that of the immediate consequences we can reach starting with this derivation. So one might call it a consequentialist view of validity. This is an original approach, which brings a fresh idea into proof-theoretic validity. It must be noted, however, that basic tenets of introduction-based validity concepts are kept. Among those is the primacy of closed derivations and the interpretation of open derivations. In both validity conceptions the definition of validity starts with closed derivations. And in both conceptions the validity of an open derivation is defined via the substitution of closed derivations for the open assumptions in open derivations, as expressed by the fact that clause (iv) of the definition of validity, which deals with open derivations, is identical in both conceptions. This means that both approaches are still biased towards assertions (by means of closed derivations), whereas assumptions are just placeholders for what can be asserted by means of closed

derivations. It is assertions which, in the elimination rule approach, are justified by their consequences. It is definitely not the case that assumptions receive a stronger stance in this sort of theory.

Therefore the approach sketched here is not the only possible and perhaps not even the most genuine way of putting elimination rules first. An elimination rule approach which reverses the conceptual priority between assertions and assumptions would be one, which considers derivations from assumptions to be primary. Such an approach can be obtained by dualizing the I-rule approach by putting “deriving from” rather than “deriving of” in front. One would then develop ideas such as the following: A *closed derivation from A* should be a derivation of absurdity from A (corresponding to the fact that a closed derivation in the standard conception can be viewed as a derivation from truth), and a derivation \mathcal{D} should be justified, if, for every closed valid

derivation $\frac{B}{\mathcal{D}'}$ from B , $\frac{A}{\mathcal{D}}$ is a closed valid derivation from A, etc. This would be in

conflict with the asymmetry of derivations, which usually have exactly one end formula, but possibly more than one open assumption. So full dualization would lead to some variant of a single-premiss/multiple-conclusion logic. A closed derivation from A, in which all downward branches end with absurdity, might be called a closed refutation of A. If one of these branches ends with a formula B different from absurdity, it is an open refutation of A in the sense that replacing B with a closed refutation of it yields a closed refutation of A. Such approaches would lead to rules for logical constants which are dual to the standard ones. Conjunction (as the dual of disjunction) would be the constant that is canonically refuted by a refutation of A as well as by one of B, disjunction (as the dual of conjunction) would be the constant that is canonically refuted by a refutation of both A and B etc. Co-implication would come in as the dual of implication, which is canonically refuted by an open refutation of B to A, i.e., of B given a refutation of A, etc. This leads essentially to an approach in which usual derivation trees are written upside down, the concept of derivation is interchanged with that of refutation etc. It corresponds to a system of dual-intuitionistic logic, which formally corresponds to Brouwer logic, in which connectives are replaced with their duals, and in particular implication by co-implication. However, structurally, the standard approach and its dual are the same — writing derivations upside down is not really an essential change. So if we want any conceptual gain from the consideration of dual concepts, we should be able to develop a joint system for both notions. A genuine E-rule approach might be desirable if one wanted to logically elaborate ideas like Popper’s falsificationism by establishing refutation as the basis of reasoning. However, it is still not clear what such an approach should look like formally, and especially how to incorporate both

implication and co-implication in it. At present it is still more wish than reality, although recently, there has been considerable research in logical systems dual to given ones. (See Tranchini, 2011a, and the references therein.)

Definitional reflection as a local approach is already beyond the distinction between introduction rule and elimination rule approaches, as the fundamental rules come in pairs and are related by principles like local cut reduction and relative uniqueness. It is not the case that definitional closure is any more primary than definitional reflection. We are not justifying one set of rules from the other one. However, as indicated at the beginning of this subsection, there is some implicit bias towards introductions since clauses are directed. Definitional closure is interpreted as expressing the direction from definiens to definiendum, and definitional reflection as expressing the opposite direction. Changing this bias and inverting it, would have to be a radical reform of what a definition looks like. We would then have to consider ‘consequential’ clauses which determine the consequences of a given atom, such as

$$\left\{ \begin{array}{l} a \Rightarrow b_1 \\ \vdots \\ a \Rightarrow b_m \end{array} \right.$$

Definitional closure would then express reasoning along these consequential clauses on the left hand side:

$$\frac{\Gamma, b_1 \vdash C}{\Gamma, a \vdash C} \quad \dots \quad \frac{\Gamma, b_m \vdash C}{\Gamma, a \vdash C}$$

and definitional reflection would be a right introduction rule telling that a can be introduced from all possible definitional consequences taken together

$$\frac{\Gamma \vdash b_1 \quad \dots \quad \Gamma \vdash b_m}{\Gamma \vdash a} .$$

To make this approach reasonably expressive, we would have to consider also complex conclusions B_i of consequential clauses rather than just atoms b_i . A multiple conclusion clause

$$a \Rightarrow c_1, \dots, c_n$$

would then be interpreted as a multiple-conclusion clause to be interpreted by a reflection rule like

$$\frac{\Gamma, c_1 \vdash C \quad \dots \quad \Gamma, c_n \vdash C}{\Gamma, a \vdash C} .$$

If we have more than one multiple-conclusion clause, we would have to consider an appropriate list of left-introduction rules. Alternatively, we could just consider single

clauses, but with structural implications (rules) as conclusion, be means we can code in principle, what can be expressed, e.g., by disjunction. Such a clause would, e.g. be

$$a \Rightarrow ((c_1 \Rightarrow p) \dots (c_n \Rightarrow p) \Rightarrow p)$$

for a variable p . Unlike the elimination rule approach for natural deduction which is based on dualizing notions towards a single-premiss / multiple-conclusion logic, this approach dualizes just the notion of a definition, but not the concept of a (sequent-style) derivation, which is, as before, a concept of derivation for single-succedent sequents. A complete dualization using single-antecedent / multiple succedent sequents would trivialize the whole notion by just exchanging the right and left sides both of definitions and of sequents. Using multiple antecedent / multiple-succedent sequents would lead to a framework related to classical logic, still independent of whether one uses standard definitional clauses, or consequential ones. How for an approach based on consequential clauses can lead, is not clear. Apart from its technical difficulties, such a theory loses its connections to logic programming and inductive definitions, so something completely new would have to be worked out.³⁸

5.2 Negation and denial

Standard proof-theoretic semantics is assertion-centred in that assertibility conditions determine the meaning of logical constants. Corresponding to the intuitionistic way of proceeding, the negation $\neg A$ of a formula A is normally understood as implication of absurdity $A \rightarrow \perp$, where \perp is a constant which cannot be asserted, i.e., for which no assertibility condition is defined. This is an ‘indirect’ way of understanding negation. In the literature there has been the discussion of what, following Kutschera (1969), might be called ‘direct’ negation. By that one understands a one-place primitive operator of negation, which cannot be, or at least is not, reduced to the implication of absurdity. It is not classical negation either. It rather obeys rules which dualize the usual rules for the logical constants. Sometimes it is called the “denial” of a sentence, sometimes also “strong negation” (for an overview see Odintsov, 2008). Typical rules for the denial $\sim A$ of A are, for example,

$$\frac{\sim A \quad \sim B}{\sim(A \vee B)} \quad \frac{\sim A}{\sim(A \wedge B)} \quad \frac{\sim B}{\sim(A \wedge B)}.$$

Essentially, the denial rules for an operator correspond to the assertion rules for the dual operator. Several logics of denial have been investigated, in particular Nelson’s logics of “constructible falsity” motivated first by Nelson (1949) with respect to a certain realizability semantics. The main focus has been on his systems later called N3 and N4 which differ with respect to the treatment of contradiction (N4 is N3 without *ex contradictione quodlibet*). It is even possible to consider systems in which direct and

³⁸Some remarks on dual frameworks can be found in Schroeder-Heister (2011c).

indirect negation (i.e., denial and intuitionistic negation) coexist. Using denial any approach to proof-theoretic semantics can be dualized by just exchanging assertion and denial and turning from logical constants to their duals. In doing so, one obtains a system based on refutation (= proof of denial) rather than proof. It can be understood as exposing a Popperian approach to proof-theoretic semantics. In this sense it is related to the elimination based conception of proof-theoretic semantics (section 5.1). Nearest to Nelson’s system N3 comes the approach of a single-assumption / multiple-conclusion variant of natural deduction. From the point of view of denial, a derivation of absurdity from A corresponds to a derivation of the denial of A , thus avoiding the inverting of derivations and the exchange of truth with absurdity. This is particularly suggestive if one uses the denial operator as an external operator which cannot be iterated. As in ‘signed’ tableaux this can always be achieved for the systems N3 and N4, even if it might seem technically inelegant to give up double negation rules in favour of an introduction rule for the denial of a negation. Many recent investigations on dual intuitionistic logic (or Brouwer logic), both from the logical and the lattice-theoretical side, are relevant here. Its potential has not been fully exploited for proof-theoretic semantics. Here we just focus on a different sort of harmony which consideration principles for denial gives rise to.

5.3 Assertion-denial harmony

If we have an external denial operator and compare the rules for assertion and denial, for example for conjunction and disjunction,

$$\frac{A}{A \vee B} \quad \frac{B}{A \vee B} \quad \frac{\sim A \quad \sim B}{\sim(A \vee B)} \qquad \frac{A \quad B}{A \wedge B} \quad \frac{\sim A}{\sim(A \wedge B)} \quad \frac{\sim B}{\sim(A \wedge B)}.$$

then we observe a striking symmetry, which in general terms can be expressed as follows: Denying each defining condition of an expression allows us to deny the expression itself. For the case of conjunction we have to observe that in order to deny the defining condition of $A \wedge B$, it suffices to deny one of the two elements of this defining condition, which means that we have two denial rules. It is obvious that this observation gives rise to a new principle of harmony, which may be called ‘assertion-denial harmony’. Whereas the harmony between introduction and elimination rules can be viewed as an assertion-assumption harmony (the conditions of asserting an expression should be in harmony with the consequences of assuming it), the harmony between assertion and denial rules expresses that the conditions of asserting an expression are in harmony with the conditions of denying it. This principle of harmony could in principle be used with respect to every concept of proof-theoretic semantics. However, as a general principle it has only been formulated for definitional reflection. Therefore we here sketch its fundamental ideas in that framework. Carrying it over to the validity-based approach would be a desideratum (for those who prefer this approach, and who

consider it worthwhile to consider a direct approach to negation rather only the indirect intuitionistic one).

We explain this approach only by an example, as the general case, although not difficult, requires too many notational conventions. As assumptions are not relevant in our context, we do not consider sequents $\Gamma \vdash a$ but just derivations of atoms a . Suppose an atom a is defined by a definition

$$\left\{ \begin{array}{l} a \Leftarrow b, c \\ a \Leftarrow d \\ a \Leftarrow e, f \end{array} \right.$$

Then the rules of definitional closure are, as before,

$$\frac{b}{a} \quad \frac{c}{a} \quad \frac{d}{a} \quad \frac{e}{a} \quad \frac{f}{a}$$

The rules of definitional reflection, which determine the denial $\sim a$ of a would be the following:

$$\frac{\sim b \quad \sim d \quad \sim e}{\sim a} \quad \frac{\sim b \quad \sim d \quad \sim f}{\sim a} \quad \frac{\sim c \quad \sim d \quad \sim e}{\sim a} \quad \frac{\sim c \quad \sim d \quad \sim f}{\sim a} .$$

Obviously, the premisses of each rule represent a way of denying all defining conditions of a , in denying one atom in each defining condition. This generalizes the way in which the denial rules for conjunction and disjunction are formulated.

This idea of assertion-denial harmony can, of course, be combined with assertion-assumption harmony. How these two harmony principles work together and perhaps interact must still be investigated. The idea that to deny something means to deny every defining condition of it, can be further generalized. One direction concerns the means of expression: What happens, when we consider structural implication in the bodies of clauses, and when we consider clauses with variables? Another direction that I want particularly mention here, is the distinction between direct and indirect denial. In the example just given, we have just started with clauses which govern the assertion of clauses, i.e., it is assertion which is *defined*. However, we might consider also clauses, which govern the denial of clauses, i.e. have a denial $\sim a$ as head, as considered in certain extensions of logic programming (Damásio & Pereira, 1998). Definitional reflection with respect to such clauses would then create new assertion rules. If we define, for example, the $\sim a$ by means of the clauses

$$\left\{ \begin{array}{l} \sim a \Leftarrow \sim b, \sim c \\ \sim a \Leftarrow \sim d \end{array} \right.$$

then definitional closure with respect to these clauses we would give us the denial rules

$$\frac{\sim b \quad \sim c}{\sim a} \quad \frac{\sim d}{\sim a}$$

whereas definitional reflection would give us the assertion rules

$$\frac{b \quad d}{a} \quad \frac{c \quad d}{a}$$

based on the principle that denying every deniability condition of a yields the assertion of a . This can even be further generalized if we assume that for some a both assertion clauses and denial clauses might be given in the definition, and that the conditions for assertion or denying a may depend on assertions and denials in a mixed way. One may then say that the denials generated by definitional reflection from assertion clauses is a *secondary denial* as compared to the denials generated directly by definitional closure based on denial clauses, and that the assertions generated by definitional reflection from denial clauses are *secondary assertions* as compared to the assertions generated directly by definitional closure based on assertion clauses. This leads to a framework in which an expression may have both definitional assertibility and deniability conditions, and where definitional assertibility and deniability is distinguished from assertibility and deniability based on definitional reflection. These four forms of judgement and their inferential relationships resemble closely to what one finds in the traditional square of opposition. For further details see Schroeder-Heister (2010b).³⁹

5.4 Harmony and reflection in the sequent calculus

Gentzen's sequent calculus exhibits a symmetry between right and left introduction rules which suggest to look for a harmony principle that makes this symmetry significant to proof-theoretic semantics. The theory of definitional reflection as explained in section 4.3.2 used a sequent-style system. However, there the sequent calculus was considered to be the inference engine which put the meaning given to expressions by means of a clausal definition into action, not rules which themselves defined the meaning of them. To deal with the sequent calculus rules themselves as semantical rules there are at least three options. One is to consider either the right-introduction rules or the left-introduction rules as introduction rules for certain sequents and justify the opposite rules (left-introductions and right-introductions, respectively) with respect to them. This approach uses the inversion principle or the principle of definitional reflection to generate left-introduction from right-introduction rules and conversely. A second approach derives the right- and left-introduction rules from a characterization in the sense of Došen's double line rules (section 4.4.1), which is then read as a definition of some sort. The third approach uses the idea of an interaction between right- and left-introduction rules in the form of a generalized symmetric principle of definitional reflection. All approaches apply to the sequent calculus in its classical form, with possible more than one formula in the succedent of a sequent, including structurally restricted versions as investigated in linear and other logics.

³⁹A similar idea has been independently developed by Zeilberger (2008).

The first approach has been pursued by Campos Sanz and Piecha (2009). As their metalinguistic framework they use the single-conclusion sequent calculus as used in the theory of definitional reflection, whose sequents are, however, hypersequents, i.e. sequents, with object-linguistic sequents in their antecedents and succedent. They consider the right- or left-rules of the object-linguistic intuitionistic or classical sequent calculus as definitional clauses for which hypersequential right- and left-introduction rules are defined. For example, the left \wedge -I rules

$$\frac{\Gamma, A \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta} \quad \frac{\Gamma, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta}$$

are read, at the metalevel, as clauses constituting the hypersequential right-introduction rules

$$\frac{\Sigma \Vdash \Gamma, A \vdash \Delta}{\Sigma \Vdash \Gamma, A \wedge B \vdash \Delta} \quad \frac{\Sigma \Vdash \Gamma, B \vdash \Delta}{\Sigma \Vdash \Gamma, A \wedge B \vdash \Delta}$$

where \Vdash is the meta-level sequent sign and Σ stands for a list of (object-level) sequents. This is then complemented, at the hypersequential meta-level, using definitional reflection, by

$$\frac{\Sigma; \Gamma, A \vdash \Delta \Vdash \Phi \quad \Sigma; \Gamma, B \vdash \Delta \Vdash \Phi}{\Sigma; \Gamma, A \wedge B \vdash \Delta \Vdash \Phi}$$

(where the semicolon separates object-level sequents within hypersequents, and Φ stands for an object-level sequent) from which the object-level right \wedge -I rule in the hypersequential form

$$\Gamma \vdash A, \Delta ; \Gamma \vdash B, \Delta \Vdash \Gamma \vdash A \wedge B, \Delta$$

can be obtained. Analogously, taking the object-linguistic right \wedge -I rules as definitional clauses for sequents, in which conjunction occurs on the left side, the left \wedge -I rules in the hypersequential form

$$\Gamma, A \vdash \Delta \Vdash \Gamma, A \wedge B \vdash \Delta \quad \Gamma, B \vdash \Delta \Vdash \Gamma, A \wedge B \vdash \Delta$$

can be obtained. By metalinguistic inversion the object-level left-introductions can be obtained from the object-level right-introductions, and the object-level right-introductions can be obtained from the object-level left-introductions. This way of proceeding does not establish a harmony between the right-occurrence and the left-occurrence of a logical constant, i.e. between two functions of a logical constant, but between two types of sequents: one in which the constant occurs on the right, and one where it occurs on the left side of the turnstile. However, this is precisely what is intended by this approach: Exhibiting some harmony between two types of rules, not between two functions of a connective, at least not primarily.

The second approach is Sambin et al.’s *Basic Logic* (Sambin et al., 2000). There one starts with rules that are direct inverses of each other, reading them as inferential definitions, and generates the sequent-calculus rules from them. To take the case of disjunction, one starts with the rules

$$\frac{\Gamma, A \vdash \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A \vee B \vdash \Delta} \text{ (formation)} \quad \frac{\Gamma, A \vee B \vdash \Delta}{\Gamma, A \vdash \Delta} \quad \frac{\Gamma, A \vee B \vdash \Delta}{\Gamma, B \vdash \Delta} \text{ (implicit reflection)}$$

which correspond to what Došen calls a double line rule (see section 4.4.1). The left rule, which introduces a conjunction in the conclusion is called a “formation rule”, whereas the right two rules, which eliminate a conjunction from the premiss are called rules of “implicit reflection”. Formation and implicit reflection together are considered a kind of “equation” which needs to be solved. Obviously, the formation rule for disjunction is already the standard left-introduction rule of the sequent calculus. The standard right-introduction rule is obtained from implicit reflection by using trivialization (initial sequents) and cut. This process is what “solving the equation” means. So the standard symmetric sequent calculus rules for a logical connective, which are considered by Sambin et al. as the “definition” of the connective, are not just laid down. They are considered as the explicit solution of an implicit characterization in terms of formation and implicit reflection rules. In this way Sambin et al. reduce the right-left-symmetry in the sequent calculus to an introduction-elimination symmetry, where formation introduces a connective and implicit reflection eliminates it. These introductions and eliminations are, of course, implicit as they introduce and eliminate full sequents in which the connective occurs at a certain place. It is thus related to the approach by Campos Sanz and Piecha. We have here taken the case of additive conjunction as an example. Sambin et al. are able to manage in a single framework the whole variety of multiplicative and additive connectives of linear logic, with substructural distinction of various kinds, containing classical, intuitionistic and even quantum logic as special cases. However, it should be noted that we cannot arbitrarily choose the right- or left-introductions as our starting point, as formation inferences must always consist of a single inference to be invertible. So in the case of conjunction, one can only use the right-introductions as formation rules, whereas for disjunction only the left-introductions.⁴⁰

The third approach uses a generalization of the principle of definitional reflection. In the assertion-assumption case this principle was based on the idea of consequence: Every consequence of each defining condition of a is a consequence of a itself (section 4.3.2). In the assertion-denial case it was based on the idea of deniability: a can be denied if every defining condition of a can be denied (section 5.2), i.e., denials of each

⁴⁰More precisely, for additive conjunction and multiplicative disjunction, one can only use the right-introductions as formation rules, and for additive disjunction and multiplicative conjunction only the left-introductions.

defining condition of a (jointly) imply the denial of a . Now we define the notion of a *complement* of a and require that complements of each defining condition of a taken together (jointly) imply the complement of a . a is now a sequent containing a certain expression. Denoting the complement by $*$, we associate with a set of definitional rules

$$\frac{\Sigma_1}{S} \quad \dots \quad \frac{\Sigma_n}{S}$$

for a sequent S one or more complementary inference rules of the form

$$\frac{\Sigma_1^* \quad \dots \quad \Sigma_n^*}{S^*}.$$

The definitional rules are sequent-style right- or left-introduction rules for a logical constant, or, more generally, sequent-style rules for an n -ary constant α . In the conclusion of a definitional rule for α the expression $\alpha(A_1, \dots, A_n)$ occurs either on the left or on the right side of the turnstile, whereas in the premisses only sequents containing its arguments A_1, \dots, A_n are allowed to occur, in addition to context variables $\Gamma, \Gamma_i, \Delta, \Delta_i$. The position and distribution of these context variables over premisses and conclusion indicate whether premisses are associated additively, conjunctively or in a mixture of these two modes. The complement of a sequent or a list of sequents Σ is defined as a sequent or list of sequents Σ^* , such that from Σ and Σ^* a sequent containing only the context variables of Σ and Σ^* in their respective positions (right or left of the turnstile) can be obtained by means of cut. For example, $\Gamma_1 \vdash A, \Delta_1$ is complementary to $\Gamma_2, A \vdash \Delta_2$, as from them the sequent $\Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2$ can be obtained by means of cut. Or the sequent $\Gamma, A_1, A_2 \vdash B, \Delta$ is complementary to the set of sequents $\{(\Gamma_1, \vdash A_1, \Delta_1), (\Gamma_2, \vdash A_2, \Delta_2), (\Gamma_3, B \vdash \Delta_3)\}$, as from these by means of cut the sequent $\Gamma, \Gamma_1, \Gamma_2, \Gamma_3 \vdash \Delta, \Delta_1, \Delta_2, \Delta_3$ can be obtained. It is then observed that the premisses of the left- and right- introduction rules for the standard connectives are complementary with each other, which means that left-introduction rules naturally complement the right-introduction rules and vice versa. For example, since both $\{\Gamma_1 \vdash A, \Delta_1\}$ and $\{\Gamma_2 \vdash B, \Delta_2\}$ complement $\{(\Gamma, A \vdash \Delta), (\Gamma, B \vdash \Delta)\}$, they can be used as premisses of right-introduction rules for disjunction

$$\frac{\Gamma_1 \vdash A, \Delta_1}{\Gamma_1 \vdash A \vee B, \Delta_1} \quad \frac{\Gamma_2 \vdash B, \Delta_2}{\Gamma_2 \vdash A \vee B, \Delta_2} \quad 41$$

complementing the left-introduction rule

$$\frac{\Gamma, A \vdash \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A \vee B \vdash \Delta}.$$

(For all details see Schroeder-Heister, 2011a.) Unlike *BasicLogic*, we can choose either the left- or the right-introduction rules for a constant as our starting point. Unlike

⁴¹The indices of Γ and Δ are, of course, irrelevant in the formulation of the individual rules

the approach of Campos Sanz and Piecha, we do not need the heavy instrument of hypersequents. What we are essentially doing, is turning the idea of local cut reduction into a semantical principle. The fact that a cut using the conclusions of right- and left-introductions can be reduced to a cut using their premisses (see section 4.3.2), which was originally used merely as an adequacy condition, corresponds to using complementation as an operation for definitional reflection.⁴² If one wanted to emphasize the general character of definitional reflection, which is used in the context of the *symmetric* notion of complementation, one might speak of *definitional interaction* instead of definitional reflection, as this more appropriately expresses the symmetry in this application.

5.5 Classical logic

Proof-theoretic semantics is intuitionistically biased. This is due to the fact that natural deduction as its preferred framework has certain features which make it particularly suited for intuitionistic logic. This bias pertains to semantics using the sequent calculus as long as it uses a single-succedent system. In classical natural deduction one normally replaces the *ex falso quodlibet*

$$\frac{\perp}{A}$$

with the rule of *classical reductio ad absurdum*

$$\frac{[A \rightarrow \perp]}{\frac{\perp}{A}}$$

This rule destroys several properties of the intuitionistic system:

1. In allowing to discharge $A \rightarrow \perp$ in order to infer A , it destroys the *subformula principle*.
2. In containing both \perp and $A \rightarrow \perp$, it refers to two different logical constants, so there is no *separation* of logical constants, i.e. we no longer have exactly one constant per inference schema.
3. As an elimination rule for \perp is falls out of the general pattern of introductions and eliminations. As a consequence, it destroys the *introduction form property* that every closed derivation can be reduced to one which uses an introduction rule in the last step.

⁴²It is not possible to generate for every given condition an appropriate complement, which is due to the fact that the criterion of (relative) uniqueness (see section 2.7) is not necessarily fulfilled. This is resolved by restricting the form of definitional rules in a certain way, which means that certain constants must be defined in two steps, by first defining auxiliary constants and then defining the final constants in terms of the auxiliary ones. See Schroeder-Heister (2011a).

All these properties are relevant to proof-theoretic semantics based on natural deduction or on a single-succedent sequent system. Property 3 is the most important one, since the idea that introduction rules are the exclusive meaning-giving inferences hinges on it. In the validity-based conceptions this is Dummett's "fundamental assumption" (see section 2.5). In the theory of definitional reflection it is the feature that definitional clauses are the only way of arriving at certain expressions, which is what justifies their inversion by means of the reflection principle. So it is no surprise that proof-theoretic semanticists often see their enterprise as a justification of intuitionistic logic rather than as opposed to classical logic.

As is well known, classical logic fits very well with the multiple-succedent sequent calculus. There we do not need any additional principles beyond those assumed for the intuitionistic case. Just the structural feature of allowing for more than one formula in the succedent suffices to obtain classical logic. And as there are plausible approaches to establish a perfect harmony between right-introductions and left-introduction in the sequent calculus — three of them have been mentioned in section 5.4 —, classical logic seems to be perfectly justified that way. Now these approaches to establish a harmony between right- and left-introduction are quite independent of the special structural properties assumed for the sequent calculus, so they yield a justification of any other logic as well (as long as it is appropriately framed in the sequent calculus). This is not a real problem, as we might discard the problem of singling out a 'true' logic as a foundationist misconception and just look for a semantically plausible framework allowing us to deal with alternative logical systems. For that purpose sequent-style systems with the facility to change their structural assumptions without changing logical inferences are perfectly suited.

However, this is only convincing if we consider the sequent calculus as a formal system. For classical logic as a 'real' reasoning system there remains the problem of how to use it in practice. Normally, deductive reasoning starts from no, from one, or from several assumptions and proceeds to a single conclusion. Reasoning according to the rules of the multiple-succedent sequent calculus would correspond to a multiple-conclusion way of proceeding, which does not correspond to our standard reasoning practice. One could try to develop an appropriate intuition by arguing that reasoning towards multiple conclusions delineates the area in which truth lies rather than establishing a single proposition as true. However, this intuition is difficult to develop and cannot be captured without serious technical difficulties, as philosophical approaches such as those by Shoesmith and Smiley (1978) and proof-theoretic approaches such as proof-nets (see Girard, 1987; Di Cosmo & Miller, 2010) demonstrate. Therefore it is only natural to look for an interpretation of classical logic in the ordinary single-conclusion framework of natural deduction.

A fundamental reason for the failure of the introduction form property in classical logic is the indeterminism inherent in the laws for disjunction. $A \vee B$ can be inferred from either A and B . Therefore, if the disjunction laws were the only way of inferring $A \vee B$, the derivability of $A \vee \neg A$, which is a key principle of classical logic, would entail that of either A or of $\neg A$, which is absurd. A way out of this difficulty is to abolish indeterministic disjunction and use instead its classical de Morgan equivalent $\neg(\neg A \wedge \neg B)$. This leads to the following introduction rule for disjunction:

$$\frac{[A \rightarrow \perp] \quad [B \rightarrow \perp]}{\perp} \frac{\perp}{A \vee B} .$$

This is still a rule without separation, i.e. with other constants occurring beyond the one introduced. However, we can do without \perp , if we consider absurdity to be some sort of structural constant $\#$, which is a marker in a proof but not a logical constant with introduction and elimination rules (representing a “logical dead-end” in the sense of Tennant, 1999). If we then use the machinery of rules of higher levels, replacing the implication arrow \rightarrow with the structural rule arrow \Rightarrow , we obtain

$$\frac{[A \Rightarrow \#] \quad [B \Rightarrow \#]}{\#} \frac{\#}{A \vee B} .$$

which is an introduction rule, in whose premiss, beyond structural expressions, only the letters A and B for the immediate subformulas of $A \vee B$ occur, which is a proper inference rule in the sense of proof-theoretic semantic, which even satisfies separation.

Its corresponding elimination rule according to the schema for generalized propositional operators (see section 4.1.3) is then:

$$\frac{A \vee B \quad \frac{[(A \Rightarrow \#), (B \Rightarrow \#)] \Rightarrow \#}{C}}{C}$$

which can easily shown to be equivalent to

$$\frac{A \vee B \quad \frac{[A] \quad [B]}{\#}}{\#} .$$

The problematic *reductio* principle can now be formulated as

$$\frac{[A \Rightarrow \#]}{\#} \frac{\#}{A} .$$

If it is restricted to atomic formulas A , it can be considered to be a general assumption about the behaviour of atomic expressions, having nothing to do with the meaning of complex formulas. We simply assume that classical logic is the logic over atomic

systems which behave such as to obey this principle. That we do not lose anything by this restriction of A to atoms, if we do not use nondeterministic disjunction, but its classical deterministic counterpart, has already been observed by Prawitz (1965), where he proves full normalization for a classical system in this restricted sense.

When classical logic is treated along these lines, all three critical principles mentioned above remain valid: Subformula principle, separation, and introduction form principle. We retain the full power of generalized introduction and elimination rules for n -ary propositional operators, as long as we consider only the deterministic case, i.e., operators with exactly one introduction rule (as done in Schroeder-Heister, 1981). This situation is not fundamentally altered by the later result by Stålmårck (1991) who showed normalization for classical logic with genuine (nondeterministic) disjunction and existential quantification. Even if we then have than standard introduction and elimination rules, the fact that classical *reductio* cannot be restricted to the nonatomic case means that we have an additional rule for complex formulas and not just something that describes our atomic domain of discourse.

This approach can be extended to a clausal definition. If a is defined as

$$\left\{ \begin{array}{l} a \Leftarrow B_1 \\ \vdots \\ a \Leftarrow B_n \end{array} \right.$$

then its introduction and elimination inferences would be

$$\frac{[B_1 \Rightarrow \#] \quad \dots \quad [B_n \Rightarrow \#]}{\#} \quad \frac{[B_1] \quad [B_n]}{a \quad \# \quad \dots \quad \#} \quad \#$$

The atomicisation of classical *reductio* now means that those atoms, for which no definitional clauses are given, obey *reductio*. The elimination inference resembles definitional reflection for denial: If every defining condition B_i of a can be denied, then so can a itself. The introduction rule transforms the nondeterministic conditions for a into a deterministic one: a can be asserted exactly when the denials of all defining conditions of a can be (jointly) denied. The introduction rule allows one to introduce a under weaker conditions than in the standard nondeterministic (intuitionistic) case. Correspondingly, the elimination rule only allows one to infer less than in the standard case from all defining conditions of a , namely only $\#$. a does not express, as in the standard case, the common content of B_1, \dots, B_n , but only the content of $((B_1 \Rightarrow \#), \dots, (B_n \Rightarrow \#)) \Rightarrow \#$. But nevertheless we have the local reduction of maximum formulas (or local cut reduction in a sequent-style framework), as well as uniqueness (provided *reductio* holds in our domain of discourse).

This seems to us to be as close to the spirit of classical logic as one can get in the framework of proof-theoretic semantics for natural deduction or single-succedent

sequent calculus, since it essentially keeps the relationship between introduction and elimination rules by means of some sort of inversion or reflection. It also builds on the idea that classical logic is concerned with specific domains, where atomicized *reductio* holds. What it gives up is the idea of genuinely nondeterministic introduction rules which is typical of intuitionism.

There are other approaches towards harmony principles for classical logic which depart from the idea that a constant comes with introductions and eliminations, or is introduced either in assertion or in assumption position. The most recent one is by Milne (2010), where implication receives the rules

$$\frac{[A \rightarrow B] \quad B}{C} \quad \frac{[A] \quad [A \rightarrow B] \quad C}{C} \quad \frac{A \rightarrow B \quad A}{B} .$$

The first two rules are considered introduction rules, as they introduce implication (albeit as an assumption), the right one is the common elimination rule. Formally, these rules satisfy strong adequacy conditions (reducibility of maximal formulas, separation, and uniqueness). To consider them to be a pair of rules that are related with one another by means of some sort of inversion, some additional intuition is needed.

An interesting point on the relationship between intuitionistically inspired proof-theoretic semantics and classical logic has been raised by Sandqvist (2009). He claims that validity-based proof-theoretic semantics can prove certain principles of classical logic, notably the double negation law. His approach builds on the way atomic systems are conceived in proof-theoretic semantics, and on the way in which extensions of atomic systems enter the definition of validity for derivations of implicational formulas.

The strength of intuitionistic logic from the standpoint of classical logic has recently been characterized by Humberstone and Makinson (2011). They show that intuitionistic logic is the weakest logic which is sufficient to prove all classically valid schematic rules for logical constants, which are elementary in the following sense: They contain only one occurrence of a logical constant, and otherwise only schematic letters for its arguments. In establishing this result, Humberstone and Makinson essentially show that the feature of separation, which is embodied in the notion of an elementary rule, is characteristic of standard intuitionistic logic.

5.6 A general perspective: The categorical and the hypothetical

Standard approaches to proof-theoretic semantics, especially Prawitz’s validity-based approach (section 4.2.2), take closed derivations as basic. The validity of open derivations is defined as the transmission of validity from closed derivations of the assumptions to a closed derivation of the assertion, where the latter is obtained by substituting the open assumptions with their closed derivations. Therefore, if one calls closed derivations ‘categorical’ and open derivations ‘hypothetical’, one may characterize this approach as following two fundamental ideas:

- (I) The primacy of the categorical over the hypothetical
- (II) the transmission view of consequence.

I have called these two assumptions (I) and (II) as *two dogmas of standard semantics* (see Schroeder-Heister, 2008b, 2010a). “Standard semantics” here not only means standard proof-theoretic semantics, but also classical model-theoretic semantics, where these dogmas are assumed as well. There one starts with the definition of truth, which is the categorical concept, and defines consequence, the hypothetical concept, as the transmission of truth from conditions to consequent. From this point of view, constructive semantics, including proof-theoretic semantics, exchange the concept of truth with a concept of construction or proof, and interpret “transmission” in terms of a constructive function or procedure, but otherwise leave the framework untouched.

There is nothing wrong in principle with these two dogmas. However, there are phenomena that are difficult to deal with in the standard framework. Such a phenomenon is non-wellfoundedness, especially circularity, where we may have consequences without transmission of truth and provability. Another phenomenon are substructural distinctions, where it is crucial to include the structuring of assumptions from the very beginning (we have not discussed this latter point here). Moreover, and this is most crucial, we might define things in a certain way without knowing in advance of whether our definition or chain of definitions is well-founded or not. We do not first involve ourselves into the metalinguistic study of the definition we start with, but would like to start to reason immediately. This problem does not obtain if we restrict ourselves to the case of logical constants, where the defining rules are trivially well-founded. But the problem arises immediately, when we consider more complicated cases that go beyond logical constants.

This makes it worthwhile to proceed in the other direction and start with the hypothetical concept of consequence, i.e., characterize consequence directly without reducing it to the categorical case. Philosophically this means that the categorical concept is a limiting concept of the hypothetical one. In the classical case, truth would be a limiting case of consequence, namely consequence without hypotheses. How this could be defined, is not the topic of this paper. In the proof-theoretic case this leads to the conceptual priority of the sequent calculus viewed as a calculus modelling consequence⁴³. Giving up the second dogma means to develop a direct intuition for consequence rather than reducing it to some other concept. Consequence would be directly defined rather than justified in terms of transmission procedures, which in proof-theoretic semantics are essentially reductions, i.e. derivation reduction systems \mathcal{J} as considered in the definition of validity. This motivates rules which directly characterize what consequence

⁴³Alternatively, we could, of course, use bidirectional natural deduction, which is a natural-deduction image of the sequent calculus, see Schroeder-Heister, 2009.

should be.

If this analysis is right, it speaks in favour of the local rule-based approach (section 4.1) and concepts related to structural characterizations (4.4.1) or definitional reflection (section 4.3.2), and against the dominating proof-based notions put forward by Dummett, Martin-Löf and Prawitz. In general it shows that the opposition between truth and proof on one side, and consequence on the other, is as important as the opposition between truth and proof, and has strong bearings on the format of a semantic theory, whether it is proof-theoretic or not.

6 Miscellaneous and outlook

I give a nonexhaustive list of further points that should be discussed in more detail.

Validity for sequent-calculus derivations. The two core conceptions we have presented were, as a global framework, Prawitz’s validity definition, and, as a local framework, definitional reflection. The first conception was developed within natural-deduction-style notion of deduction, the second one within a sequent-style framework. An approach towards validity for sequent-style rather than natural-deduction-style derivations which uses both the right and the left introduction rules as meaning constituting, might be sketched as follows: A derivation is valid if (i) it uses a right-introduction rule in the last step, or (ii) it uses an an elimination rule in the last step, or (iii) it reduces to one of these cases. The reductions according to this definition of validity correspond to the cut reductions. Translated into natural deduction, by using generalized elimination rules, this corresponds to a definition of validity recently proposed by Dyckhoff and Francez. It can also be seen as pursuing the idea of ‘bidirectional natural deduction’, in which natural deduction derivations can be extended both to the bottom and to the top. See Francez and Dyckhoff (2008); Schroeder-Heister (2009).

Subatomic derivations. If proof-theoretic semantics should be successful, it must be able to take the internal structure of atomic sentences into account. This idea has been put forward by Więckowski (2008, 2011), who developed a proof-theoretic approach with introduction and elimination rules for atomic sentences, where these atomic sentences are not just reduced to other atomic sentences, but to subatomic expressions representing the meaning of predicates and individual names. This can be seen as a proof-theoretic semantics of atomic expressions which goes beyond the common view according to which the meaning of atomic expressions is defined by an atomic system. This common view is essentially a generalization of the view that atomic sentences receive their truth-value by valuation. Even if, as in definitional reflection, we are considering definitional rules for atoms, the defining conditions do not decompose atoms.

Speaker-meaning vs. hearer-meaning. Francez (2010) has proposed a linguistic theory, in which he associates the canonical conditions of an expression with its speaker-meaning, and the canonical consequences with its hearer-meaning. According to Francez, the harmony between the two aspects, which logically is expressed by validity or definitional reflection, is established by the lossless communication between speaker and hearer. Logically, in terms of the sequent calculus, this means that cut is viewed as a kind of communication channel, by means of which, in the case of full harmony, nothing is gained or lost. Using this approach he is able to develop a proof-theoretic semantics of a fragment of English. The fact that speaker-meaning and hearer-meaning are on the same level, suggests a principle of harmony according to which both introduction and elimination rules are meaning-constituting (see above, and Francez & Dyckhoff, 2008).

Quantifiers. We have practically exclusively dealt with propositional logic, and here often only with the implicational fragment, in order to make matters of principle clear. Only in the context of definitional reflection we have mentioned definitions with individual variables in clauses. This should not suggest that the consideration of quantifiers is a straightforward and easy matter. The handling of variables and substitutions is already complicated, and if it comes to the binding of variables, it will be even more difficult. Systems dealing with advanced quantifier logic from the point of view of proof-theoretic semantics are those based on Martin-Löf type theory. Dealing with these theories means also dealing with the meaning of individual expressions, which is a much neglected topic in proof-theoretic semantics, in contradistinction to traditional denotational semantics which starts with the denotations of terms. Also, one would have to include the idea of dependent types, which give type theories their real strength. The restriction to propositional logic is a virtue only if is considered a preparatory step towards a more encompassing theory.

Inductive definitions. Reading texts in proof-theoretic semantics give the impressions as if introduction and elimination rules (or right introductions and left introductions) are the only type or rule to be considered, and that everything else can be reduced to this pattern. This is not true, at least not in its generality. Mathematics uses inference principles that go beyond this pattern, if introductions and eliminations are understood in the limited sense in which they occur in propositional logic. Powerful induction principles are of this kind. Although the definitional reflection is closely related to ideas in the theory of inductive definitions, the latter go far beyond what can be accomplished by the elementary means of expression we have been considering. Already in Lorenzen (1955) the inversion principle is just one of several principles to establish admissibility (among those in particular induction as a principle of its

own), and his development of mathematics proceeds by a transfinite iteration of the construction of language levels. Again, proof-theoretic semantics at the propositional level can only be the beginning.

Modal logic. Already at the propositional level there are systems which transcend the means we have considered for proof-theoretic semantics. Modal logic in its various forms is here prominent. The topic of finding plausible natural-deduction or sequent systems for common systems such as S4 or S5 is not without problems. One approach to deal with the non-locality of certain modal inference rules is to consider hypersequents, which allow one to interpret the necessity $\Box A$ as expressing $\vdash A$ at the object-linguistic level. Most powerful tools are recent considerations of sequents which have a tree structure (“nested sequents”, “tree-hypersequents”, see Brünnler, 2010; Poggiolesi, 2010).

Closed versus open proofs. It is one of the basic tenets of the theory of definitional reflection as a local theory of rules that the traditional preoccupation with closed proofs is ill-guided. Unlike the validity-based concepts which start from closed proofs and adhere, with respect to the interpretation of open proofs, on the placeholder view of assumptions and the transmission view of consequence, definitional reflection deals with open proofs from the very beginning. In a different framework, Martin-Löf (2009) has considered the evaluation of open expressions, based on the observation that in mathematical proof theory, the evaluation of open expressions has reached a more advanced stage than in the early 1970s, when the key concepts of proof-theoretic semantics were framed.

Substructural issues. A wide variety of substructural logics is considered in harmony and reflection principles for the sequent calculus (section 5.4). Frameworks such as *Basic Logic* have been explicitly designed to deal with these issues. This is only natural if one considers whole sequents as premisses and conclusions of rules. However, even in the normal framework of definitional reflection substructural distinctions can be made by extending the concept of a definitional clause. If we allow for clauses which in their bodies, in addition to the comma (and perhaps structural implication) may have other dividers, certain substructural systems can be developed in a very general way. An example is relevant logic as a logic of bunched implication (see Schroeder-Heister, 1987, 1991b, for bunched implications in general O’Hearn & Pym, 1999).

Structural background logic and the logic of implication. As remarked several times, in a proof-theoretic semantics of complex propositions, or of atoms deductively behaving like those, we sometimes use a structural concept of implication, which is conceptually

prior to the presentation of definitional clauses or inference schemata, as it is needed to formulate them. We have written it using the arrow “ \Rightarrow ”. It is naturally understood as expressing a rule which can be assumed and asserted in the sense of the theory of higher-level rules (section 4.1.3). The idea of structural implications as rules gives rise to a notion of implication in the sequent calculus, with a left introduction schema slightly weaker than Gentzen’s, which is narrower to implication in natural deduction (see Schroeder-Heister, 2011b).

The alleged primacy of logical constants. Standard proof-theoretic semantics has practically exclusively been occupied with logical constants. This is particularly true for the validity-based conceptions (section 4.2.2). In the rule-based conceptions this preference is not so strong. Especially, in the theory of definitional reflection, logical constants are a limiting case of expressions inferentially defined. Logical constants are often considered primary, as our fundamental inference machinery is based on it. This latter claim can, however, be questioned. According to definitional reflection, we have local definitions that define the expressions we want to deal with, and every reasoning is reasoning with respect to local definitions. Thus this approach calls a fundamental tenet of theoretizing since Aristotle into question, namely that there is one universal inference engine — formal logic—, and that every non-logical content is put into assumptions from which we draw logical conclusions. So if one wants to give some general relevance to certain investigations within proof-theoretic semantics, one might say that the idea of reasoning with respect to specific local definitions challenges an important ingredient of our model of rationality.

Dialogue semantics. The semantics in terms of dialogues and games has received quite some attention during the last two decades. It is a rival of proof-theoretic semantics, and in some of its versions, in particular those who go back to Lorenzen’s and Lorenz’s theory, also a rival to model-theoretic semantics. In fact, the dialogical / game-theoretical approach was developed by Lorenzen in the late 1950s (see Lorenzen, 1960), as he saw serious problems with his own proof-theoretic semantics (see section 4.1.1). This transition to dialogical logics, which was later jointly developed by Lorenzen and Lorenz, is described in Lorenz (2001). Dialogical semantics distinguishes between the levels of individual plays and winning strategies. The explanation of meaning happens at the level of plays, not of winning strategies. From the point of view of proof-theoretic semantics, the level of proofs corresponds to that of winning strategies. This means that with the level of plays, the dialogical approach has a conceptual layer which has no obvious analogue in proof-theoretic semantics. Proponents of the dialogical approach often see the advantage of their theory in the availability of this layer and its significance to semantics. Within proof-theoretic semantics this point has not been

seriously discussed so far, and in particular not whether a dialogical approach can solve certain shortcomings of the proof-theoretic approach. Dialogical semantics is more tied to the sequent calculus than to natural deduction, with the proponent and opponent of a game corresponding to the two sides of a sequent. It is therefore not committed to the placeholder view of assumptions and the transmission view of consequence which proof-theoretic semantics adheres to in validity-based semantics (and in most global conceptions related to the BHK interpretation).

Final remark. Our concentration on harmony, inversion principles, definitional reflection and the like might be misleading, as it suggests that proof-theoretic semantics consists of only that. Though this might be right for elementary systems such as first-order logic, it must be re-emphasized that already when it comes to arithmetic, stronger principles are needed in addition to inversion. Present proof-theoretic semantics is very much occupied with elementary logic and its particular constants. In the long run, without detaching itself from this restriction, it will never become a full-fledged alternative to model-theory. What counts in the end is the explanatory power of a theory with respect to the relevant phenomena, and they comprise not just the constants of propositional or first-order logic.

A Appendix: Gentzen-style natural deduction and sequent calculus

The rules of first-order intuitionistic natural deduction can be stated as follows:

$$\begin{array}{c}
 \frac{A \quad B}{A \wedge B} \\
 \\
 \frac{A}{A \vee B} \quad \frac{B}{A \vee B} \\
 \\
 \frac{[A] \quad B}{A \rightarrow C} \\
 \\
 \frac{A(y)}{\forall x A(x)} \\
 \\
 \frac{A(t)}{\exists x A(x)}
 \end{array}
 \qquad
 \begin{array}{c}
 \frac{A \wedge B}{A} \quad \frac{A \wedge B}{B} \\
 \\
 \frac{[A] \quad [B] \quad C}{A \vee B \quad C} \\
 \\
 \frac{A \rightarrow B \quad A}{C} \\
 \\
 \frac{\perp}{A} \text{ (intuitionistic case)} \\
 \\
 \frac{\forall x A(x)}{A(t)} \\
 \\
 \frac{[A(z)] \quad C}{\exists x A(x) \quad C}
 \end{array}$$

where the eigenvariable y is not free in any assumption, on which $A(y)$ depends, and the eigenvariable z is not free in any assumption except the displayed assumption $A(z)$. The leftmost premiss in an elimination inference is called its major premiss, whereas the other premisses are called its minor premisses. Assumptions which can be discharged at the application of the rule in question are indicated by square brackets. In the classical system, the *ex falso quodlibet* rule for absurdity is replaced with that for classical *reductio ad absurdum*

$$\begin{array}{c}
 [\neg A] \\
 \frac{\perp}{A}
 \end{array}$$

with $\neg A$ being an abbreviation for $A \rightarrow \perp$.

Following Prawitz, we use the following notation: If a derivation \mathcal{D} ends with A , we also write $\frac{\mathcal{D}}{A}$, if it also depends on an assumption B , we also write $\frac{B}{\mathcal{D}}$ or $\frac{B}{\mathcal{D}, A}$. This

means that the notations \mathcal{D} , $\frac{\mathcal{D}}{A}$, $\frac{B}{\mathcal{D}}$ and $\frac{B}{\mathcal{D}, A}$ do not denote different derivations, but just differ in what they make explicit. A maximum formula is a formula occurrence which is the conclusion of an application of an introduction rule and at the same time major premiss of an application of an elimination rule. The main reduction steps devised by Prawitz (1965) to remove maximum formulas (and therefore detours) are as follows.

$$\begin{array}{l}
 sr(\wedge) : \quad \frac{\frac{\mathcal{D}_1 \quad \mathcal{D}_1}{A_1 \quad A_2}}{A_1 \wedge A_2}}{A_i} \quad \text{reduces to} \quad \frac{\mathcal{D}_i}{A_i} \quad (i = 1, 2) \\
 \\
 sr(\vee) : \quad \frac{\frac{\mathcal{D} \quad [A_1] \quad [A_2]}{A_i \quad \mathcal{D}_1 \quad \mathcal{D}_2}}{A_1 \vee A_2} \quad C}{C} \quad \text{reduces to} \quad \frac{\mathcal{D} \quad A_i}{\mathcal{D}_i} \quad (i = 1, 2) \\
 \\
 sr(\rightarrow) : \quad \frac{\frac{A \quad \mathcal{D}}{B} \quad \mathcal{D}'}{A \rightarrow B} \quad A}{B} \quad \text{reduces to} \quad \frac{\mathcal{D}'}{A} \\
 \\
 sr(\forall) : \quad \frac{\frac{\mathcal{D}(y) \quad A(y)}{\forall x A(x)}}{A(t)} \quad \text{reduces to} \quad \frac{\mathcal{D}(t)}{A(t)} \\
 \\
 sr(\exists) : \quad \frac{\frac{\mathcal{D} \quad [A(z)]}{A(t)} \quad \mathcal{D}'(z)}{\exists x A(x)} \quad C}{C} \quad \text{reduces to} \quad \frac{\mathcal{D} \quad A(t)}{\mathcal{D}'(t)} \\
 \\
 \end{array}$$

Here “*sr*” stands for “standard reduction”. In order to prove full normalization for intuitionistic logic, additional ‘permutative’ reductions are needed, which affect the global structure of derivations. They are due to the fact that there is the possibility of maximum *segments*, which consist of sequences of identical formulas in a branch of a derivation, beginning with a conclusion of an I rule, passing through minor premisses of \vee or \exists elimination rules and ending with the major premiss of an elimination rule.

Prawitz then shows that by iterated application of reduction steps, every derivation in intuitionistic logic can be normalized, i.e., can be rewritten to a derivation in normal form. A corollary of this result, which is fundamental for Dummett-Prawitz-style proof-theoretic semantics, is that every closed derivation in intuitionistic logic can be reduced to one using an introduction rule in the last step, as a closed normal derivation is of exactly that form. This property is here called the *introduction form property*. In a more philosophical context, Dummett calls it the *fundamental assumption* (Dummett, 1991, p. 254 and Ch. 12).

The normalization result mentioned is also called *weak normalization*. The *strong normalization* result says that any reduction sequence terminates in a normal derivation, no matter in which order reduction steps are applied. The standard methods used to prove strong normalization, i.e., by using computability predicates (see section 2.6),

are closely related to methods used in proof-theoretic semantics.

Gentzen's sequent calculus for first-order logic can be stated as follows:

$$\begin{array}{c}
\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B} \quad \frac{\Gamma \vdash B}{\Gamma \vdash A \vee B} \qquad \frac{\Gamma, A \vdash C \quad \Delta, B \vdash C}{\Gamma, \Delta, A \vee B \vdash C} \\
\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B} \qquad \frac{\Gamma, A \vdash C}{\Gamma, A \wedge B \vdash C} \quad \frac{\Gamma, B \vdash C}{\Gamma, A \wedge B \vdash C} \\
\frac{\Gamma, A \vdash B}{\Gamma \vdash A \rightarrow B} \qquad \frac{\Gamma \vdash A \quad \Delta, B \vdash C}{\Gamma, \Delta, A \rightarrow B \vdash C} \\
\overline{\Gamma, \perp \vdash C} \\
\frac{\Gamma \vdash A(y)}{\Gamma \vdash \forall x A(x)} \qquad \frac{\Gamma, A(t) \vdash C}{\Gamma, \forall x A(x) \vdash C} \\
\frac{\Gamma \vdash A(t)}{\Gamma \vdash \exists x A(x)} \qquad \frac{\Gamma, A(y) \vdash C}{\Gamma, \exists x A(x) \vdash C}
\end{array}$$

where the eigenvariable y is not free in the conclusion of the rule. The crucial feature of this system is that it has introductions of connectives on the right side (right introductions) and on the left side (left introductions) of the sequent sign (here the turnstile). Whereas the right introductions correspond to the introduction rules in natural deduction, the left introductions have no direct counterpart. Left introductions in the sequent calculus and elimination rules in natural deductions rest on different ideas, even if, by means of certain transformations, they can be related to each other. The classical case is not dealt with by changing the laws for negation and absurdity, but by the structural change of allowing for succedents consisting of more than one formula. This yields an elegant formulation of classical logic which makes the sequent calculus particularly suited for it. However, in order to use this technical feature for a proof-theoretic semantics of classical logic, a philosophically plausible interpretation of multiple-formulae succedents is required.

Natural deduction can be given a sequent-style formulation, so-called 'sequent-style natural deduction', where one uses the antecedent to list the assumptions on which the succedent depends. In such a formulation, which is a notational variant of 'standard' natural deduction, one has elimination inferences such as, for example,

$$\frac{\Gamma \vdash A \rightarrow B \quad \Delta \vdash A}{\Gamma, \Delta \vdash B}.$$

The advantage of sequent-style natural deduction is that one can make substructural distinctions, i.e., the way assumptions are associated, fully explicit.

In the genuine (or 'symmetric') sequent calculus the rule of cut

$$\frac{\Gamma \vdash A \quad \Delta, A \vdash B}{\Gamma, \Delta \vdash B}$$

is eliminable. As in natural deduction, certain reductions are applied to achieve this goal, where the main reductions in natural deduction now correspond to cut reduction in the case where the right and the left premiss of the cut have been introduced by respective introduction inferences and are reduced to a cut with their premisses. In the case of implication and disjunction, these reductions are as follows:

$$\frac{\frac{\Gamma, A \vdash B}{\Gamma \vdash A \rightarrow B} \quad \frac{\Delta_1 \vdash A \quad \Delta_2, B \vdash C}{\Delta_1, \Delta_2, A \rightarrow B \vdash C}}{\Gamma, \Delta_1, \Delta_2 \vdash C} \quad \text{reduces to} \quad \frac{\frac{\Delta_1 \vdash A \quad \Gamma, A \vdash B}{\Gamma, \Delta_1 \vdash B} \quad \Delta_2, B \vdash C}{\Gamma, \Delta_1, \Delta_2 \vdash C}$$

$$\frac{\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B} \quad \frac{\Delta, A \vdash C \quad \Delta, B \vdash C}{\Delta, A \vee B \vdash C}}{\Gamma, \Delta \vdash C} \quad \text{reduces to} \quad \frac{\Gamma \vdash A \quad \Delta, A \vdash C}{\Gamma, \Delta \vdash C}$$

Again, as in natural deduction, certain permutative reductions have to be performed to reach a full cut-elimination theorem. In order to reach a strong cut elimination theorem for the sequent calculus (corresponding to strong normalization in natural deduction), additional concepts have so be introduced (see Sørensen & Urzyczyn, 2006, Ch. 7, and the references therein). It should nevertheless be emphasized that we still have a ‘harmonious’ relationship between right and left introduction, albeit of a different kind than the ‘harmonious’ relationship between introductions and eliminations in natural deduction (see section 5.4).

References

- Aczel, P. (1977). An introduction to inductive definitions. In J. Barwise (Ed.), *Handbook of mathematical logic* (pp. 739–782). Amsterdam: North-Holland.
- Arndt, M. (2008). *Logical tomography: Exposing the structural constituents of logic*. University of Tübingen: Diss.
- Aronsson, M., Erksso, L.-H., Gåredal, A., Hallnäs, L., & Olin, P. (1990). The programming language GCLA: A definitional approach to logic programming. *New Generation Computing*, 4, 381–404.
- Atten, M. van. (2009). The development of intuitionistic logic. In E. N. Zalta (Ed.), *The Stanford encyclopedia of philosophy* (Summer 2009 ed.). <http://plato.stanford.edu/archives/sum2009/entries/intuitionistic-logic-development/>.
- Beall, J., & Restall, G. (2006). *Logical pluralism*. Oxford: Oxford University Press.
- Belnap, N. D. (1962). Tonk, plonk and plink. *Analysis*, 22, 130–134.
- Belnap, N. D. (1982). Display logic. *Journal of Philosophical Logic*, 11, 375–417.
- Brandom, R. B. (2000). *Articulating reasons: An introduction to inferentialism*. Cambridge Mass.: Harvard University Press.
- Brotherston, J., & Simpson, A. (2007). Complete sequent calculi for induction and infinite descent. In *Proceedings of the 22nd annual IEEE symposium on Logic in Computer Science (LICS)* (pp. 51–62). Los Alamitos: IEEE Press.
- Brünnler, K. (2010). *Nested sequents*. Universität Bern.
- Campos Sanz, W. d., & Piecha, T. (2009). Inversion by definitional reflection and the admissibility of logical rules. *Review of Symbolic Logic*, 2, 550–569.
- Carnap, R. (1947). *Meaning and necessity*. University of Chicago Press.
- Damáσιο, C. V., & Pereira, L. M. (1998). A survey of paraconsistent semantics for logic programs. In D. Gabbay & P. Smets (Eds.), *Handbook of defeasible reasoning and uncertainty management systems, vol. 2* (pp. 241–320). Dordrecht: Kluwer.
- de Groote, P. (Ed.). (1995). *The Curry-Howard isomorphism*. Academia.
- Denecker, M., Bruynooghe, M., & Marek, V. (2001). Logic programming revisited: Logic programs as inductive definitions. *ACM Transactions on Computational Logic*, 2, 623–654.
- Di Cosmo, R., & Miller, D. (2010). Linear logic. In E. N. Zalta (Ed.), *The Stanford encyclopedia of philosophy* (Fall 2010 ed.). <http://plato.stanford.edu/archives/fall2010/entries/logic-linear/>.
- Došen, K. (1980). *Logical constants: An essay in proof theory*. Oxford: D.Phil. Thesis.
- Došen, K. (1989). Logical constants as punctuation marks. *Notre Dame Journal of Formal Logic*, 30, 362–381.
- Došen, K. (2000). *Cut elimination in categories*. Berlin: Springer.

- Došen, K., & Schroeder-Heister, P. (1985). Conservativeness and uniqueness. *Theoria*, 51, 159–173.
- Došen, K., & Schroeder-Heister, P. (1988). Uniqueness, definability and interpolation. *Journal of Symbolic Logic*, 53, 554–570.
- Dummett, M. (1978). The justification of deduction (1973). In *Truth and other enigmas*. Duckworth.
- Dummett, M. (1991). *The logical basis of metaphysics*. London: Duckworth.
- Dyckhoff, R. (1988). Implementing a simple proof assistant. In *Workshop on programming for logic teaching (Leeds, 6-8 July 1987): Proceedings 23/1988* (pp. 49–59). University of Leeds: Centre for Theoretical Computer Science.
- Falkman, G. (2003). *Issues in structured knowledge representation: A definitional approach with application to case-based reasoning and medical informatics*. Chalmers University of Technology and Göteborg University: Ph.D. Thesis.
- Ferreira, F. (2006). Comments on predicative logic. *Journal of Philosophical Logic*, 35, 1–8.
- Francez, N. (2010). Proof-theoretic reconstruction of communication in natural language: Speaker-meaning vs. hearer-meaning.
- Francez, N., & Dyckhoff, R. (2008). A note on proof-theoretic validity (unpublished manuscript).
- Fredholm, D. (1995). Intensional aspects of function definitions. *Theoretical Computer Science*, 152, 1–66.
- Frege, G. (1892). Über Sinn und Bedeutung. *Zeitschrift für Philosophie und philosophische Kritik*, NF 100, 25–50.
- Gentzen, G. (1933). Über die Existenz unabhängiger Axiomensysteme zu unendlichen Satzsystemen. *Mathematische Annalen*, 107, 329–350 (English translation in: *The Collected Papers of Gerhard Gentzen* (ed. M. E. Szabo), Amsterdam: North Holland (1969), pp. 29–52).
- Gentzen, G. (1934/35). Untersuchungen über das logische Schließen. *Mathematische Zeitschrift*, 39, 176–210, 405–431 (English translation in: *The Collected Papers of Gerhard Gentzen* (ed. M. E. Szabo), Amsterdam: North Holland (1969), pp. 68–131).
- Girard, J.-Y. (1971). Une extension de l'interprétation de Gödel à l'analyse, et son application à l'élimination des coupures dans l'analyse et la théorie des types. In J. E. Fenstad (Ed.), *Proceedings of the second Scandinavian logic symposium* (pp. 63–92). Amsterdam: North-Holland.
- Girard, J.-Y. (1987). Linear logic. *Theoretical Computer Science*, 1–102.
- Gödel, K. (1933). Eine Interpretation des intuitionistischen Aussagenkalküls. *Ergebnisse eines mathematischen Kolloquiums*, 4, 39–40.
- Gödel, K. (1958). Über eine bisher noch nicht benützte Erweiterung des finiten Stand-

- punktes. *Dialectica*, 12, 280–287.
- Goodman, N. D. (1970). A theory of constructions equivalent to arithmetic. In J. Myhill, A. Kino, & R. E. Vesley (Eds.), *Intuitionism and proof theory* (pp. 101–120). Amsterdam: North-Holland.
- Hacking, I. (1977). Do-it-yourself semantics for classical sequent calculi, including ramified type theory. In R. E. Butts & J. Hintikka (Eds.), *Logic, foundations of mathematics and computability theory* (pp. 371–390). Dordrecht: Reidel.
- Hacking, I. (1979). What is logic? *Journal of Philosophy*, 76, 285–319.
- Hallnäs, L. (1991). Partial inductive definitions. *Theoretical Computer Science*, 87, 115–142.
- Hallnäs, L. (2006). On the proof-theoretic foundation of general definition theory. *Synthese*, 148, 589–602.
- Hallnäs, L., & Schroeder-Heister, P. (1990/91). A proof-theoretic approach to logic programming: I. Clauses as rules. II. Programs as definitions. *Journal of Logic and Computation*, 1, 261–283, 635–660.
- Harper, R., Honsell, F., & Plotkin, G. (1987). A framework for defining logics. *Journal of the Association for Computing Machinery*, 40, 194–204.
- Humberstone, L., & Makinson, D. (2011). Intuitionistic logic and elementary rules. *Mind*.
- Jäger, G., & Stärk, R. F. (1998). A proof-theoretic framework for logic programming. In S. R. Buss (Ed.), *Handbook of proof theory* (pp. 639–682). Amsterdam: Elsevier.
- Joachimski, F., & Matthes, R. (2003). Short proofs of normalization for the simply-typed λ -calculus, permutative conversions and Gödel's T. *Archive for Mathematical Logic*, 42, 59–87.
- Kahle, R., & Schroeder-Heister, P. (Eds.). (2006). *Proof-theoretic semantics*. Special issue of *Synthese* (Vol. 148).
- Kleene, S. (1945). On the interpretation of intuitionistic number theory. *Journal of Symbolic Logic*, 10, 109–124.
- Kneale, W. (1956). The province of logic. In H. D. Lewis (Ed.), *Contemporary British philosophy* (pp. 237–261). London: Allen and Unwin.
- Koslow, A. (1992). *A structuralist theory of logic*. Cambridge: Cambridge University Press.
- Kreisel, G. (1962). Foundations of intuitionistic logic. In E. Nagel, P. Suppes, & A. Tarski (Eds.), *Logic, methodology and philosophy of science I* (pp. 198–210). Stanford: Stanford University Press.
- Kreisel, G. (1971). A survey of proof theory II. In J. E. Renstad (Ed.), *Proceedings of the second Scandinavian logic symposium* (pp. 109–170). Amsterdam: North-Holland.
- Kremer, P. (2009). The revision theory of truth. In E. N. Zalta (Ed.), *The Stanford*

- encyclopedia of philosophy* (Spring 2009 ed.). <http://plato.stanford.edu/archives/spr2009/entries/truth-revision/>.
- Kreuger, P. (1994). Axioms in definitional calculi. In R. Dyckhoff (Ed.), *Extensions of logic programming. 4th international workshop, ELP'93 (St. Andrews, U.K., March/April 1993). Proceedings (Lecture Notes in Computer Science, vol. 798)* (pp. 196–205). Berlin: Springer.
- Kutschera, F. v. (1968). Die Vollständigkeit des Operatorensystems $\{\neg, \wedge, \vee, \supset\}$ für die intuitionistische Aussagenlogik im Rahmen der Gentzensemantik. *Archiv für mathematische Logik und Grundlagenforschung*, 11, 3–16.
- Kutschera, F. v. (1969). Ein verallgemeinerter Widerlegungs begriff für Gentzenkalküle. *Archiv für mathematische Logik und Grundlagenforschung*, 12, 104–118.
- Lambek, J., & Scott, P. (1986). *Introduction to higher order categorical logic*. Cambridge: Cambridge University Press.
- López-Escobar, E. G. K. (1999). Standardizing the N systems of Gentzen. In X. Caicedo & C. H. Montenegro (Eds.), *Models, algebras and proofs (Lecture Notes in Pure and Applied Mathematics, Vol. 203)* (pp. 411–434). New York: Marcel Dekker.
- Lorenz, K. (2001). Basic objectives of dialogue logic in historical perspective. *Synthese*, 127, 255–263.
- Lorenzen, P. (1950). Konstruktive Begründung der Mathematik. *Mathematische Zeitschrift*, 53, 162–202.
- Lorenzen, P. (1955). *Einführung in die operative Logik und Mathematik*. Berlin: Springer (2nd edition 1969).
- Lorenzen, P. (1960). Logik und Agon. In *Atti del XII Congresso Internazionale di Filosofia (Venezia, 1958)* (pp. 187–194). Firenze: Sansoni.
- MacFarlane, J. (2009). Logical constants. In E. N. Zalta (Ed.), *The Stanford encyclopedia of philosophy* (Fall 2009 ed.). <http://plato.stanford.edu/archives/fall2009/entries/logical-constants/>.
- Martin-Löf. (2009). Evaluation of open expressions. In *Symposium on programming, types, and languages: Dedicated to Bengt Nordström on occasion of his 60th birthday (Chalmerska Huset, Gothenburg, 19 March 2009)*. Göteborg.
- Martin-Löf, P. (1971). Hauptsatz for the intuitionistic theory of iterated inductive definitions. In J. E. Fenstad (Ed.), *Proceedings of the Second Scandinavian Logic Symposium* (pp. 179–216). Amsterdam: North-Holland.
- Martin-Löf, P. (1984). *Intuitionistic type theory*. Napoli: Bibliopolis.
- Martin-Löf, P. (1995). Verificationism then and now. In W. DePauli-Schimanovich, E. Köhler, & F. Stadler (Eds.), *The foundational debate: Complexity and constructivity in mathematics and physics* (pp. 187–196). Dordrecht: Kluwer.
- Martin-Löf, P. (1998). Truth and knowability: On the principles C and K of Michael

- Dummett. In H. G. Dales & G. Oliveri (Eds.), *Truth in mathematics* (pp. 105–114). Oxford: Clarendon Press.
- Milne, P. (2010). Subformula and separation properties in natural deduction via small kripke models. *Review of Symbolic Logic*, 3, 175–227.
- Montague, R. (1970). English as a formal language. In P. Visentini et al. (Ed.), *Linguaggi nella Società e nella Tecnica (repr. in: R.H. Thomason (ed.), Formal Philosophy: Selected Papers of Richard Montague, New Haven, Yale University Press 1974, 188–221)*. Milano.
- Moschovakis, J. (2010). Intuitionistic logic. In E. N. Zalta (Ed.), *The Stanford encyclopedia of philosophy* (Summer 2010 ed.). <http://plato.stanford.edu/archives/sum2010/entries/logic-intuitionistic/>.
- Negri, S., & von Plato, J. (2001). *Structural proof theory*. Cambridge University Press.
- Nelson, D. (1949). Constructible falsity. *Journal of Symbolic Logic*, 14, 16–26.
- Nordström, B., Petersson, K., & Smith, J. (1990). *Programming in Martin-Löf's type theory: An introduction*. Oxford: Clarendon Press.
- Odintsov, S. P. (2008). *Constructive negations and paraconsistency*. Berlin: Springer.
- O'Hearn, P., & Pym, D. (1999). The logic of bunched implications. *Bulletin of Symbolic Logic*, 5, 215–244.
- Plato, J. v. (2001). Natural deduction with general elimination rules. *Archive for Mathematical Logic*, 40, 541–567.
- Poggiolesi, F. (2010). *Gentzen calculi for modal propositional logic*. Berlin: Springer.
- Prawitz, D. (1965). *Natural deduction: A proof-theoretical study*. Stockholm: Almqvist & Wiksell (Reprinted Mineola NY: Dover Publ., 2006).
- Prawitz, D. (1971). Ideas and results in proof theory. In J. E. Fenstad (Ed.), *Proceedings of the second Scandinavian logic symposium (Oslo 1970)* (pp. 235–308). Amsterdam: North-Holland.
- Prawitz, D. (1972). The philosophical position of proof theory. In R. E. Olson & A. M. Paul (Eds.), *Contemporary philosophy in Scandinavia* (pp. 123–134). Baltimore, London: John Hopkins Press.
- Prawitz, D. (1973). Towards a foundation of a general proof theory. In P. Suppes et al. (Ed.), *Logic, methodology and philosophy of science IV* (p. 225–250). North-Holland.
- Prawitz, D. (1974). On the idea of a general proof theory. *Synthese*, 27, 63–77.
- Prawitz, D. (1979). Proofs and the meaning and completeness of the logical constants. In J. Hintikka, I. Niiniluoto, & E. Saarinen (Eds.), *Essays on mathematical and philosophical logic: Proceedings of the fourth Scandinavian logic symposium and the first Soviet-Finnish logic conference, Jyväskylä, Finland, June 29 – July 6, 1976* (pp. 25–40 (revised German translation ‘Beweise und die Bedeutung und Vollständigkeit der logischen Konstanten, *Conceptus*, 16, 1982, 31–44)). Dor-

- drecht: Kluwer.
- Prawitz, D. (1985). Remarks on some approaches to the concept of logical consequence. *Synthese*, 62, 152–171.
- Prawitz, D. (2006). Meaning approached via proofs. *Synthese*, 148, 507–524.
- Prawitz, D. (2007). Pragmatist and verificationist theories of meaning. In R. E. Auxier & L. E. Hahn (Eds.), *The philosophy of Michael Dummett* (pp. 455–481). Open Court.
- Prawitz, D. (2009). Truth and proof: Ontological or epistemic concepts. In *Philosophy and foundations of mathematics: Epistemological and ontological aspects*. Uppsala.
- Prawitz, D. (2010). An approach to general proof theory and a conjecture of a kind of completeness of intuitionistic logic revisited. In E. H. Haeusler, L. C. Pereira, & V. de Paiva (Eds.), *Advances in natural deduction*.
- Prior, A. N. (1960). The runabout inference-ticket. *Analysis*, 21, 38–39.
- Restall, G. (2009). Substructural logics. In E. N. Zalta (Ed.), *The Stanford encyclopedia of philosophy* (Summer 2009 ed.). <http://plato.stanford.edu/archives/sum2009/entries/logic-substructural/>.
- Sambin, G., Battilotti, G., & Faggian, C. (2000). Basic logic: Reflection, symmetry, visibility. *Journal of Symbolic Logic*, 65, 979–1013.
- Sandqvist, T. (2009). Classical logic without bivalence. *Analysis*, 69, 211–218.
- Schroeder-Heister, P. (1981). *Untersuchungen zur regellogischen Deutung von Aussagenverknüpfungen*. Universität Bonn: Diss.
- Schroeder-Heister, P. (1984a). Generalized rules for quantifiers and the completeness of the intuitionistic operators \wedge , \vee , \rightarrow , \perp , \forall , \exists . In M. M. Richter, E. Börger, W. Oberschelp, B. Schinzel, & W. Thomas (Eds.), *Computation and proof theory. proceedings of the logic colloquium held in Aachen, July 18-23, 1983, Part II. (Lecture Notes in Mathematics Vol. 1104)* (pp. 399–426). Berlin: Springer.
- Schroeder-Heister, P. (1984b). A natural extension of natural deduction. *Journal of Symbolic Logic*, 49, 1284–1300.
- Schroeder-Heister, P. (1984c). Popper’s theory of deductive inference and the concept of a logical constant. *History and Philosophy of Logic*, 5, 79–110.
- Schroeder-Heister, P. (1987). *Structural frameworks with higher-level rules*. Universität Konstanz: Habilitationsschrift.
- Schroeder-Heister, P. (1991a). Hypothetical reasoning and definitional reflection in logic programming. In P. Schroeder-Heister (Ed.), *Extensions of logic programming. International workshop, Tübingen, December 1989, Proceedings* (pp. 327–340). Berlin: Springer.
- Schroeder-Heister, P. (1991b). Structural frameworks, substructural logics, and the role of elimination inferences. In G. Huet & G. Plotkin (Eds.), *Logical frameworks*

- (pp. 385–403). Cambridge University Press.
- Schroeder-Heister, P. (1991c). Uniform proof-theoretic semantics for logical constants (Abstract). *Journal of Symbolic Logic*, 56, 1142.
- Schroeder-Heister, P. (1992). Cut elimination in logics with definitional reflection. In D. Pearce & H. Wansing (Eds.), *Nonclassical logics and information processing: International workshop, Berlin, November 1990, Proceedings (Lecture Notes in Computer Science Vol. 619)* (pp. 146–171). Berlin: Springer.
- Schroeder-Heister, P. (1993). Rules of definitional reflection. In *Proceedings of the 8th annual IEEE symposium on Logic in Computer Science (Montreal 1993)* (pp. 222–232). Los Alamitos: IEEE Press.
- Schroeder-Heister, P. (1994). Definitional reflection and the completion. In R. Dyckhoff (Ed.), *Extensions of logic programming. Proceedings of the 4th international workshop, ELP '93 (St. Andrews, March/April 1993, Lecture Notes in Computer Science, Vol. 798)* (pp. 333–347). Berlin: Springer.
- Schroeder-Heister, P. (2002). Resolution and the origins of structural reasoning: Early proof-theoretic ideas of Hertz and Gentzen. *Bulletin of Symbolic Logic*, 8, 246–265.
- Schroeder-Heister, P. (2004). On the notion of *assumption* in logical systems. In R. Bluhm & C. Nimtz (Eds.), *Selected papers contributed to the sections of GAP5, Fifth International Congress of the Society for Analytical Philosophy, Bielefeld, 22-26 September 2003* (pp. 27–48). Paderborn: mentis (Online publication: www.gap5.de/proceedings).
- Schroeder-Heister, P. (2005). Popper's structuralist theory of logic. In I. Jarvie, K. Milford, & D. Miller (Eds.), *Karl Popper: A centenary assessment. Vol. III: Science* (pp. 17–36). Aldershot: Ashgate.
- Schroeder-Heister, P. (2006). Validity concepts in proof-theoretic semantics. *Synthese*, 148, 525–571.
- Schroeder-Heister, P. (2007). Generalized definitional reflection and the inversion principle. *Logica Universalis*, 1, 355–376.
- Schroeder-Heister, P. (2008a). Lorenzen's operative justification of intuitionistic logic. In M. van Atten, P. Boldini, M. Bourdeau, & G. Heinzmann (Eds.), *One hundred years of intuitionism (1907-2007): The Cerisy conference* (pp. 214–240 [References for whole volume: 391–416]). Basel: Birkhäuser.
- Schroeder-Heister, P. (2008b). Proof-theoretic versus model-theoretic consequence. In M. Peliš (Ed.), *The Logica Yearbook 2007* (pp. 187–200). Prague: Filosofia.
- Schroeder-Heister, P. (2009). Sequent calculi and bidirectional natural deduction: On the proper basis of proof-theoretic semantics. In M. Peliš (Ed.), *The Logica Yearbook 2008*. London: College Publications.
- Schroeder-Heister, P. (2010a). The categorical and the hypothetical: A critique of

- some fundamental assumptions of standard semantics. *Synthese*.
- Schroeder-Heister, P. (2010b). Definitional reasoning in proof-theoretic semantics and the square of opposition. In J.-Y. Béziau (Ed.), *Proceedings of the first international congress on the square of opposition*. Frankfurt: Peter Lang.
- Schroeder-Heister, P. (2010c). Generalized elimination inferences, higher-level rules, and the implications-as-rules interpretation of the sequent calculus. In E. H. Haeusler, L. C. Pereira, & V. de Paiva (Eds.), *Advances in natural deduction*.
- Schroeder-Heister, P. (2011a). Definitional reflection and Basic Logic. *Annals of Pure and Applied Logic (Special issue, Festschrift 60th Birthday Giovanni Sambin)* submitted.
- Schroeder-Heister, P. (2011b). Implications-as-rules vs. implications-as-links: An alternative implication-left schema for the sequent calculus. *Journal of Philosophical Logic*, 40, 95–101.
- Schroeder-Heister, P. (2011c). Schluß und Umkehrschluß: Ein Beitrag zur Definitionstheorie. In C. F. Gethmann (Ed.), *Lebenswelt und Wissenschaft. Kolloquienbeiträge und öffentliche Vorträge des XXI. Deutschen Kongresses für Philosophie (Essen 15. 19.09. 2008)*. Hamburg: Meiner.
- Shoemith, D. J., & Smiley, T. J. (1978). *Multiple-conclusion logic*. Cambridge: Cambridge University Press.
- Sommaruga, G. (2000). *History and philosophy of constructive type theory*. Dordrecht: Kluwer.
- Sørensen, M. H. B., & Urzyczyn, P. (2006). *Lectures on the Curry-Howard isomorphism*. Amsterdam: Elsevier.
- Stålmarmark, G. (1991). Normalization theorems for full first order classical natural deduction. *The Journal of Symbolic Logic*, 56, 129–149.
- Sundholm, G. (2000). Proofs as acts and proofs as objects: Some questions for Dag Prawitz. *Theoria*, 64, 187–216.
- Tait, W. W. (1967). Intensional interpretations of functionals of finite type I. *Journal of Symbolic Logic*, 32, 198–212.
- Tarski, A. (1935). Der Wahrheitsbegriff in den formalisierten Sprachen (translated from the Polish original of 1933, with a postscript). *Studia Philosophica*, 1, 261–405.
- Tennant, N. (1978). *Natural logic*. Edinburgh: Edinburgh University Press.
- Tennant, N. (1987). *Anti-realism and logic: Truth as eternal*. Oxford: Clarendon Press.
- Tennant, N. (1992). *Autologic*. Edinburgh: Edinburgh University Press.
- Tennant, N. (1997). *The taming of the true*. Oxford: Clarendon Press.
- Tennant, N. (1999). Negation, absurdity and contrariety. In D. Gabbay & H. Wansing

- (Eds.), *Negation* (pp. 199–222). Dordrecht: Kluwer.
- Tennant, N. (2002). Ultimate normal forms for parallelized natural deductions. *Logic Journal of the IGPL*, 10, 299–337.
- Torgersson, O. (2000). *On GCLA, Gisela, and MedView: Studies in declarative programming with application of clinical medicine*. Chalmers University of Technology and Göteborg University: Ph.D. Thesis.
- Tranchini, L. (2011a). Natural deduction for dual intuitionistic logic. *Studia Logica*.
- Tranchini, L. (2011b). *Proof and truth: An anti-realist perspective*. Milano.
- Troelstra, A. S., & Schwichtenberg, H. (1996). *Basic proof theory*. Cambridge University Press.
- Troelstra, A. S., & van Dalen, D. (1988). *Constructivism in mathematics: An introduction*. Amsterdam: North-Holland.
- Usberti, G. (2006). Towards a semantics based on the notion of justification. *Synthese*, 148, 675–699.
- Wansing, H. (1993). *The logic of information structures*. Berlin: Springer Springer Lecture Notes in Artificial Intelligence, Vol. 681.
- Więckowski, B. (2008). Predication in fiction. In M. Peliš (Ed.), *The Logica Yearbook 2007* (pp. 267–285). Prague: Filosofia.
- Więckowski, B. (2011). Rules for subatomic derivation. *Review of Symbolic Logic*, 4.
- Zeilberger, N. (2008). On the unity of duality. *Annals of Pure and Applied Logic*, 153, 66–96.
- Zimmermann, E. (2010). [combinatorial proof of strong normalization, exact title tba]. In E. H. Haeusler, L. C. Pereira, & V. de Paiva (Eds.), *Advances in natural deduction*.