Morphing Rules of Evaluation into Rules of Deduction: Preserving Relevance and Epistemic Gain

by

Neil Tennant*

Department of Philosophy
The Ohio State University

August 12, 2023

Abstract

This study seeks to reveal the proper source of the (*correct*) rules of natural deduction (and their associated rules of the sequent calculus). Perhaps surprisingly, this source consists of just the familiar *truth tables* (deriving from Frege). These tables can be construed inferentially. The primitive steps of value-computation correspond to primitive steps of 'inference'. We shall call them, however, primitive steps (or rules) of *evaluation*. These can be steps of *verification* or of *falsification*. The rules of evaluation constitute the inductive clauses in a metalinguistic co-inductive definition of model-relative verifications and falsifications.

We then show how the rules of evaluation can be 'morphed' into rules of natural deduction. Rules of verification thereby become introduction rules, and rules of falsification become elimination rules. The morphing produces model-*invariant* rules in the simplest way possible. It preserves, for natural deduction, the feature of *relevance* that is involved in truth-tabular computation. This makes for a system of natural deduction (and a directly corresponding sequent calculus) that is relevant.

*To appear in the forthcoming volume 'New Directions in Relevant Logic', edited by Igor Sedlár, Shawn Standefer, and Andrew Tedder, in Springer's *Trends in Logic* series.

1 Introduction

This study is inferentialist in spirit. We aim to show how to pass from the rules of evaluation (verification and falsification) that are enshrined in the truth tables, to rules of natural deduction (introduction and elimination). Rules of evaluation are *model relative*. Rules of natural deduction, however, are *model invariant*. We show how to morph the former rules into the latter rules. Our method of morphing preserves the connections of relevance that are intrinsic to evaluations (which are computations, hence relevant); and produces rules of inference so formulated as to ensure the relevance of premises to conclusions of the proofs that they generate. The deductive rules that result from this investigation are those of Core Logic $\mathbb{C}$. We shall also have occasion to invoke its classicized extension, $\mathbb{C}^+$.

2 Notation

Λ	A set of literals (encoding a model)
φ, ψ, θ	Sentences
Δ, Γ	Sets of sentences
$\Delta : \varphi$	A sequent (in single-conclusion logic)

For any logical operator @:

(@-$\mathcal{V}$)	the Verification rule for @
(@-$\mathcal{F}$)	the Falsification rule for @
(@-I)	the Introduction rule for @ in Natural Deduction
(@-E)	the Elimination rule for @ in Natural Deduction
(@-R)	the Right rule for @ in Sequent Calculus
(@-L)	the Left rule for @ in Sequent Calculus

Evaluation

Λ	an interpretation (truth-value assignment, or model)
$\mathcal{V}(V, \varphi, \Lambda)$	V is a verification of φ relative to Λ
$\mathcal{F}(F, \varphi, \Lambda)$	F is a falsification of φ relative to Λ

$\Lambda \Vdash \varphi$	$\exists V\ \mathcal{V}(V, \varphi, \Lambda)$	'Λ makes φ true'
$\Lambda, \varphi \Vdash \bot$	$\exists F\ \mathcal{F}(F, \varphi, \Lambda)$	'Λ makes φ false'

Deduction

$\mathcal{P}_{\mathcal{S}}(\Pi, \varphi, \Delta)$ Π is an $\mathcal{S}$-proof of φ, with the set Δ of premises
($\mathcal{S}$ is the system of natural deduction, or the sequent calculus.)

$\Delta \vdash_{\mathcal{S}} \varphi$	$\exists \Gamma \subseteq \Delta\ \exists \Pi\ \mathcal{P}_{\mathcal{S}}(\Pi, \varphi, \Gamma)$	'Δ $\mathcal{S}$-proves φ', or 'φ is $\mathcal{S}$-deducible from Δ'

Note that $\mathcal{P}_{\mathcal{S}}(\Pi, \varphi, \Delta)$ implies that Δ is the exact set of premises (i.e., undischarged assumptions) of the proof Π. This set is finite, because proofs are finite.

3 The method of framing rules of verification and of falsification

The rules of verification and of falsification (collectively: 'evaluation rules') are 'model-relative'. They were put forward in Tennant [2010] and Tennant [2018]. They were also dealt with in Chapter 3 of Tennant [2017].

The value T (for true) and the value F (for false) are conceptually coequal. Each is just as important as the other, as far as the familiar truth tables are concerned. We remind the reader of a point made in Tennant [2017], at p. 53:

> The truth tables should really be called *truth-value* tables, because they involve two truth values, T and F, with F figuring just as importantly within them as T. We need to bear that overriding consideration in mind, and not let it be eclipsed by our preference, when forming beliefs or making assertions, for true propositions over false ones.

The rules of verification and of falsification, likewise, are coequal. Verifications and falsifications are coinductively defined. The rules of verification and of falsification for the connectives can be read off from their familiar truth tables. Reciprocally, the familiar truth tables can be constructed directly from the rules of verification and of falsification themselves. This means that we can be inferentialists *au fond*.

If, however, one wishes to take the truth tables as one's starting point, one can advance to the rules of verification and of falsification in the way we shall now describe. The reader is invited to read each row within each of the following truth tables from left to right. We have supplied, in the final column, the valid sequents expressing the value-determining contribution of the pertinent row. Assignment of the value T to an immediate constituent is registered by having that constituent stand as a premise of the sequent, on the left. Assignment of the value F to an immediate constituent is registered by having the negation of that constituent stand as a premise of the sequent, on the left. When the compound sentence receives the value T, this is registered by having the compound sentence feature as the conclusion of the sequent, whose premises represent the assignments of values to the immediate constituents. When the compound sentence receives the value F, this is registered by having $\bot$ on the right as the conclusion of the sequent in question, and having the compound sentence featuring on the left as a premise.

φ	$\neg\varphi$	
T	F	$\neg\varphi, \varphi : \bot$
F	T	$\neg\varphi : \neg\varphi$ (trivial)

φ	ψ	$\varphi \wedge \psi$	
T	T	T	$\varphi, \psi : \varphi \wedge \psi$
T	F	F	$\varphi \wedge \psi, \neg\psi : \bot)$
F	T	F	$\varphi \wedge \psi, \neg\varphi : \bot)$
F	F	F	overkill

φ	ψ	$\varphi \vee \psi$	
T	T	T	overkill
T	F	T	$\varphi : \varphi \vee \psi$
F	T	T	$\psi : \varphi \vee \psi$
F	F	F	$\varphi \vee \psi, \neg\varphi, \neg\psi : \bot$

φ	ψ	$\varphi \rightarrow \psi$	
T	T	T	$\psi : \varphi \rightarrow \psi$
T	F	F	$\varphi \rightarrow \psi, \varphi, \neg\psi : \bot$
F	T	T	overkill
F	F	T	$\neg\varphi : \varphi \rightarrow \psi$

The rules of verification and of falsification for the quantifiers treat quantified sentences as generalized conjunctions or disjunctions of all instances of the predicate that is quantified. Such generalized conjunctions or disjunctions can of course be infinitary when the domain of interpretation is infinite. That also leads to evaluations being infinitary ('infinitely wide' in places)—in particular, when verifying universal generalizations, or falsifying existential ons. It is for this reason that we have to regard evaluations as very different indeed from proofs. For the latter are finite and effectively checkable for correctness; whereas evaluations need not be. Nevertheless, the seeds of proof—namely, rules of inference—are to be found in the rules of evaluation themselves. The present study aims to explain how this is so.

1. Our rules of evaluation determine whether a sentence is **verified** or **falsified** in a given interpretation.

2. An *interpretation*, or *model* (in the first-order case) consists of a *domain of individuals*, and a specification of *extensions of predicates* within that domain. In the propositional case, an interpretation is an assignment of truth values to the atomic sentences.

3. A **verification** of a sentence φ (relative to an interpretation M) establishes φ as its *conclusion*:

$$\begin{array}{c}\vdots\\ \varphi\end{array}$$

4. A **falsification** of a sentence φ (relative to an interpretation M) establishes absurdity ($\perp$) from φ as its *major premise*:

$$\begin{array}{c}\varphi\\ \vdots\\ \perp\end{array}$$

Note that major premises for steps of falsification always 'stand proud', with no 'evaluation work' above them. This important feature of major premises will be preserved when we make (in §6) the promised transition to Elimination rules in natural deduction.

The simplest verifications (relative to a first-order interpretation M) are of the form

$$\frac{}{P(\alpha_1, \ldots, \alpha_n)}\,^M$$

where in M the predicate P holds of the individuals $\alpha_1, \ldots, \alpha_n$.

This is a verification of $P(\alpha_1, \ldots, \alpha_n)$ relative to M.

The simplest falsifications (relative to an interpretation M) are of the form

$$\frac{P(\alpha_1, \ldots, \alpha_n)}{\perp}\,_M$$

where in M the predicate P fails to hold of the individuals $\alpha_1, \ldots, \alpha_n$.

This is a falsification of $P(\alpha_1, \ldots, \alpha_n)$ relative to M.

Any interpretation M is *atomically complete*: every atomic sentence features either in a positive M-factoid, or in a negative M-factoid. Any interpretation M is also *coherent*: no atomic sentence features both in a positive M-factoid, and in a negative M-factoid.

More complex verifications and falsifications (relative to M) are built up in accordance with the rules (@-$\mathcal{V}$) and (@-$\mathcal{F}$) for logical operators @, which will be stated presently.

In the propositional case the simplest verifications and falsifications relative to a truth-value assignment τ take the respective forms

$$\frac{}{A}\,^\tau \qquad \text{and} \qquad \frac{A}{\perp}\,_\tau \ ,$$

for atomic sentences A.

Note that we are dealing here, in the first-order case, with *saturated* formulae. These are created by taking a formula $\varphi(x_1, \ldots, x_n)$ (with the indicated variables free) and replacing each free occurrence of a variable x_i $(1 \leq i \leq n)$ with an individual α_i:

$$\varphi^{x_1}_{\alpha_1} \cdots {}^{x_n}_{\alpha_n} \;=\; \varphi(\alpha_1, \ldots, \alpha_n).$$

Saturated formulae do not have any free variables. That is why they are able to be true, or false, in an interpretation. They can be thought of as *sentences*, liable to come out true, or come out false, in any model that interprets their extra-logical vocabulary.

A *literal* is an atomic sentence, or the negation of an atomic sentence. Thus atomic saturated formulae and their negations are literals. Construed inferentially, literals are our simplest possible evaluations. This constitutes the *basis clause* in our co-inductive definition of (the tree-like arrays that we are calling) verifications and falsifications.

In the propositional case, the simplest possible verifications and falsifications are, respectively,

$$\frac{}{A}\,\tau \qquad \text{and} \qquad \frac{A}{\bot}\,\tau\ .$$

In the first-order case, they are

$$\frac{}{P(\alpha_1,\ldots,\alpha_n)}\,M \qquad \text{and} \qquad \frac{P(\alpha_1,\ldots,\alpha_n)}{\bot}\,M\ .$$

So: the simplest possible verifications and falsifications express, respectively, *positive* or *negative* factoids.

We shall use Λ as a placeholder for sets of literals of the interpretation in question. Every such Λ is a coherent set of factoids. We choose purple as the coloration here to remind the reader that these sets Λ can in general contain both positive and negative factoids.

Such sets Λ feature as the sets of 'undischarged side assumptions' in the following graphic statements of our rules of evaluation. These rules are of the respective forms @-$\mathcal{V}$ and @-$\mathcal{F}$ for each logical operator @.

Each such rule can be construed as an inductive clause in our co-inductive definition of the notions

$$\mathcal{V}(\Pi,\varphi,M,D) \text{ and } \mathcal{F}(\Pi,\varphi,M,D).$$

These metalinguistic predications can be read, respectively, as

'Π is a verification of φ relative to model M with domain D'

and

'Π is a falsification of φ relative to model M with domain D'.

3.1 Verification and Falsification rules for $\neg$

$$(\neg\text{-}\mathcal{V}) \qquad \begin{array}{c} \Box\overline{}^{(i)} \\ \underbrace{\varphi\ ,\ \Lambda} \\ \vdots \\ \dfrac{\bot}{\neg\varphi}\,(i) \end{array} \qquad \text{Note the box } \Box\,.$$

To verify a negation $\neg\varphi$ one must falsify φ. That is, one must actually *use* φ as an 'evaluative assumption' and show it to be false (*modulo* Λ). This requirement of 'non-vacuous discharge' is what is represented by the box that annotates the discharge stroke. The same remark applies to subsequent boxes in other evaluation rules.

($\neg$-$\mathcal{F}$)

$$\frac{\neg\varphi \qquad \begin{array}{c}\Lambda\\ \vdots\\ \varphi\end{array}}{\bot}$$

To falsify a negation $\neg\varphi$ one must verify φ.

Note the classicism that has immediately crept in by virtue of this dualizing.

3.2 Verification and Falsification rules for $\wedge$

($\wedge$-$\mathcal{V}$)

$$\frac{\begin{array}{c}\Lambda_1\\ \vdots\\ \varphi\end{array} \qquad \begin{array}{c}\Lambda_2\\ \vdots\\ \psi\end{array}}{\varphi\wedge\psi}$$

To verify a conjunction, one must verify both conjuncts.

($\wedge$-$\mathcal{F}$)

$$\frac{\varphi\wedge\psi \qquad \begin{array}{c}\overset{\Box\text{—}(i)}{\underbrace{\varphi\ ,\ \Lambda}}\\ \vdots\\ \bot\end{array}}{\bot}{\scriptstyle (i)} \qquad \frac{\varphi\wedge\psi \qquad \begin{array}{c}\overset{\Box\text{—}(i)}{\underbrace{\psi\ ,\ \Lambda}}\\ \vdots\\ \bot\end{array}}{\bot}{\scriptstyle (i)}$$

Note the boxes.

To falsify a conjunction, one has to falsify one of the conjuncts.

3.3 Verification and Falsification rules for $\vee$

($\vee$-$\mathcal{V}$)

$$\frac{\begin{array}{c}\Lambda\\ \vdots\\ \varphi\end{array}}{\varphi\vee\psi} \qquad \frac{\begin{array}{c}\Lambda\\ \vdots\\ \psi\end{array}}{\varphi\vee\psi}$$

To verify a disjunction, one has to verify one of the disjuncts.

$$(\vee\text{-}\mathcal{F}) \qquad \dfrac{\varphi \vee \psi \qquad \begin{array}{c} \square\!\!-\!(i) \\ \underbrace{\varphi\ ,\ \Lambda_1} \\ \vdots \\ \bot \end{array} \qquad \begin{array}{c} \square\!\!-\!(i) \\ \underbrace{\psi\ ,\ \Lambda_2} \\ \vdots \\ \bot \end{array}}{\bot}\,(i) \qquad \text{Note the boxes.}$$

To falsify a disjunction, one must falsify both disjuncts.

3.4 Verification and Falsification rules for $\rightarrow$

$$(\rightarrow\text{-}\mathcal{V}) \qquad \dfrac{\begin{array}{c} \square\!\!-\!(i) \\ \underbrace{\varphi\ ,\ \Lambda} \\ \vdots \\ \bot \end{array}}{\varphi \rightarrow \psi}\,(i) \qquad \dfrac{\begin{array}{c} \Lambda \\ \vdots \\ \psi \end{array}}{\varphi \rightarrow \psi} \qquad \text{Note the box.}$$

To verify a conditional, one must either falsify its antecedent or verify its consequent.

$$(\rightarrow\text{-}\mathcal{F}) \qquad \dfrac{\varphi \rightarrow \psi \qquad \begin{array}{c} \Lambda_1 \\ \vdots \\ \varphi \end{array} \qquad \begin{array}{c} \square\!\!-\!(i) \\ \underbrace{\psi\ ,\ \Lambda_2} \\ \vdots \\ \bot \end{array}}{\bot}\,(i) \qquad \text{Note the box.}$$

To falsify a conditional, one must both verify its antecedent and falsify its consequent.

3.5 Verification and Falsification rules for $\exists$

$$(\exists\text{-}\mathcal{V}) \qquad \dfrac{\begin{array}{c} \Lambda \\ \vdots \\ \varphi^x_{\boldsymbol{\alpha}} \end{array}}{\exists x \varphi} \qquad \text{where } \boldsymbol{\alpha} \text{ is any individual in the domain}$$

To verify an existential, one must verify some instance of it.

$$(\exists\text{-}\mathcal{F}) \qquad \dfrac{\exists x\varphi \qquad \begin{array}{c}\underbrace{\overset{\Box\,(i)}{\varphi^x_{\boldsymbol{\alpha_1}}}\ ,\ \Lambda_1}\\ \vdots \\ \bot\end{array} \ \cdots \ \begin{array}{c}\underbrace{\overset{\Box\,(i)}{\varphi^x_{\boldsymbol{\alpha_n}}}\ ,\ \Lambda_n}\\ \vdots \\ \bot\end{array} \ \cdots}{\bot}\,(i) \qquad \text{Note the boxes.}$$

where $\boldsymbol{\alpha_1, \ldots, \alpha_n, \ldots}$ are all the individuals in the domain

To falsify an existential, one must falsify every instance of it.

3.6 Verification and Falsification rules for $\forall$

$$(\forall\text{-}\mathcal{V}) \qquad \dfrac{\begin{array}{c}\Lambda_1\\ \vdots \\ \psi(\boldsymbol{\alpha_1})\end{array} \quad \cdots \quad \begin{array}{c}\Lambda_n\\ \vdots \\ \psi(\boldsymbol{\alpha_n})\end{array} \quad \cdots}{\forall x\psi(x)}$$

where $\boldsymbol{\alpha_1, \ldots, \alpha_n, \ldots}$ are all the individuals in the domain

To verify a universal, one must verify every instance of it.

$$(\forall\text{-}\mathcal{F}) \qquad \dfrac{\forall x\psi(x) \qquad \begin{array}{c}\underbrace{\overset{\Box\,(i)}{\psi(\boldsymbol{\alpha})}\ ,\ \Lambda}\\ \vdots \\ \bot\end{array}}{\bot}\,(i) \qquad \text{Note the box.}$$

where $\boldsymbol{\alpha}$ is any individual in the domain

To falsify a universal, one must falsify some instance of it.

4 Possession of a verification is equivalent to Tarskian truth

Theorem 1. Modulo *a metatheory which contains the mathematics of* $\overline{\overline{D}}$*-furcating trees of finite depth, we have, for all models* M *with domain* D*,*

$$\exists\Pi\ \mathcal{V}(\Pi, \varphi, M, D) \ \Leftrightarrow\ M \Vdash \varphi,\ \textit{i.e., } \varphi \textit{ is true in } M$$

where the right-hand side is in the sense of Tarski.

We are explaining here what Tarskian truth (in a model M) *consists in.* It consists in the *existence of an M-relative verification* of the sentence in question.

Proof. The proof is by the obvious induction on the complexity of the sentence (or saturated formula) φ. Note that the proof is intuitionistic, provided only that the M-relative truthmakers for universals and the M-relative falsitymakers for existentials, in the case where D is infinite, can be assumed to exist, courtesy of the background metamathematics. This observation affects the right-to-left direction in the relevant cases of the inductive step.

5 Bivalence and Non-Contradiction

We can see from the rules of evaluation for negation that Bivalence will hold for evaluations with respect to any interpretation M : every sentence being evaluated will either come out true (by having an M-relative verification), or come out false (by having an M-relative falsification).

Moreover, the Law of Non-Contradiction will hold at the meta-level: no sentence will have both an M-relative verification and an M-relative falsification. This can be proved by induction on the complexity of sentences, by appeal to the basis fact that the interpretation M is both *atomically complete* and *coherent.*

6 Framing rules of natural deduction by morphing rules of evaluation

The rules of *verification* and of *falsification*, to repeat, are *model-relative.* But rules of inference for *deduction* have to be *model-invariant.*

Our rules of inference for natural deduction are nevertheless going to be sourced in the rules of evaluation.

The rules of verification and of falsification will be respectively 'morphed into' *model-invariant* rules of introduction and elimination in natural deductions. The morphing will preserve as much of the former rules' overall features as possible. One such feature has already been mentioned: major premises for eliminations (like major premises earlier, for falsifications) must stand proud, with no proof-work above them. Another important feature that will be preserved is the *relevance* of constituents to compounds—now in the deductive setting, as earlier in the evaluational or computational setting. This means that in the deductive setting we continue to *respect the boxes* that annotated discharges in the evaluational setting.

The morphings also yield the Right and Left rules for the logical operators in sequent calculus. These sequent rules *are* the Introduction and Elimination rules; they are just set out in a slightly different format.

This is how we shall pass from rules of evaluation to rules of deduction.

1. Replace sets Λ of literals with sets Δ of arbitrary sentences.

2. In the $\mathcal{F}$-rules for $\wedge$, $\vee$, $\rightarrow$, $\forall$ and $\exists$, allow for an arbitrary sentence θ as the conclusion, as an alternative to $\perp$. One can still, of course, have $\perp$ as a conclusion (as in instance of θ).

3. In step (2) above, when formulating $\vee$-E from $\vee$-$\mathcal{F}$, allow for a sentence θ to be the overall conclusion even when it is the conclusion of just one of the case-proofs, the other case-proof concluding with $\perp$. This is called 'liberalized' $\vee$-E.

4. In step (2) above, when formulating $\forall$-I from $\forall$-$\mathcal{V}$, and $\exists$-E from $\exists$-$\mathcal{F}$, replace the domain-many *sub-evaluations of instances* with a *single subproof template* involving a parameter a. This respects the fact that the aim of deduction is to preserve truth over all models, and to do so by means of finitary proofs.

The following clarificatory remarks should help one understand the morphings.

1. We shall use V (with or without numerical subscripts) as a placeholder for verifications.

2. Reminder: A verification has a sentence as its conclusion.

3. We shall use F (with or without numerical subscripts) as a placeholder for falsifications.

4. Reminder: A falsification has the absurdity symbol $\bot$ as its conclusion.

5. We shall use Λ (with or without numerical subscripts) as a placeholder for sets of literals of the model, relative to which a sentence is being evaluated.

6. We shall use Π (with or without numerical subscripts) as a placeholder for proofs (natural deductions).

7. A proof can have either a sentence or the absurdity symbol $\bot$ as its conclusion.

8. We shall use Δ (with or without numerical subscripts) as a placeholder for sets of sentences (the set of premises of a proof)

9. We shall color in brown the changes wrought by the morphings, which will be indicated by the squiggly arrow $\rightsquigarrow$.

10. We first take all the Verification rules, and morph them into Introduction rules:

$$(@\text{-}\mathcal{V}) \qquad \rightsquigarrow \qquad (@\text{-I})$$

11. We then take all the Falsification rules, and morph them into Elimination rules:

$$(@\text{-}\mathcal{F}) \qquad \rightsquigarrow \qquad (@\text{-E})$$

6.1 Morphing verification rules to introduction rules

6.1.1 Morphing $\neg$-$\mathcal{V}$ to $\neg$-I

$$(\neg\text{-}\mathcal{V}) \quad \dfrac{\begin{array}{c}\underbrace{\overset{\square\!-\!(i)}{\varphi}\;,\;\Lambda}_{F}\\ \bot\end{array}}{\neg\varphi}(i) \qquad \rightsquigarrow \qquad (\neg\text{-I}) \quad \dfrac{\begin{array}{c}\underbrace{\overset{\square\!-\!(i)}{\varphi}\;,\;\Delta}_{\Pi}\\ \bot\end{array}}{\neg\varphi}(i)$$

6.1.2 Morphing $\wedge$-$\mathcal{V}$ to $\wedge$-I

$$(\wedge\text{-}\mathcal{V})\quad \dfrac{\begin{array}{c}\Lambda_1\\ V_1\\ \varphi_1\end{array}\qquad\begin{array}{c}\Lambda_2\\ V_2\\ \varphi_2\end{array}}{\varphi_1\wedge\varphi_2}\qquad\rightsquigarrow\qquad(\wedge\text{-I})\quad\dfrac{\begin{array}{c}\Delta_1\\ \Pi_1\\ \varphi_1\end{array}\qquad\begin{array}{c}\Delta_2\\ \Pi_2\\ \varphi_2\end{array}}{\varphi_1\wedge\varphi_2}$$

6.1.3 Morphing $\vee$-$\mathcal{V}$ to $\vee$-I

$$(\vee\text{-}\mathcal{V})\quad \dfrac{\begin{array}{c}\Lambda\\ V\\ \varphi\end{array}}{\varphi\vee\psi}\qquad\dfrac{\begin{array}{c}\Lambda\\ V\\ \psi\end{array}}{\varphi\vee\psi}\qquad\rightsquigarrow\qquad(\vee\text{-I})\quad\dfrac{\begin{array}{c}\Delta\\ \Pi\\ \varphi\end{array}}{\varphi\vee\psi}\qquad\dfrac{\begin{array}{c}\Delta\\ \Pi\\ \psi\end{array}}{\varphi\vee\psi}$$

6.1.4 Morphing $\rightarrow$-$\mathcal{V}$ to $\rightarrow$-I

$$(\rightarrow\text{-}\mathcal{V})\quad \dfrac{\begin{array}{c}\square\text{—}(i)\\ \underbrace{\varphi\ ,\ \Lambda}\\ F\\ \bot\end{array}}{\varphi\rightarrow\psi}{\scriptstyle(i)}\qquad\dfrac{\begin{array}{c}\Lambda\\ V\\ \psi\end{array}}{\varphi\rightarrow\psi}\qquad\rightsquigarrow\qquad(\rightarrow\text{-I})\quad\dfrac{\begin{array}{c}\square\text{—}(i)\\ \underbrace{\varphi\ ,\ \Delta}\\ \Pi\\ \bot\end{array}}{\varphi\rightarrow\psi}{\scriptstyle(i)}\qquad\dfrac{\begin{array}{c}\Delta\\ \Pi\\ \psi\end{array}}{\varphi\rightarrow\psi}$$

Note that these two parts of ($\rightarrow$-I) resulting from this direct morphing of the verification rule ($\rightarrow$-$\mathcal{V}$) do not yet furnish that form of ($\rightarrow$-I) (known as the rule of 'conditional proof') that permits one to assume φ for the sake of argument, then to deduce ψ by using assumption φ, and finally to discharge φ when one infers the conclusion $\varphi\rightarrow\psi$. We shall see in due course how to get this 'missing part' of ($\rightarrow$-I).

6.1.5 Morphing $\forall$-$\mathcal{V}$ to $\forall$-I

$$(\forall\text{-}\mathcal{V})\quad \dfrac{\left\{\begin{array}{c}\Lambda_{\boldsymbol{\alpha}}\\ V_{\boldsymbol{\alpha}}\\ \psi(\boldsymbol{\alpha})\end{array}\right\}_{\boldsymbol{\alpha}\in M}}{\forall x\psi(x)}\qquad\rightsquigarrow\qquad(\forall\text{-I})\quad\dfrac{\begin{array}{c}\Delta^{\cancel{a}}\\ \Pi\\ \psi(a)\end{array}}{\forall x\psi(x)}$$

6.1.6 Morphing $\exists$-$\mathcal{V}$ to $\exists$-I

$$(\exists\text{-}\mathcal{V})\quad \dfrac{\begin{matrix}\Lambda\\ V\\ \varphi^x_{\boldsymbol{\alpha}}\end{matrix}}{\exists x\varphi}\quad \begin{matrix}\text{where } \boldsymbol{\alpha} \text{ is any}\\ \text{individual in}\\ \text{the domain}\end{matrix}\quad \rightsquigarrow \quad (\exists\text{-I})\quad \dfrac{\begin{matrix}\Delta\\ \Pi\\ \varphi^x_t\end{matrix}}{\exists x\varphi}\quad \begin{matrix}\text{where } t \text{ is any}\\ \text{closed term}\end{matrix}$$

6.2 Morphing falsification rules to elimination rules

6.2.1 Morphing $\neg$-$\mathcal{F}$ to $\neg$-E

$$(\neg\text{-}\mathcal{F})\quad \dfrac{\neg\varphi \qquad \begin{matrix}\Lambda\\ V\\ \varphi\end{matrix}}{\bot}\quad \rightsquigarrow \quad (\neg\text{-E})\quad \dfrac{\neg\varphi \qquad \begin{matrix}\Delta\\ \Pi\\ \varphi\end{matrix}}{\bot}$$

6.2.2 Morphing $\wedge$-$\mathcal{F}$ to $\wedge$-E

$$(\wedge\text{-}\mathcal{F})\quad \dfrac{\varphi\wedge\psi \qquad \begin{matrix}\underbrace{\overset{\square\text{—}(i)}{\varphi}\;,\;\Lambda}\\ F\\ \bot\end{matrix}}{\bot}{\scriptstyle(i)} \quad \dfrac{\varphi\wedge\psi \qquad \begin{matrix}\underbrace{\overset{\square\text{—}(i)}{\psi}\;,\;\Lambda}\\ F\\ \bot\end{matrix}}{\bot}{\scriptstyle(i)} \quad \rightsquigarrow \quad (\wedge\text{-E})\quad \dfrac{\varphi\wedge\psi \qquad \begin{matrix}\underbrace{\overset{\square\text{—}(i)}{\varphi}\;,\;\Delta}\\ \Pi\\ \theta/\bot\end{matrix}}{\theta/\bot}{\scriptstyle(i)} \quad \dfrac{\varphi\wedge\psi \qquad \begin{matrix}\underbrace{\overset{\square\text{—}(i)}{\psi}\;,\;\Delta}\\ \Pi\\ \theta/\bot\end{matrix}}{\theta/\bot}{\scriptstyle(i)}\;;$$

hence, more efficiently:

$$\dfrac{\varphi\wedge\psi \qquad \begin{matrix}\underbrace{\overset{(i)\text{—}\square\text{—}(i)}{\varphi\,,\,\psi}\;,\;\Delta}\\ \Pi\\ \theta/\bot\end{matrix}}{\theta/\bot}{\scriptstyle(i)}\;,$$

where the box between the two discharge strokes indicates that at least one of φ and ψ must feature as an undischarged assumption of the subproof Π.

6.2.3 Morphing $\vee$-$\mathcal{F}$ to $\vee$-E

$$(\vee\text{-}\mathcal{F})\quad \dfrac{\varphi_1\vee\varphi_2 \qquad \begin{array}{c}\underbrace{\overset{\Box\text{---}(i)}{\varphi_1}\;,\;\Lambda_1}\\ F_1\\ \bot\end{array} \qquad \begin{array}{c}\underbrace{\overset{\Box\text{---}(i)}{\varphi_2}\;,\;\Lambda_2}\\ F_2\\ \bot\end{array}}{\bot}\,(i) \quad\rightsquigarrow\quad (\vee\text{-E})\quad \dfrac{\varphi_1\vee\varphi_2 \qquad \begin{array}{c}\underbrace{\overset{\Box\text{---}(i)}{\varphi_1}\;,\;\Delta_1}\\ \Pi_1\\ \theta/\bot\end{array} \qquad \begin{array}{c}\underbrace{\overset{\Box\text{---}(i)}{\varphi_2}\;,\;\Delta_2}\\ \Pi_2\\ \theta/\bot\end{array}}{\theta/\bot}\,(i)$$

'If either of the two case-proofs Π_1, Π_2 has $\bot$ as its conclusion, bring down as the main conclusion the conclusion of the other case-proof.'

6.2.4 Morphing $\to$-$\mathcal{F}$ to $\to$-E

$$(\to\text{-}\mathcal{F})\quad \dfrac{\varphi\to\psi \qquad \begin{array}{c}\Lambda_1\\ V\\ \varphi\end{array} \qquad \begin{array}{c}\underbrace{\overset{\Box\text{---}(i)}{\psi}\;,\;\Lambda_2}\\ F\\ \bot\end{array}}{\bot}\,(i) \quad\rightsquigarrow\quad (\to\text{-E})\quad \dfrac{\varphi\to\psi \qquad \begin{array}{c}\Delta_1\\ \Pi_1\\ \varphi\end{array} \qquad \begin{array}{c}\underbrace{\overset{\Box\text{---}(i)}{\psi}\;,\;\Delta_2}\\ \Pi_2\\ \theta/\bot\end{array}}{\theta/\bot}\,(i)$$

6.2.5 Morphing $\forall$-$\mathcal{F}$ to $\forall$-E

$$(\forall\text{-}\mathcal{F})\quad \dfrac{\forall x\psi(x) \qquad \begin{array}{c}\underbrace{\overset{\Box\text{------}(i)}{\psi(\boldsymbol{\alpha})}\;,\;\Lambda}\\ F\\ \bot\end{array}}{\bot}\,(i) \quad\rightsquigarrow\quad (\forall\text{-E})\quad \dfrac{\forall x\psi(x) \qquad \begin{array}{c}\underbrace{\overset{\Box\text{------}(i)}{\psi(t)}\;,\;\Delta}\\ \Pi\\ \theta/\bot\end{array}}{\theta/\bot}\,(i)\ ;$$

where $\boldsymbol{\alpha}$ is any individual in the domain

where t is any closed term

hence

$$
\dfrac{\forall x\psi(x) \qquad \begin{array}{c} \underbrace{\overset{(i)}{\overline{\psi(t_1)}} \;,\; \ldots \;,\; \overset{\cdots\,\square\,\cdots}{} \overline{\psi(t_n)}^{(i)} \;,\; \Delta}_{\Pi} \\ \theta/\bot \end{array}}{\theta/\bot}\,(i)
$$

where $t_1, \ldots, t_n$ are closed terms, and the box indicates that at least one instance $\psi(t_i)$ must feature as an undischarged assumption of the subproof Π.

6.2.6 Morphing $\exists$-$\mathcal{F}$ to $\exists$-E

$$
(\exists\text{-}\mathcal{F}) \quad \dfrac{\exists x\varphi \qquad \left\{ \begin{array}{c} \square\overline{}^{(i)} \\ \underbrace{\varphi^x_{\boldsymbol{\alpha}} \,,\, \Lambda_{\boldsymbol{\alpha}}} \\ F_{\boldsymbol{\alpha}} \\ \bot \end{array} \right\}_{\boldsymbol{\alpha}\in M}}{\bot}\,(i)
\quad \rightsquigarrow \quad
(\exists\text{-E}) \quad \dfrac{\exists x\varphi^{\cancel{a}} \qquad \begin{array}{c} \square\overline{}^{(i)} \\ \underbrace{\varphi^x_a \,,\, \Delta^{\cancel{a}}}_{\Pi} \\ \theta^{\cancel{a}}/\bot \end{array}}{\theta^{\cancel{a}}/\bot}\,(i)
$$

7 Allowing $\to$-I to *discharge* an assumption

Note that we have thus far obtained an introduction rule for $\to$ in only the two parts that correspond to the respective considerations that it suffices, for the truth of a conditional, to have its antecedent false, or to have its consequent true. But this leaves out of the picture the most familiar form of $\to$-Introduction, known as Conditional Proof, which allows for the discharge of the antecedent φ as an assumption if it has been used in the derivation of the consequent ψ as the conclusion of the subordinate proof:

$$
\dfrac{\begin{array}{c} \overline{}^{(i)} \\ \varphi \\ \vdots \\ \psi \end{array}}{\varphi \to \psi}\,(i)
$$

The first two parts of $\to$-Introduction immediately below were obtained by morphing the rule of $\to$-Verification. The third part—

the usual 'Rule of Conditional Proof' allowing discharge of the antecedent-as-assumption—can now be added to them:

$$\dfrac{\begin{matrix}\overset{\Box\overline{}^{(i)}}{\underbrace{\varphi\ ,\ \Delta}}\\ \Pi\\ \bot\end{matrix}}{\varphi\to\psi}\,(i) \qquad \dfrac{\begin{matrix}\Delta\\ \Pi\\ \psi\end{matrix}}{\varphi\to\psi} \qquad \dfrac{\begin{matrix}\overset{\overline{}^{(i)}}{\underbrace{\varphi\ ,\ \Delta}}\\ \Pi\\ \psi\end{matrix}}{\varphi\to\psi}\,(i)$$

This is in light of the *classical* derivations that follow. They employ Bivalence (in the form of the rule of Dilemma) at their final steps, marked (2). The first derivation uses Dilemma on the antecedent φ as its positive horn-assumption; the second derivation uses Dilemma on the consequent ψ as the same:

$$\dfrac{\dfrac{\begin{matrix}\overline{\varphi}^{(2)}\\ \vdots\\ \psi\end{matrix}}{\varphi\to\psi} \quad \dfrac{\dfrac{{}^{(2)}\overline{\neg\varphi} \quad \overline{\varphi}^{(1)}}{\bot}}{\varphi\to\psi}\,(1)}{\varphi\to\psi}\,(2) \qquad \dfrac{\dfrac{{}^{(2)}\overline{\psi}}{\varphi\to\psi} \quad \dfrac{\dfrac{{}^{(2)}\overline{\neg\psi} \quad \begin{matrix}\overline{\varphi}^{(1)}\\ \vdots\\ \psi\end{matrix}}{\bot}}{\varphi\to\psi}\,(1)}{\varphi\to\psi}\,(2)$$

8 Succinct graphic statement of $\to$-I

We now have the following three 'parts' of the rule of $\to$-Introduction:

$$\dfrac{\begin{matrix}\overset{\Box\overline{}^{(i)}}{\underbrace{\varphi\ ,\ \Delta}}\\ \Pi\\ \bot\end{matrix}}{\varphi\to\psi}\,(i) \qquad \dfrac{\begin{matrix}\Delta\\ \Pi\\ \psi\end{matrix}}{\varphi\to\psi} \qquad \dfrac{\begin{matrix}\overset{\overline{}^{(i)}}{\underbrace{\varphi\ ,\ \Delta}}\\ \Pi\\ \psi\end{matrix}}{\varphi\to\psi}\,(i)$$

The second and third of these can be melded so that the display becomes

$$\dfrac{\begin{array}{c}\overset{\Box\overline{\quad\ }(i)}{\underbrace{\varphi\ ,\ \Delta}}\\ \Pi\\ \bot\end{array}}{\varphi\rightarrow\psi}{\scriptstyle(i)} \qquad\qquad \dfrac{\begin{array}{c}\overset{\Diamond\overline{\quad\ }(i)}{\underbrace{\varphi\ ,\ \Delta}}\\ \Pi\\ \psi\end{array}}{\varphi\rightarrow\psi}{\scriptstyle(i)}$$

The diamond in the graphic rule-part on the *right* says 'vacuous' discharge is permitted: φ need not be an undischarged assumption of the subordinate proof Π. This is exactly the rule of ($\rightarrow$-I) in the Gentzen–Prawitz tradition. The graphic rule-part on the *left* is a 'new part' of ($\rightarrow$-I) supplied by the Core logician *via* the foregoing method of morphing. It directly respects the third and fourth lines of the truth table for $\rightarrow$. It is derivable in the Gentzen–Prawitz system by appeal to their rule of EFQ, *which we, by contrast, eschew.*

It is easy to see how to re-write the foregoing rules of natural deduction as sequent rules. Introduction rules can be re-written as Right rules, and Elimination rules can be re-written as Left rules. A single illustration, say with the connective $\rightarrow$, should suffice to indicate the way this is done.

$$(\rightarrow\text{-I}) \quad \dfrac{\begin{array}{c}\overset{\Box\overline{\quad\ }(i)}{\underbrace{\varphi\ ,\ \Delta}}\\ \vdots\\ \bot\end{array}}{\varphi\rightarrow\psi}{\scriptstyle(i)} \quad \dfrac{\begin{array}{c}\overset{\Diamond\overline{\quad\ }(i)}{\underbrace{\varphi\ ,\ \Delta}}\\ \vdots\\ \psi\end{array}}{\varphi\rightarrow\psi}{\scriptstyle(i)} \quad \text{are re-written as} \quad (\rightarrow\text{-R}) \quad \dfrac{\Delta,\varphi:\bot}{\Delta:\varphi\rightarrow\psi} \quad \dfrac{\Delta:\psi}{\Delta\setminus\{\varphi\}:\varphi\rightarrow\psi}$$

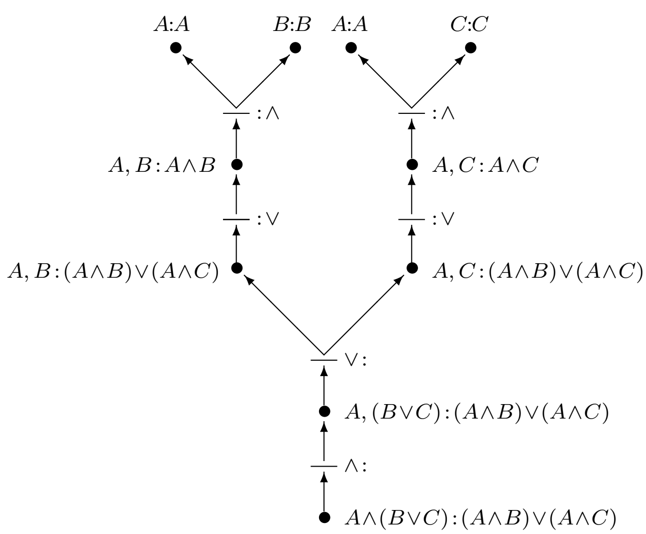
$$(\rightarrow\text{-E}) \quad \dfrac{\varphi\rightarrow\psi \quad \begin{array}{c}\Delta_1\\ \vdots\\ \varphi\end{array} \quad \begin{array}{c}\overset{\Box\overline{\quad\ }(i)}{\underbrace{\psi\ ,\Delta_2}}\\ \vdots\\ \theta\end{array}}{\theta}{\scriptstyle(i)} \quad \text{is re-written as} \quad (\rightarrow\text{-L}) \quad \dfrac{\Delta_1:\varphi \quad \Delta_2,\psi:\theta}{\Delta_1,\Delta_2,\varphi\rightarrow\psi:\theta}$$

The result of this re-writing of rules of natural deduction as their corresponding rules of the sequent calculus is that every core proof in natural deduction is 'isomorphic' to the corresponding core proof in the sequent calculus. Here is an illustration of this.

$$\dfrac{A\wedge(B\vee C) \qquad \dfrac{{}^{(2)}\dfrac{}{B\vee C} \quad \dfrac{\dfrac{{}^{(2)}\dfrac{}{A}\quad\dfrac{}{B}{}^{(1)}}{A\wedge B}}{(A\wedge B)\vee(A\wedge C)} \quad \dfrac{\dfrac{{}^{(2)}\dfrac{}{A}\quad\dfrac{}{C}{}^{(1)}}{A\wedge C}}{(A\wedge B)\vee(A\wedge C)}}{(A\wedge B)\vee(A\wedge C)}{}^{(1)}}{(A\wedge B)\vee(A\wedge C)}{}^{(2)}$$

$$\dfrac{\dfrac{\dfrac{\dfrac{A:A \quad B:B}{A,B:A\wedge B}}{A,B:(A\wedge B)\vee(A\wedge C)} \qquad \dfrac{\dfrac{A:A \quad C:C}{A,C:A\wedge C}}{A,C:(A\wedge B)\vee(A\wedge C)}}{A,B\vee C:(A\wedge B)\vee(A\wedge C)}}{A\wedge(B\vee C):(A\wedge B)\vee(A\wedge C)}$$

Next we illustrate the tree structure of a successful proof search for the foregoing result.



Core proofs can now be conceived of as the naturally arising objects that can be directly constructed by the method of proof search that proceeds by breaking down deductive problems into the naturally arising immediate subproblems resulting from either (i) choosing to focus on the dominant operator of the sought conclusion, so as to end up with a terminal Introduction (equivalently, application of a Right rule); or (ii) choosing to focus on the dominant operator in an available premise, so as to end up with a terminal Elimination (equivalently, application of a Left rule). Efficient proof-search is a matter of alternating insightfully between (i) and (ii) as the search proceeds. Normalization results in proof theory yield many such useful insights into how the syntactic form of a sequent to be proved can guarantee that if there is a proof at all to be had, then there is a proof of such-and-such a constrained form. (See Tennant [1992].) To this end, similarly, a valuable constraint leading one to make successful choices during proof-search is the 'relevance filtration' afforded by the relevance property $\mathcal{R}(\Delta, \varphi)$ explicated in Tennant [2015], and proved to hold between the premise-set Δ and the conclusion φ of any Classical Core proof.

9 Summary of main points

One starts with model-relative rules of evaluation, i.e. of verification and of falsification. One morphs those rules respectively into model-invariant introduction and elimination rules of natural deduction (or, equivalently, into Right rules and Left rules of sequent calculus).

The method of morphing, by its very design, produces no such rule as *Ex Falso Quodlibet* in natural deduction. Nor does it produce any rule of Cut or of Thinning in the sequent calculus. We therefore avoid the First Lewis Paradox ($A, \neg A : B$), which is the single most important aim of the relevantist.

Important features of the evaluation rules are preserved by the morphing. These are

- Major premises of falsifications stood proud; so too do major premises of elimination.

- Discharges were obligatory in evaluations; so too are they in natural deductions.

- Falsification rules were parallelized; so too are the elimination rules.

- Natural deductions become more 'general' by having arbitrary premise sets in place of sets of literals; and by having arbitrary sentences as conclusions of eliminations, in place of $\perp$ as the conclusion of a falsification.

Semantic *evaluations* are perforce *relevant*. This relevance is preserved by our morphing of rules of *evaluation* into rules of natural *deduction*.

Every rule of natural deduction (and every rule in the sequent calculus) preserves fortuitous epistemic gains (i.e., gains in logical strength). At any stage in a search for a proof if one succeeds fortuitously in finding a proof of a logically stronger result than the actual sequent that one set out to prove, that stronger result can be used at that very point in the construction of the desired proof. Put another way: logically stronger solutions to deductive subproblems generated during proof-search can always be 'inherited' to help deliver a stronger (and shorter) solution to the problem that generated those subproblems. The isomorphism between natural deductions and sequent proofs affords a conception of proofs as abstract objects of search, with their formal presentation being a matter of perspicuousness or pragmatic convenience.

10 Final note

A reviewer helpfully inquired how the present study might be situated within the tradition of 'bilateralism' in proof theory. That tradition derives from the seminal paper Rumfitt [2000], where the words 'bilateral', 'bilateralist', and 'bilateralism' were first introduced. The very first occurrence of any of these words was the occurrence of 'bilateral' on page 790, where Rumfitt wrote

> The standard introduction and elimination rules for "$\wedge$" and "$\vee$" and "$\rightarrow$" are harmonious in *this bilateral sense*. [Emphasis added]

The immediately preceding context supplied the sense in question. Rumfitt generously wrote

> For reasons that Neil Tennant (1987, pp. 94–97) has explained, ... it makes sense to favour the strongest elim-

> ination rule that is a Prawitzian inverse of a given introduction rule.[fn] Tennant, indeed, shows how to detach Gentzen's underlying idea from the explanatory priority that he had accorded to introduction rules. Let us say that an introduction rule I is in harmony with an elimination rule E when (a) E's major premiss expresses the weakest proposition that can be eliminated using E, with I taken as given, and (b) I's conclusion expresses the strongest proposition that can be introduced using I, with E taken as given.

On page 97 of the passage that Rumfitt cited, the present author suggested that

> ... there is an intrinsic meaning to conjunction, for example, that is invariant across minimal, intuitionistic, and classical logic. I intend thereby to reveal as unjustifiable marginal excesses or excrescences the extra ingredients in the consequence relation of classical logic that have earned the generic labels (a) the fallacies of relevance, and (b) the classical laws of negation. The intrinsic meanings of the logical operators, characterized 'from below', as it were, provide no justification whatever for this fleshing out of the correct consequence relation.

The present study can be understood as underscoring further that very suggestion. Our methodology of morphing rules of evaluation into rules of deduction has avoided the fallacies of relevance. It has also left us in a position to eschew the strictly classical rules of negation (notwithstanding the appearance of classicism in our BHK-appropriation above of the Rule of Conditional Proof).

Rumfitt attributed another important facet of bilateralism as deriving from investigations by the present author. Just as in bilateralism the values T and F are coequal, along with the respective notions of assertion and rejection, so too in proof theory the notions of *proof* and of *disproof* are coequal. Rumfitt wrote (*loc. cit.*, p. 794

> ... as Tennant has put it, the expression [$\perp$] plays the role of a punctuation mark in the deduction (1999). It marks the point where the supposition ... has been shown

to lead to a logical dead end, and is thus discharged, prior to an assertion of its negation.

The work Tennant [1999] that Rumfitt cites presented the notions of proof and of disproof as coinductively defined. It is in the same spirit that we have here deployed the rules for generating the coinductively defined constructs that we called verifications and falsifications; and then morphed these rules into rules of natural deduction. In keeping with the metalogical prescription of Tennant [1999], the latter rules can likewise be regarded as generating the coequal constructs that we called proofs and disproofs. For want of space here, we set aside for further investigation the interesting question whether Rumfitt's bilateralism should shift one's choice of logic from Core Logic $\mathbb{C}$ to its classicized extension Classical Core Logic $\mathbb{C}^{+}$.

References

Ian Rumfitt. "Yes" and "No". *Mind*, 109(436):781–823, 2000.

Neil Tennant. *Anti-Realism and Logic: Truth as Eternal.* Clarendon Library of Logic and Philosophy, Oxford University Press, Oxford. June, 1987.

Neil Tennant. *Autologic.* Edinburgh University Press, 1992.

Neil Tennant. Negation, Absurdity and Contrariety. In Dov Gabbay and Heinrich Wansing, editors, *What is Negation?*, pages 199–222. Kluwer, Dordrecht, 1999.

Neil Tennant. Inferential Semantics. In Jonathan Lear and Alex Oliver, editors, *The Force of Argument: Essays in Honor of Timothy Smiley*, pages 223–57. Routledge, London, 2010.

Neil Tennant. The Relevance of Premises to Conclusions of Core Proofs. *Review of Symbolic Logic*, 8(4):743–784, 2015.

Neil Tennant. *Core Logic.* Oxford University Press, Oxford, 2017.

Neil Tennant. A Logical Theory of Truthmakers and Falsitymakers. In Michael Glanzberg, editor, *Handbook on Truth*, pages 355–393. Oxford University Press, Oxford, 2018.