
A Proof Procedure for Adaptive Logics

PETER VERDEE, *Centre for Logic and Philosophy of Science, Ghent University, Blandijnberg 2, B-9000 Ghent, Belgium.*
E-mail: peter.verdee@ugent.be.

Abstract

In this paper I present a procedure that generates adaptive proofs for finally derivable adaptive logic consequences. The proof procedure for the inconsistency adaptive logic \mathbf{CLuN}^f is already presented in [10]. In this paper the procedure for \mathbf{CLuN}^m is presented and the results for both logics are generalized to all adaptive logics, on the presupposition that there exists a total proof procedure for lower limit logic derivability of the adaptive logic and a finite set of problem relevant abnormalities.

Keywords: adaptive logic, dynamic proofs, proof procedure, formal problem solving, goal directed proofs

1 Introduction

In this paper I present a procedure that generates adaptive logic proofs for finally derivable adaptive logic consequences. Let a proof procedure for an adaptive logic be a procedure that, if it terminates, returns an adaptive logic proof for some formula G from a premise set Γ whenever G is a finally derivable consequence of Γ , and returns a negative answer whenever G is not a finally derivable consequence of Γ . A proof procedure is partial iff it does not always terminate and total iff it does.

Adaptive logics are logics that formalize defeasible reasoning forms. The first adaptive logics were inconsistency adaptive logics (see [4]). These logics can cope with inconsistent theories by localizing the inconsistencies in the theories. They interpret these theories as consistently as possible. Nowadays, adaptive logic forms a wide research area, in which logical solutions are developed for different common sense and scientific reasoning notions: induction (see [9], [11], and [13]), abduction (see [33] and [34]), compatibility (see [14] and [31]), causality (see [23] and [42]), prioritized reasoning (see [51], [47], [46] and [52]), relevance ([6]), ambiguity (see [48], [44], and [45]), vagueness (see [43], [50], and [41]), and diagnosis (see [15]). They all share the same metatheoretical structure and hence they can be examined in general.

There are logical solutions for defeasible reasoning forms outside of the adaptive logic framework, but it has been shown for a number of these logics and logical mechanisms that they can be characterized by an adaptive logic. Moreover, this characterization led for several systems to an interesting strengthening or variant. Among the finished results are [5], [7], [17] and [52] for the consequence relations from [38], [18] and [19]; [29] and [28] for [53]; [34] for [1]; [32] for the notion of empirical progress from [26]; [30] for [35] and [21]; [22] and [3] for default reasoning and circumscription respectively (see [2], [20] and [27]).

Adaptive logics have a typical dynamic proof theory. Lines of adaptive proofs are

2 A Proof Procedure for Adaptive Logics

conditional. As the proof continues, some lines may get marked, denoting that the formula of this line is not derived at that stage of the proof. Markings may come and go. These dynamic proofs explicate the actual defeasible reasoning processes humans use in the dynamic reasoning contexts for which adaptive logics are developed.

For most interesting adaptive logics, there is also a stable derivability notion. From some (possibly infinite¹) stage of the proof on, some lines are unmarked and will not be marked in any possible extension of the proof. Adaptive logicians call the formulas that are derived on these stably unmarked lines the finally derived consequences. They constitute a consequence relation called final derivability. This stable consequence relation is the actual adaptive logic consequence relation.

The final derivability consequence set is equivalent to the following semantic notion: the set of all formulas that are true in all models of the premises that are as normal as possible. How the ambiguous expression “as normal as possible” is to be specified depends on the abnormalities and the strategy of the specific logic. An inconsistency adaptive logic with Minimal Abnormality strategy for example selects the models of premises that verify as little (in the set theoretic sense) inconsistencies as possible (the abnormalities are in this case the inconsistencies). The consequence set is the set of all formulas that are true in all the selected models.

There is no positive test for adaptive logics in general². Hence, there cannot be a generally applicable algorithm that constructs adaptive proofs in a finite time in such a way that some formula is derived if and only if it is finally derivable. This does not exclude the decidability of a very wide range of concrete adaptive logic problems³. So, even in these complex problem solving contexts, creating proof procedures is possible and useful. Evidently, it is sensible to try to solve a decidable problem, but it makes also sense to try to solve undecidable problems. The reasoner may not be aware of the undecidability when he starts the process or he may prefer an uncertain but defensible solution above a purely random guess.

Moreover, it is important to develop procedural approaches to adaptive logic because adaptive logics are devised as useful tools to understand creative human reasoning processes and not as abstract standards of deduction. Adaptive logic proofs form explications of actual reasoning processes rather than demonstrations of the correctness of statements. The explicated reasoning processes are often parts of concrete problem solving processes. Procedures show how an agent is able to solve his problem in some adaptive logic context and which heuristics he can apply when he is solving the problem. The procedure I present in this paper, shows how the agent can be rationally critical towards his own defeasible derivations, and hence, how he is able to gain certainty about the final derivability of interesting statements.

The procedure I will present uses an existing system for goal directed reasoning (elsewhere also called prospective dynamics with prospective proofs). It is described in [37] and [16] and is already developed for several different logics. It can be seen as

¹There is an alternative but equivalent notion, that eliminates the fact that some formulas are only finally derived at infinite stages (see next section).

²In [49] it is shown that *usual* Minimal Abnormality predicative adaptive logics are Π_1^1 -complex and in [25] that *usual* Reliability predicative adaptive logics are Σ_3^0 -complex. *Usual* propositional adaptive logics turn out to be as complex if one considers infinite premise sets. Usual adaptive logics are adaptive logics in standard format with a lower limit logic that falls within the same complexity class as classical logic, and that has the ability to express abnormalities and classical disjunctions in the object language.

³An adaptive logic problem is a question whether some G is a finally derivable consequence of some Γ . Solving a problem is answering that question.

a mix of a tableaux method (it forms a decision method for the consequence relation) and a proof theory (it defines a special type of proofs). It is defined by the so called goal directed proof format and a goal directed heuristics. A line of a goal directed proof contains, apart from a normal formula, a condition element. This is a set of formulas such that the formula of the line is a consequence of this set plus the set of premises. The heuristics guides the reasoner to the derivation of the goal formula on an empty condition by starting with a line that contains this goal (with the goal itself as condition), allowing only for the analysis of formulas, and only introducing premises and analyzing formulas when this is possibly useful.

It is important to stress that the procedure I will give, does not only generate a yes/no answer to a problem of the form $\Gamma \vdash_{\mathbf{AL}} A$, but also gives a proof for this result and this proof can be seen as the formalization of a reasoning process towards the goal. In such a way the human reasoner can reconstruct the process towards the solution of mechanically solved problems and obtain insight in the problem and its solution. People get to know *why* the answer is yes or no, rather than only finding out *that* the answer is yes or no. When the answer is negative obviously no successful proof can be returned. Still, also in this case all the failed reasoning steps and reasoning steps towards the observation of their failure may turn out to be very useful. The human reasoner is able to obtain insight in the negative conclusion.

The paper is conceived in a modular way. Although I only present a full-blown proof procedure for two actual logics, the propositional fragments of $\mathbf{CLuN}^{\mathbf{m}}$ and $\mathbf{CLuN}^{\mathbf{r}}$, my aim is more general. There is a standard format for adaptive logics, which serves as a generic means to build adaptive logics on lower limit logics. For many important lower limit logics, it is not difficult to construct a proof procedure. In what follows, I will assume that one already has such a procedure. This enables me to describe the proof procedure in a general way. In order to achieve this, the proof procedure is divided into 3 modules: the proof procedure for the lower limit logic, the one for conditional derivability, and finally the one for the final derivability relation of the adaptive logic. Once one has a proof procedure for lower limit logic derivability and one for conditional derivability of a concrete adaptive logic, the procedure for final derivability can be obtained using the general procedure in section 5. Next, for most of the existing adaptive logics, the conditional derivability procedure is also easily obtainable from the lower limit logic procedure, based on the same ideas as the procedure for $\mathbf{CLuN}^{\mathbf{m}}$ or $\mathbf{CLuN}^{\mathbf{r}}$ in section 4, or by means of a brute force procedure when a finite set of relevant abnormalities is isolatable.

The goal directed proofs that result from the procedures are in another proof format than the regular adaptive logic proofs. Nevertheless, they are easily transformable into adaptive logic proofs.

In section 2, I give the proof theoretical and semantical characterization of adaptive logic in standard form, and apply this to obtain the inconsistency adaptive logics $\mathbf{CLuN}^{\mathbf{m}}$ and $\mathbf{CLuN}^{\mathbf{r}}$. In section 3 a proof procedure for \mathbf{CLuN} is presented. In section 4 and section 5 respectively a proof procedure for conditional derivability and one for final derivability (both for Minimal Abnormality and for Reliability) are presented. Finally, in section 6 the correctness of the procedures is demonstrated.

2 Adaptive logics: standard format, \mathbf{CLuN}^m and \mathbf{CLuN}^r

2.1 The standard format of \mathbf{AL}

In this section adaptive logics are very briefly presented (see [12] for an overview and [8] for the philosophical basis). An adaptive logic in standard format is defined as a triple consisting of:

- a **LLL**: a monotonic, reflexive, transitive and compact extension of classical logic (**CL**) which has a characteristic semantics,
- a set of abnormalities: a set of **LLL**-contingent formulas Ω , characterized by a (possibly restricted) logical form, and
- a strategy (the most important strategies in **AL** are ‘Reliability’ and ‘Minimal Abnormality’).

The standard format demands that the **LLL**-language, next to its own standard logical symbols, also encompasses formulas with the standard logical symbols of **CL**. They must behave classically, i.e. they should function in a **CL**-standard manner (e.g. $M \vDash \neg A$ iff $M \not\vDash A$) and do not need to occur in the conclusion or the premises. In this paper, I will denote the **CL**-symbols by means of \neg (negation) and $\check{\vee}$ (disjunction).

2.2 The proof theory of \mathbf{AL}

The proof theory of an **AL** consists of a set of inference rules (determined by the **LLL** and Ω) and a marking definition (determined by Ω and the chosen strategy). A line of an annotated **AL**-proof consists of four elements: (1) a line number i , (2) a formula A , (3) the name of a rule and the line number of the rule premises, (4) a condition consisting of a set of abnormalities $\Theta \subset \Omega$. A stage s of a proof is the subproof that is completed up to line number s . The inference rules govern the addition of lines. There are 3 types of rules.

PREM	If $A \in \Gamma$	$\begin{array}{cc} \dots & \dots \\ A & \emptyset \end{array}$
RU	If $A_1, \dots, A_n \vdash_{\mathbf{LLL}} B$	$\begin{array}{cc} A_1 & \Delta_1 \\ \dots & \dots \\ A_n & \Delta_n \\ \hline B & \Delta_1 \cup \dots \cup \Delta_n \end{array}$
RC	If $A_1, \dots, A_n \vdash_{\mathbf{LLL}} B \check{\vee} Dab(\Theta)$	$\begin{array}{cc} A_1 & \Delta_1 \\ \dots & \dots \\ A_n & \Delta_n \\ \hline B & \Delta_1 \cup \dots \cup \Delta_n \cup \Theta \end{array}$

The classical disjunction of the members of a finite $\Delta \subset \Omega$, $Dab(\Delta)$, is called a *Dab-formula*. $Dab(\Delta)$ is a *minimal Dab-formula* of stage s iff $Dab(\Delta)$ is derived at stage s on the condition \emptyset and no $Dab(\Delta')$ with $\Delta' \subset \Delta$ is derived on the condition \emptyset . The most important strategies are Reliability and Minimal Abnormality.

DEFINITION 2.1

Marking definition for Reliability.

Where $Dab(\Delta_1), \dots, Dab(\Delta_n)$ are the minimal *Dab*-formulas derived on the condition \emptyset at stage s , $U_s(\Gamma) = \Delta_1 \cup \dots \cup \Delta_n$, and Δ is the condition of line i , line i is marked at stage s iff $\Delta \cap U_s(\Gamma) \neq \emptyset$.

DEFINITION 2.2

Marking definition for Minimal Abnormality.

Where $Dab(\Delta_1), \dots, Dab(\Delta_n)$ are the minimal *Dab*-formulas derived on the condition \emptyset at stage s , $\Phi_s^\circ(\Gamma)$ is the set of all sets that contain one member of each Δ_i , $\Phi_s(\Gamma)$ are the $\varphi \in \Phi_s^\circ(\Gamma)$ that are not proper supersets of a $\varphi' \in \Phi_s^\circ(\Gamma)$, A is the formula and Δ is the condition of line i , line i is marked at stage s iff

- (i) there is no $\varphi \in \Phi_s(\Gamma)$ such that $\varphi \cap \Delta = \emptyset$, or
- (ii) for some $\varphi \in \Phi_s(\Gamma)$, there is no line on which A is derived on a condition Θ for which $\varphi \cap \Theta = \emptyset$.

Two types of derivability are defined in **AL**. A formula A is *derived at a stage* iff A is derived on an unmarked line at the stage. A formula A is *finally derived* at stage s iff A is derived on an unmarked line i at stage s and line i will not be marked in any extension of the stage. Or alternatively, a formula is finally derived iff it is derived on an unmarked line and any extension of the proof in which the line is marked, can be further extended to a proof in which the line is unmarked. This latter definition is equivalent to the former, but has the advantage that any finally derivable formula is finally derived at some finite stage of a proof. The finally derivable consequences of a premise set are independent of the stage and constitute the consequence sets for **AL^r** and **AL^m**: $Cn_{\mathbf{AL}^r}(\Gamma)$, respectively $Cn_{\mathbf{AL}^m}(\Gamma)$ and their consequence relations $\vdash_{\mathbf{AL}^r}$, respectively $\vdash_{\mathbf{AL}^m}$.

2.3 The semantics of **AL**

$Dab(\Delta)$ is a *minimal Dab-consequence* of Γ iff $\Gamma \models_{\mathbf{LLL}} Dab(\Delta)$ and, for all $\Delta' \subset \Delta$, $\Gamma \not\models_{\mathbf{LLL}} Dab(\Delta')$. Where $Dab(\Delta_1), Dab(\Delta_2), \dots$ are the minimal *Dab*-consequences of Γ , let $U(\Gamma) =_{df} \Delta_1 \cup \Delta_2 \cup \dots$. Finally, where M is a **LLL**-model, $Ab(M) =_{df} \{A \in \Omega \mid M \models A\}$.

DEFINITION 2.3

Reliable model and the corresponding semantical consequence relation $\models_{\mathbf{AL}^r}$.

A **LLL**-model M of Γ is reliable iff $Ab(M) \subseteq U(\Gamma)$. $\Gamma \models_{\mathbf{AL}^r} A$ iff all reliable models of Γ verify A .

DEFINITION 2.4

Minimally abnormal model and the corresponding semantical consequence relation $\models_{\mathbf{AL}^m}$.

A **LLL**-model M of Γ is minimally abnormal iff there is no **LLL**-model M' of Γ for which $Ab(M') \subset Ab(M)$. $\Gamma \models_{\mathbf{AL}^m} A$ iff all minimally abnormal models of Γ verify A .

2.4 **CLuN^m** and **CLuN^r**

Let us consider the inconsistency-adaptive logics **CLuN^m** and **CLuN^r** (elsewhere these names denote predicative logics, but here I only use their propositional frag-

6 A Proof Procedure for Adaptive Logics

ments). The lower limit logic is the propositional fragment of the paraconsistent logic **CLuN**. **CLuN** is the full positive fragment of **CL** with simple gluts for the negation connective. For any formula A , both A and $\neg A$ may be true in **CLuN** (yet they cannot be both false). The set of abnormalities is $\Omega = \{A \wedge \neg A \mid A \in \mathcal{W}\}$, with \mathcal{W} the set of well formed formulas. The strategies are respectively Minimal Abnormality (resulting in the adaptive logic **CLuN^m**) and Reliability (resulting in the adaptive logic **CLuN^r**). If the strategy of the logic does not matter, I will refer to it as **ACLuN**. In what follows $!A$ will abbreviate $A \wedge \neg A$. Because the negation is the only non-classical symbol in **CLuN**, there is no need to add formulas with symbols $\check{\vee}$ and $\check{\wedge}$ to the language, as they would be equivalent to respectively \vee and \wedge .

2.5 Generic notation

In this paper I give a generally applicable procedure. In order to realize this I use a minimal amount of properties of the adaptive logics under consideration. Let **LLL** denote some **LLL**-ready logic (cf. the properties of the standard format) with a disjunction \vee and a conjunction \wedge with standard behaviour ($M \models_{\text{LLL}} A \vee B$ iff $M \models_{\text{LLL}} A$ or $M \models_{\text{LLL}} B$ and $M \models_{\text{LLL}} A \wedge B$ iff $M \models_{\text{LLL}} A$ and $M \models_{\text{LLL}} B$). Let **AL^m** and **AL^r** denote adaptive logics that use the Minimal Abnormality strategy respectively the Reliability strategy and have some **LLL** with the mentioned connectives. If the strategy does not matter I will simply refer to it as **AL**.

3 A (partial) proof procedure for the lower limit logic CLuN

In this section, a proof procedure for **CLuN** is presented. The procedure generates a special kind of proofs: goal directed proofs. Prospective proofs for **CLuN** have lines that contain, apart from the derived formula, a set of formulas called the *D-condition*:

$$i \quad [\Delta] A \quad \dots \quad \dots$$

A is the formula of the line and Δ the D-condition. I also add an adaptive condition Θ called the *A-condition*. For **CLuN** this element can remain empty and is thus obviously useless. It is added in behalf of the procedures for conditional and final derivability presented in the following sections. These procedures use the lower limit logic rules as well. I add the useless condition already here, to avoid having to list the **CLuN**-rules again, where the extra condition is necessary. With this adjustment, the lines contain *two* conditions:

$$i \quad [\Delta] A \quad \dots \quad \dots \quad \Theta$$

Referring to this line, I will say that A^Θ is derived on line i on D-condition Δ or that $[\Delta]A^\Theta$ is derived on line i . If Δ is empty, $[\emptyset]$ is omitted.

It is not of my concern to give an efficient procedure (a really efficient procedure requires heuristic information with a non formal character). The mere existence of a procedure suffices. The procedure is defined by a set of ordered rules, a few restrictions on the application of the rules, and the command “apply to the first line of the proof to which some rule may be applied, the first permitted rule”. Although the logics under consideration are decidable, I speak of a partial proof procedure to include the

case of infinite premises. In this case one will need a procedure that produces for each possible target A a list of all premises of which A is a positive part⁴. This is always possible when the premise set is recursive.

There are several types of rules. The formula analyzing rules and the condition analyzing rules for **CLuN** may be summarized by distinguishing **a**-formulas from **b**-formulas (varying on a theme from [40]). To each formula two other formulas are assigned according to the following table:

a	a ₁	a ₂		b	b ₁	b ₂
$A \wedge B$	A	B		$\neg(A \wedge B)$	$*A$	$*B$
$A \equiv B$	$A \supset B$	$B \supset A$		$\neg(A \equiv B)$	$\neg(A \supset B)$	$\neg(B \supset A)$
$\neg(A \vee B)$	$*A$	$*B$		$A \vee B$	A	B
$\neg(A \supset B)$	A	$*B$		$A \supset B$	$*A$	B
$\neg\neg A$	A	A				

The formula analyzing rules for **a**-formulas and **b**-formulas are respectively (the ‡ in the name of the rule stands for the logical symbols in the **a**- or **b**-formula that are analyzed, for example when **a** is $\neg(A \vee B)$, ‡ is $\neg\vee$):⁵

$$\begin{aligned} \ddagger E & \frac{[\Delta] \mathbf{a}^\ominus}{[\Delta] \mathbf{a}_1^\ominus \quad [\Delta] \mathbf{a}_2^\ominus} & \ddagger E & \frac{[\Delta] \mathbf{b}^\ominus}{[\Delta \cup \{*\mathbf{b}_2\}] \mathbf{b}_1^\ominus \quad [\Delta \cup \{*\mathbf{b}_1\}] \mathbf{b}_2^\ominus} \\ \neg\neg E & \frac{[\Delta] \neg\neg A^\ominus}{[\Delta] A^\ominus} \end{aligned}$$

The condition analysing rules for **a**-formulas and **b**-formulas are respectively:

$$\begin{aligned} C\ddagger E & \frac{[\Delta \cup \{\mathbf{a}\}] A^\ominus}{[\Delta \cup \{\mathbf{a}_1, \mathbf{a}_2\}] A^\ominus} & C\ddagger E & \frac{[\Delta \cup \{\mathbf{b}\}] A^\ominus}{[\Delta \cup \{\mathbf{b}_1\}] A^\ominus \quad [\Delta \cup \{\mathbf{b}_2\}] A^\ominus} \\ C\neg E & \frac{[\Delta \cup \{\neg B\}] A^\ominus}{[\Delta \cup \{*B\}] A^\ominus} \end{aligned}$$

The other rules are as follows:

- Prem If $A \in \Gamma$, introduce A^\ominus .
 Goal Introduce $[G] G^\ominus$.
 EFQ If $A \in \Gamma$, introduce $[*A] G^\ominus$.

$$\begin{aligned} \text{Trans} & \frac{[\Delta \cup \{B\}] A^\ominus \quad [\Delta'] B^{\ominus'}}{[\Delta \cup \Delta'] A^{\ominus \cup \ominus'}} & \text{EM} & \frac{[\Delta \cup \{B\}] A^\ominus \quad [\Delta' \cup \{\neg B\}] A^{\ominus'}}{[\Delta \cup \Delta'] A^{\ominus \cup \ominus'}} \\ \text{EM0} & \frac{[\Delta \cup \{\neg A\}] A^\ominus}{[\Delta] A^\ominus} & \text{IC} & \frac{[\Delta] Dab(\Lambda \cup \Lambda')^{\ominus \cup \Lambda'}}{[\Delta] Dab(\Lambda \cup \Lambda')^\ominus} \end{aligned}$$

⁴for the definition of the terms positive part and target, see below

⁵The rule to the left actually summarizes two rules: both $[\Delta] \mathbf{a}_1^\ominus$ and $[\Delta] \mathbf{a}_2^\ominus$ may be derived from $[\Delta] \mathbf{a}^\ominus$; similarly for the rule to the right and for the condition analyzing rule to the right below.

8 A Proof Procedure for Adaptive Logics

For the restrictions on applications of the rules, the positive part relation is needed. That A is a *positive part* of another formula is recursively defined by the following clauses:⁶

1. $\text{pp}(A, A)$.
2. $\text{pp}(A, \neg\neg A)$.
3. $\text{pp}(*A, \neg A)$.
4. If $\text{pp}(A, \mathbf{a}_1)$ or $\text{pp}(A, \mathbf{a}_2)$, then $\text{pp}(A, \mathbf{a})$.
5. If $\text{pp}(A, \mathbf{b}_1)$ or $\text{pp}(A, \mathbf{b}_2)$, then $\text{pp}(A, \mathbf{b})$.
6. If $\text{pp}(A, B)$ and $\text{pp}(B, C)$, then $\text{pp}(A, C)$.

Next, some line marking is needed. A-marking is the adaptive logic marking. For the current procedure no A-marks are needed yet. D-marking (marking in view of D-conditions) is governed by the following definition.

DEFINITION 3.1

Where $[\Delta] A^\Theta$ is derived at line i , line i is D-marked iff one of the following conditions is fulfilled:

1. line i is not an application of a goal rule and $A \in \Delta$,
2. for some $\Delta' \subset \Delta$ and $\Theta' \subseteq \Theta$, $[\Delta'] A^{\Theta'}$ occurs in the proof,
3. no application of EFQ occurs at a line preceding i and $B, \neg B \in \Delta$ for some B ,
4. no application of EFQ occurs at a line preceding i and, for some $B \in \Delta$, $\neg B^\emptyset$ occurs in the proof.

The members of the D-conditions of *unmarked* lines of the proof are called the *targets* of the proof.

The procedure $\text{GP}_{\text{CLuN}}(\Gamma, G)$. The above rules are applied with premise set Γ and goal G under the conditions below (just apply the first permitted rule to the first line to which this rule is applicable), until the line G^\emptyset is added to the proof (the procedure concludes that $\Gamma \vdash_{\text{CLuN}} G$) or no more lines can be added. In the last case the procedure concludes that $\Gamma \not\vdash_{\text{CLuN}} G$.

1. The proofs start by applying the goal rule.
2. Premises are introduced and formulas analyzed iff a target is a positive part of the formula of the added line.
3. Condition analyzing rules are only applied to targets.
4. A formula analyzing rule is never applied to a formula that does not have a premise in its path.
5. Once $[\Delta] A^\Theta$ occurs in the proof, one never adds another line with that same formula, D-condition and A-condition (even if the justification of the line is different).
6. Finally, EFQ is only applied if no other rules are applicable.⁷

⁶Unlike what is done in [39] and [16], I do not introduce negative parts because this complicates the predicative case. Clause 6 is only required in view of clauses 2 and 3.

⁷It can be shown that, if \neg does not occur in the premises then the premises cannot be \neg -inconsistent and hence the rule EFQ is useless. I nevertheless include it here for the sake of completeness.

Example⁸. Consider the problem $\Gamma_1 \vdash_{\text{CLuN}} s$ with $\Gamma_1 = \{\neg p \vee r, p \wedge (\neg q \supset (r \wedge t)), \neg q \vee r, \neg r \vee s\}$.

1	[s] s		Goal	\emptyset
---	-------	--	------	-------------

To start with, the only target is the goal s itself. s is a positive part of premise $\neg r \vee s$. So, this premise is introduced and analyzed in such a way that s becomes the formula element.

2	$\neg r \vee s$		Prem	\emptyset
3	[r] s	2	VE	\emptyset

r is added to the targets and r is a positive part of $\neg p \vee r$. This premise is introduced and analyzed. This makes $\neg p$ a new target. This target is not a positive part of any formula and can't be analyzed either. This is a dead end. But, r is also a positive part of $\neg q \vee r$. Analyzing this premise results in the new target q .

4	$\neg p \vee r$		Prem	\emptyset
5	[$\neg p$] r	4	VE	\emptyset
6	$\neg q \vee r$		Prem	\emptyset
7	[q] r	6	VE	\emptyset

q is a positive part of $p \wedge (\neg q \supset (r \wedge t))$ (this may be clarifying: $\text{pp}(\neg q \supset (r \wedge t), p \wedge (\neg q \supset (r \wedge t)))$, $\text{pp}(\neg q, \neg q \supset (r \wedge t))$, and finally $\text{pp}(q, \neg q)$). Therefore the premise $p \wedge (\neg q \supset (r \wedge t))$ is introduced and analyzed (lines 8 to 12). The target q is now the formula element of a line. This enables the application of the transitivity rule. Now the negation of r is in the condition for r and so $\neg r$ can be omitted, resulting in line 14. The target r is now in the formula element of an unconditional line. Hence, the goal s is derived after one more application of Trans.

8	$p \wedge (\neg q \supset (r \wedge t))$		Prem	\emptyset
9	$\neg q \supset (r \wedge t)$	2	\wedge E	\emptyset
10	[$\neg(r \wedge t)$] $\neg q$	9	\supset E	\emptyset
11	[$\neg r$] $\neg q$	10	C \neg \wedge E	\emptyset
12	[$\neg r$] q	11	\neg \rightarrow E	\emptyset
13	[$\neg r$] r	12, 7	Trans	\emptyset
14	r	13	EM0	\emptyset
15	s	14, 3	Trans	\emptyset

In this example proof, no marks were necessary.

4 Two (partial) proof procedures for conditional derivability

A proof procedure for conditional derivability is a procedure that, given a premise set Γ , a candidate conclusion G and a set Υ of finite sets of abnormalities, returns (if possible) a proof from Γ with G^Δ derived on the last line such that for every $\Delta' \subseteq \Delta$, $\Delta' \notin \Upsilon$.

⁸In the examples I follow a more efficient heuristics than simply applying the first permitted rule to the first line. This only done to save space.

DEFINITION 4.1

$\text{MinCon}_{\mathbf{AL}}(\Gamma, A)$ is the set of all sets of abnormalities Δ such that $\Gamma \vdash_{\mathbf{LLL}} A \check{\vee} Dab(\Delta)$ and if $\Delta \in \text{MinCon}_{\mathbf{AL}}(\Gamma, A)$ then there is no Δ' such that $\Delta \subset \Delta'$ and $\Delta' \in \text{MinCon}_{\mathbf{AL}}(\Gamma, A)$.

If one runs a procedure like this again after every positive answer, starting of with an empty Υ and adding the resulting Δ at the end of every procedure to Υ until the procedure stops with a positive answer, one will obtain a superset of $\text{MinCon}_{\mathbf{AL}}(\Gamma, A)$ (if this set is finite).

4.1 *A brute force proof procedure*

There is an evident way to generate proofs for conditional derivability if one has a proof procedure for the lower limit logic and a method to select a finite set Ω' of abnormalities relevant to the problem. Abnormalities are called *relevant* to a problem $\Gamma \vdash_{\mathbf{AL}} G$ iff they occur in $\text{MinCon}_{\mathbf{AL}}(\Gamma, G)$. If this set is finite, the set $\Theta = \mathcal{P}(\Omega')$ is finite as well. The set $\text{MinCon}_{\mathbf{AL}}(\Gamma, A)$ is a subset of this set Θ . Suppose this Ω' can be determined before starting the procedure. Note that this supposition is often true. Relevant abnormalities are in a lot of cases only constructed with primitive formulas that do occur in the premises or the candidate conclusion. If there are only finitely many premises, there are only finitely many subformulas of premises and candidate conclusions. Hence, there are only finitely many relevant abnormalities.

The procedure $\text{EPC}_{\mathbf{AL}}(\Gamma, G, \Upsilon)$. Let $\Omega' = R(\Gamma, G)$ denote the finite set of all relevant abnormalities for $\Gamma \vdash_{\mathbf{AL}} G$ and let Θ be the finite set $\mathcal{P}(\Omega')$. For every $\Delta \in \Theta - \{\Delta' \mid \text{there is a } \Delta'' \in \Upsilon, \text{ such that } \Delta'' \subset \Delta'\}$, run the proof procedure for $\Gamma \vdash_{\mathbf{CLuN}} G \vee Dab(\Delta)$, until a proof is found. If a proof is found for a certain Δ , add the line G^Δ to the proof and stop the procedure. If all $\Delta \in \Theta$ are finished and no proof is found, the procedure returns no proof and a negative answer.

4.2 *A goal directed proof procedure for ACLuN*

A finite set Ω' is not always findable in any logic, for any Γ and any G . For propositional \mathbf{CLuN} , for example, a set Θ is easily constructible whenever Γ is finite, but when Γ is infinite, it is possible that there is no such finite Θ . Moreover, the above procedure is a brute force method and is therefore terribly inefficient. In this subsection I present a more efficient procedure for the \mathbf{ACLuN} , that does not presuppose a set of relevant abnormalities (or in other words: it constructs such a set within the process in an intelligent way).

The procedure for conditional derivability for \mathbf{ACLuN} generates goal directed proofs and is defined from a set of rules and a recursive positive part function. All the rules from $\text{GP}_{\mathbf{CLuN}}$ are valid here, but two more rules are necessary. A formula analyzing rule and a condition analyzing rule:

$$\begin{array}{l} \neg\text{-E} \quad \frac{[\Delta] \neg A^\Theta}{[\Delta] * A^{\Theta \cup \{A \wedge \neg A\}}} \\ \text{C}\check{\neg}\text{-E} \quad \frac{[\Delta \cup \{\check{\neg} B\}] A^\Theta}{[\Delta \cup \{B\}] A^{\Theta \cup \{B \wedge \neg B\}}} \end{array}$$

A-marking (marking in view of the A-conditions) is not yet relevant for conditional derivability. D-marking (marking in view of D-conditions) is governed by the following definition.

DEFINITION 4.2

Where $[\Delta] A^\Theta$ is derived at line i , line i is D-marked iff one of the following conditions is fulfilled:

1. line i is not an application of a goal rule and $A \in \Delta$,
2. for some $\Delta' \subset \Delta$ and $\Theta' \subseteq \Theta$, $[\Delta'] A^{\Theta'}$ occurs in the proof,
3. no application of EFQ occurs at a line preceding i and $B, \neg B \in \Delta$ for some B ,
4. no application of EFQ occurs at a line preceding i and, for some $B \in \Delta$, $\neg B^\emptyset$ occurs in the proof.

The members of the D-conditions of *unmarked* lines of the proof are called the *targets* of the proof.

The procedure $\text{GPC}_{\text{ACLuN}}(\Gamma, G, \Upsilon)$. The rules from the procedure GP_{CLuN} together with the new rules $\neg\text{E}$ and $\text{C}\neg\text{E}$ are applied under the following conditions:

1. The proofs start by applying the goal rule.
2. No rules are applied that result in a line that has a formula element G and an A-condition Δ , such that it is a superset of some element in Υ .
3. Premises are introduced and formulas analyzed iff a target is a positive part of the formula of the added line.
4. Condition analyzing rules are only applied to targets.
5. A formula analyzing rule is never applied to a formula that does not have a premise in its path.
6. Once $[\Delta] A^\Theta$ occurs in the proof, one never adds another line with that same formula, D-condition and A-condition (even if the justification of the line is different).
7. Finally, EFQ is only applied if no other rules are possible anymore.⁹

Example. For an example, see the example for the proof procedure for final derivability. This procedure makes extensively use of the conditional derivability procedure.

5 (Partial) proof procedures for final derivability

The two proof procedures for final derivability (one for the Minimal Abnormality logics and one for the Reliability logics) make use of the proof procedure for the lower limit logic and the one for conditional derivability. They do not generate new proofs, but rather combine the proofs that result from different applications of the other two proof procedures, add markings, and (in the case of Minimal Abnormality) add some lines. These lines are mere combinations of a number of lines that were the conclusions of applications of the other procedures. The proof format is obviously the same as in the aforementioned procedures. For adaptive logics that only differ with

⁹It can be shown that, if \neg does not occur in the premises, then the premises cannot be \neg -inconsistent and hence phase 1B is useless. I nevertheless include it here for the sake of completeness.

12 A Proof Procedure for Adaptive Logics

respect to the strategy, the same lower limit logic- and conditional derivability proof procedure is used. The procedure I will give is universal; it works for all adaptive logics in standard format with a classically behaving conjunction and disjunction.

In contrast to the aforementioned procedures, this procedure will A-mark lines. Remark that these A-marks are not the same as the marks generated by the adaptive logics marking definition. Nevertheless, they are very similar to the adaptive logic marks, and they have the same function. To generate adaptive proofs from the goal directed proofs generated by this procedure, replace every line

$$i \quad [\Delta]A \quad j, k \quad R \quad \Theta$$

by a line

$$i \quad \neg \bigwedge \Delta \vee A \quad j, k \quad R' \quad \Theta$$

where $R' = R$ if $R = Prem$ and $R' = RC$ if $R = C\check{\neg}E$ or $R = \neg E$ (in case of proofs for **ALuN**) or if R is a similar rule for other logics, otherwise $R' = RU$. The classical negation (\neg) is added to a logic to become a possible lower limit logic for an adaptive logic, but does not need to occur in the premises or the conclusion (see subsection 2.1). Remove all A-marks and D-marks and apply the adaptive logic's marking definition to add the right adaptive logic marks.

5.1 Minimal Abnormality

Phase 1.

(1.1) *Subphase 1A.* To start, let $\Upsilon = \emptyset$.

(1.2) Run the conditional derivability procedure with premise set Γ , candidate conclusion G and set of conditions Υ . Let Θ be the A-condition of the last line of the resulting proof (if there is any). Let i be the line number of this line. There are three possibilities:

- If $\Theta = \emptyset$, then G^\emptyset is derived. The procedure stops and $\Gamma \vdash_{\mathbf{AL}^m} G$.
- If $\Theta \neq \emptyset$, add Θ to Υ . Add the line

$$j \quad G \vee \bigwedge \{Dab(\Delta) \mid \Delta \in \Upsilon\} \quad i \quad RU \quad \emptyset$$

to the proof. the procedure moves to phase 2 (go to (2.1)) and later returns to phase 1. There are two possibilities:

- line j is not A-marked. The procedure stops and $\Gamma \vdash_{\mathbf{AL}^m} G$.
- line j is A-marked. Go on, back to (1.2).
- The conditional derivability procedure did not return a result: the procedure terminates and G^Θ is not derived at an unmarked line for any Θ : move to subphase 1B (go to (1.3)).

(1.3) *Subphase 1B.* Aim: to derive G^\emptyset by applications of EFQ as well as well of the other **CLuN**-rules.

Phase 2.

- (2.1) $G \vee \bigwedge\{Dab(\Delta)|\Delta \in \Upsilon\}$ was derived in phase 1, say at line j . To start, let $\Upsilon' = \emptyset$.
- (2.2) Run the conditional derivability procedure with premise set Γ , candidate conclusion $\bigwedge\{Dab(\Delta)|\Delta \in \Upsilon\}$ and set of conditions Υ' . Let Λ be the A-condition of the last line of the resulting proof, if there is any. Let i be the line number of this line. There are three possibilities:
- If $\Lambda = \emptyset$, then $\bigwedge\{Dab(\Delta)|\Delta \in \Upsilon\}^\emptyset$ is derived. Line j is A-marked, the procedure returns to phase 1.
 - If $\Lambda \neq \emptyset$, then $\bigwedge\{Dab(\Delta)|\Delta \in \Upsilon\}^\Lambda$ is derived, say at line k . Add Λ to Υ' . The procedure moves to phase 3 and later returns to phase 2. There are two possibilities:
 - line k is not A-marked: line j is A-marked. The procedure returns to phase 1.
 - line k is A-marked: go on, back to (2.2).
 - The conditional derivability procedure did not return a result: phase 2 terminates and $\bigwedge\{Dab(\Delta)|\Delta \in \Upsilon\}^\Lambda$ is not derived at an unmarked line for any Λ : line j is not A-marked and the procedure returns to phase 1.

Phase 3.

- (3.1) $G \vee \bigwedge\{Dab(\Delta)|\Delta \in \Upsilon\}$ was derived in phase 1, say at line j , and $\bigwedge\{Dab(\Delta)|\Delta \in \Upsilon\}^\Lambda$ was derived in phase 2 for some Λ , say at line k . Phase 3 starts by applying the **LLL**-proof procedure with premise set Γ and candidate conclusion $Dab(\Lambda)$. Either the procedure returns a proof for $Dab(\Lambda)$ or it returns nothing:
- $Dab(\Lambda)$ is derived. Line k is A-marked, the procedure returns to phase 2.
 - $Dab(\Lambda)^\emptyset$ is not derived: line k is not A-marked. The procedure returns to phase 2.

Example¹⁰. Consider the problem $\Gamma_2 \vdash_{\mathbf{CLuN}^m} s$ with $\Gamma_2 = !r, s \vee !p \vee !q, s \vee !p \vee !r, s \vee !q \vee !r$.

1.1 Phase 1. The procedure for conditional derivability $\text{GPC}_{\mathbf{ACLuN}}(\Gamma_2, G, \Upsilon)$ is started with goal $G = s$ and $\Upsilon = \emptyset$. This results in a positive answer and the following proof:

1	[s] s		Goal	∅
2	$s \vee (p \wedge \neg p) \vee (q \wedge \neg q)$		Prem	∅
3	$[\neg((p \wedge \neg p) \vee (q \wedge \neg q))]s$	2	∨E	∅
4	$[\neg(p \wedge \neg p), \neg(q \wedge \neg q)]s$	3	C∨E	∅
5	$[\neg p, \neg q]s$	4	C∧E	∅
6	$[\neg p, \neg\neg q]s$	4	C∧E	∅
7	$[\neg\neg p, \neg q]s$	4	C∧E	∅
8	$[\neg\neg p, \neg\neg q]s$	4	C∧E	∅
9	$[p, \neg q]s$	7	C∧E	{!p}
10	$[\neg q]s$	5,9	EM	{!p}
11	$[\neg p, q]s$	6	C∧E	{!q}
12	$[\neg p]s$	10,11	EM	{!p, !q}
13	$[p, q]s$	8	C∧E	{!p, !q}

¹⁰In the examples in this section, evident lines of the proofs generated by the procedures are omitted. The line after the omitted block gets no justification and line number $n + 1$, where n is the line number of the line before the block.

14 *A Proof Procedure for Adaptive Logics*

14	$[q]s$	11,13	EM	$\{!p, !q\}$
15	s	10,14	EM	$\{!p, !q\}$

The following line is added:

16	$s \vee !p \vee !q$			\emptyset
----	---------------------	--	--	-------------

- 1.2 Phase 2. The procedure for conditional derivability $\text{GPC}_{\mathbf{ACLuN}}(\Gamma_2, G, \Upsilon)$ is started with goal $G = !p \vee !q$ and $\Upsilon = \emptyset$. This results in a positive answer and the following proof:

17	$[!p \vee !q] !p \vee !q$	Goal	\emptyset
18	$!p \vee !q \vee (r \wedge \neg r)$	Prem	\emptyset
19	$[\neg(r \wedge \neg r)] !p \vee !q$	18	$\vee E$ \emptyset
20	$[\neg r] !p \vee !q$	19	$C \neg \wedge E$ \emptyset
21	$[\neg \neg r] !p \vee !q$	19	$C \neg \wedge E$ \emptyset
22	$[r] !p \vee !q$	21	$C \neg \neg E$ $\{!r\}$
23	$!p \vee !q$	20,22	EM $\{!r\}$

- 1.3 Phase 3. The procedure for \mathbf{CLuN} -derivability $\text{GP}_{\mathbf{CLuN}}(\Gamma_2, G)$ is started with goal $G = !r$. This results in a negative answer. Line 16 is A-marked.

- 2.1 Phase 1. The procedure for conditional derivability $\text{GPC}_{\mathbf{ACLuN}}(\Gamma_2, G, \Upsilon)$ is started with goal $G = s$ and $\Upsilon = \{\{!p, !q\}\}$. This results in a positive answer and a proof with the following last line:

24	s	EM	$\{!p, !r\}$
----	-----	----	--------------

The following line is added:

25	$s \vee (!p \vee !q) \wedge (!p \vee !r)$		\emptyset
----	---	--	-------------

- 2.2 Phase 2. The procedure for conditional derivability $\text{GPC}_{\mathbf{ACLuN}}(\Gamma_2, G, \Upsilon)$ is started with goal $G = (!p \vee !q) \wedge (!p \vee !r)$ and $\Upsilon = \emptyset$. This results in a positive answer and a proof with following last line:

26	$(!p \vee !q) \wedge (!p \vee !r)$	2	$\vee E$ $\{!q, !r\}$
----	------------------------------------	---	-----------------------

- 2.3 Phase 3. The procedure for \mathbf{CLuN} -derivability $\text{GP}_{\mathbf{CLuN}}(\Gamma_2, G)$ is started with goal $G = !q \vee !r$. This results in a negative answer. Line 25 is A-marked.

- 3.1 Phase 1. The procedure for conditional derivability $\text{GPC}_{\mathbf{ACLuN}}(\Gamma_2, G, \Upsilon)$ is started with goal $G = s$ and $\Upsilon = \{\{!p, !q\}, \{!p, !r\}\}$. This results in a positive answer and a proof with the following last line:

27	s		$\{!q, !r\}$
----	-----	--	--------------

The following line is added:

28	$s \vee (!p \vee !q) \wedge (!p \vee !r) \wedge (!q \vee !r)$		\emptyset
----	---	--	-------------

- 3.2 Phase 2. The procedure for conditional derivability $\text{GPC}_{\mathbf{ACLuN}}(\Gamma_2, G, \Upsilon)$ is started with goal $G = (!p \vee !q) \wedge (!p \vee !r) \wedge (!q \vee !r)$ and $\Upsilon = \emptyset$. This results in a positive answer and a proof with following last line:

29	$(!p \vee !q) \wedge (!p \vee !r) \wedge (!q \vee !r)$		$\{!p, !q, !r\}$
----	--	--	------------------

3.3 Phase 3. The procedure for **CLuN**-derivability $\text{GP}_{\text{CLuN}}(\Gamma_2, G)$ is started with goal $G = !p \vee !q \vee !r$. This results in a positive answer and a proof with following last line:

30 $!p \vee !q \vee !r$ \emptyset

Line 29 is A-marked.

3.2b Phase 2. The procedure for conditional derivability $\text{GPC}_{\text{ACLuN}}(\Gamma_2, G, \Upsilon)$ is started with goal $G = (!p \vee !q) \wedge (!p \vee !r) \wedge (!q \vee !r)$ and $\Upsilon = \{\{!p, !q, !r\}\}$. This results in a negative answer. Phase 2 terminates and line 28 is not A-marked. $\Gamma_2 \vdash_{\text{CLuN}^m} s$.

5.2 Reliability

Phase 1.

- (1.1) *Subphase 1A.* To start, let $\Upsilon = \emptyset$.
- (1.2) Run the conditional derivability procedure with premise set Γ , candidate conclusion G and set of conditions Υ . Let Θ be the A-condition of the last line of the resulting proof (if there is any). Let i be the line number of this line. There are three possibilities:
- If $\Theta = \emptyset$, then G^\emptyset is derived. The procedure stops and $\Gamma \vdash_{\text{AL}^r} G$.
 - If $\Theta \neq \emptyset$, add Θ to Υ . The procedure moves to phase 2 (go to (2.1)) and later returns to phase 1. There are two possibilities:
 - line j is not A-marked. The procedure stops and $\Gamma \vdash_{\text{AL}^r} G$.
 - line j is A-marked. Go on, back to (1.2).
 - The conditional derivability procedure did not return a result: the procedure terminates and G^Θ is not derived at an unmarked line for any Θ : move to subphase 1B (go to (1.3)).
- (1.3) *Subphase 1B.* Aim: to derive G^\emptyset by applications of EFQ as well as well of the other **LLL**-rules.

Phase 2.

- (2.1) G^Θ was derived in phase 1, say at line j . To start, let $\Upsilon' = \emptyset$. Repeat the following instructions.
- (2.2) Run the conditional derivability procedure with premise set Γ , candidate conclusion $Dab(\Theta)$ and set of conditions Υ' . Let Λ be the A-condition of the last line of the resulting proof, if there is any. Let i be the line number of this line. There are three possibilities:
- If $\Lambda = \emptyset$, then $Dab(\Theta)^\emptyset$ is derived. Line j is A-marked, the procedure returns to phase 1.
 - If $\Lambda \neq \emptyset$, then $Dab(\Theta)^\Lambda$ is derived, say at line k . Add Λ to Υ' . The procedure moves to phase 3 and later returns to phase 2. There are two possibilities:
 - line k is not A-marked: line j is A-marked. The procedure returns to phase 1.
 - line k is A-marked: go on, back to (2.2).
 - The conditional derivability procedure did not return a result: phase 2 terminates and $Dab(\Theta)^\Lambda$ is not derived at an unmarked line for any Λ : line j is

not A-marked and the procedure returns to phase 1.

Phase 3.

(3.1) G^Θ was derived in phase 1, say at line j , and $Dab(\Theta)^\Lambda$ was derived in phase 2 for some Λ , say at line k . Phase 3 starts by applying the **LLL**-proof procedure with premise set Γ and candidate conclusion $Dab(\Lambda)$. Either the procedure returns a proof for $Dab(\Lambda)$ or it returns nothing:

- $Dab(\Lambda)$ is derived. Line k is A-marked, the procedure returns to phase 2.
- $Dab(\Lambda)^\theta$ is not derived: line k is not A-marked. The procedure returns to phase 2.

Example. Consider the problem $\Gamma_3 \vdash_{\mathbf{CLuN}^r} s$ with $\{\Gamma_3 = !p \vee !q \vee !r, s \vee !p \vee !q, s \vee !p \vee !r, s \vee !q \vee !r, t, s \vee \neg t\}$.

1.1 Phase 1. The procedure for conditional derivability $\text{GPC}_{\mathbf{ACLuN}}(\Gamma_3, G, \Upsilon)$ is started with goal $G = s$ and $\Upsilon = \emptyset$. This results in a positive answer and a proof with the following last line:

$$1 \quad s \quad \{!p, !q\}$$

1.2 Phase 2. The procedure for conditional derivability $\text{GPC}_{\mathbf{ACLuN}}(\Gamma_3, G, \Upsilon)$ is started with goal $G = !p \vee !q$ and $\Upsilon = \emptyset$. This results in a positive answer and a proof with the following last line:

$$2 \quad !p \vee !q \quad \{!r\}$$

1.3 Phase 3. The procedure for **CLuN**-derivability $\text{GP}_{\mathbf{CLuN}}(\Gamma_3, G)$ is started with goal $G = !r$. This results in a negative answer. Line 1 is A-marked.

2.1 Phase 1. The procedure for conditional derivability $\text{GPC}_{\mathbf{ACLuN}}(\Gamma_3, G, \Upsilon)$ is started with goal $G = s$ and $\Upsilon = \{\{!p, !q\}\}$. This results in a positive answer and a proof with the following last line:

$$3 \quad s \quad \{!p, !r\}$$

2.2 Phase 2. The procedure for conditional derivability $\text{GPC}_{\mathbf{ACLuN}}(\Gamma_3, G, \Upsilon)$ is started with goal $G = !p \vee !r$ and $\Upsilon = \emptyset$. This results in a positive answer and a proof with following last line:

$$4 \quad !p \vee !r \quad \{!q\}$$

2.3 Phase 3. The procedure for **CLuN**-derivability $\text{GP}_{\mathbf{CLuN}}(\Gamma_3, G)$ is started with goal $G = !q$. This results in a negative answer. Line 3 is A-marked.

3.1 Phase 1. The procedure for conditional derivability $\text{GPC}_{\mathbf{ACLuN}}(\Gamma_3, G, \Upsilon)$ is started with goal $G = s$ and $\Upsilon = \{\{!p, !q\}, \{!p, !r\}\}$. This results in a positive answer and a proof with the following last line:

$$5 \quad s \quad \{!q, !r\}$$

3.2 Phase 2. The procedure for conditional derivability $\text{GPC}_{\mathbf{ACLuN}}(\Gamma_3, G, \Upsilon)$ is started with goal $G = !q \vee !r$ and $\Upsilon = \emptyset$. This results in a positive answer and a proof with following last line:

$$6 \quad !q \vee !r \quad \{!p\}$$

THEOREM 6.3

If Γ is finite, the procedure $\text{GPC}_{\mathbf{ACLUuN}}(\Gamma, G, \Upsilon)$ terminates. If this procedure terminates, G^Θ is derived on the last line of the generated proof iff $\Delta \notin \Upsilon$, for any $\Delta \subseteq \Theta$, and $\Gamma \vdash_{\mathbf{CLuN}} G \vee \bigvee \Theta$

PROOF. The procedure for conditional derivability in this paper is equivalent to the application of the rules in [10]. I use a set Υ of already derived conditions. This is not present in [10]. Batens did not care for the construction of a conditional derivability procedure and in his paper all necessary conditions are derived within one final derivability proof. The D-marking of lines with a condition that is a superset of the condition of a line with the same formula has the same function as my set Υ . In view of this remark, the following theorem for Batens' final derivability procedure is equivalent to my theorem 6.3.

Theorem 7 from [10]. If $\Gamma \vdash_{\mathbf{CLuN}} G \vee (A_1 \wedge \neg A_1) \vee \dots \vee (A_n \wedge \neg A_n)$ and $\Gamma \not\vdash_{\mathbf{CLuN}} \Delta$ for every $\Delta \subset \{G, A_1 \wedge \neg A_1, \dots, A_n \wedge \neg A_n\}$, then, $G^{\{A_1 \wedge \neg A_1, \dots, A_n \wedge \neg A_n\}}$ is derivable in every prospective proof for $\Gamma \vdash_{\mathbf{ACLUuN1}} G$.

The proof that the final derivability procedure is correct requires more metatheory. I introduce some new definitions and lemmas on choice sets (the crucial mathematical object involved in the marking definition for Minimal Abnormality). Let $\Pi(\Upsilon)$ denote the set of choice sets of a set of sets of formulas Υ and $\Pi_m(\Upsilon) = \{\pi \mid \pi \in \Pi(\Upsilon)\}$; there is no $\pi' \in \Pi(\Upsilon)$ such that $\pi' \subset \pi$. Let $C_\Gamma(A)$ denote the set of sets of abnormalities, such that $\Delta \in C_\Gamma(A)$ iff $\Gamma \vdash_{\mathbf{LLL}} A \vee \text{Dab}(\Delta)$

LEMMA 6.4

$\bigwedge \{\bigvee A \mid A \in \Upsilon\} \dashv\vdash_{\mathbf{LLL}} \bigvee \{\bigwedge A \mid A \in \Pi(\Upsilon)\} \dashv\vdash_{\mathbf{LLL}} \bigvee \{\bigwedge A \mid A \in \Pi_m(\Upsilon)\}$ and
 $\bigvee \{\bigwedge A \mid A \in \Upsilon\} \dashv\vdash_{\mathbf{LLL}} \bigwedge \{\bigvee A \mid A \in \Pi(\Upsilon)\} \dashv\vdash_{\mathbf{LLL}} \bigwedge \{\bigvee A \mid A \in \Pi_m(\Upsilon)\}$.

PROOF. The proofs (using the truth tables for \vee and \wedge) are easy but a little bit long winding. ■

I first show that whenever the procedure for final derivability stops with a positive answer, the candidate conclusion is indeed a correct conclusion from the premises. The basic idea behind the procedure is that a consequence G is Minimal Abnormality finally derivable iff the disjunction of G and some conjunction of *Dab*-formulas A is **LLL**-derivable, and the disjunction of A and a *Dab*-formula B is only **LLL**-derivable if B is **LLL**-derivable. Remark that this fact does not always hold when the conclusion is **LLL**-derivable on infinitely many conditions, because infinite conjunctions are not allowed in **LLL**.

LEMMA 6.5

When the final derivability procedure for Minimal Abnormality ends phase 2 without A-marking G , $\Gamma \vdash_{\mathbf{AL}^m} G$

PROOF. I will derive an inconsistency from the hypothesis that the procedure ends phase 2 without A-marking G in phase 2 and $\Gamma \not\vdash_{\mathbf{AL}^m} G$. Remark that the set of

minimal conditions on which G is derivable is finite, otherwise phase 2 would not have terminated. Formally this comes down to:

$$\begin{aligned} &\text{There is a } \Theta \subseteq C_\Gamma(G) \text{ such that for all } \Delta \subset \Omega: \\ &\text{if } \Gamma \vdash_{\mathbf{LLL}} \bigwedge \{ \bigvee \theta \mid \theta \in \Theta \} \vee Dab(\Delta), \text{ then } \Gamma \vdash_{\mathbf{LLL}} Dab(\Delta). \end{aligned} \quad (6.1)$$

AND (this is obtained by combining definition 2.2 and the first definition of final derivability)

$$\text{For some } \phi \in \Phi(\Gamma), \text{ there is no } \theta \in C_\Gamma(G) \text{ for which } \phi \cap \theta = \emptyset. \quad (6.2)$$

6.2 entails:

$$\text{For some } \phi \in \Phi(\Gamma), \text{ for all } \theta \in C_\Gamma(G): \phi \cap \theta \neq \emptyset. \quad (6.3)$$

From 6.3 we know that at least one ϕ must be a superset of at least one choice set of $C_\Gamma(G)$, or:

$$\text{For some } \phi \in \Phi(\Gamma) \text{ and some } \pi \in \Pi(C_\Gamma(G)): \pi \subseteq \phi. \quad (6.4)$$

Because of the fact that in view of lemma 6.4 $\bigvee \{ \bigwedge \phi' \mid \phi' \in \Phi(\Gamma) \}$ is **LLL**-equivalent to $\bigwedge \{ Dab(\Delta_1), \dots, Dab(\Delta_n) \}$, where $Dab(\Delta_1), \dots, Dab(\Delta_n)$ are all the minimal *Dab*-formulas derivable from Γ , $\Gamma \vdash_{\mathbf{LLL}} \bigwedge \phi \vee \bigvee \{ \bigwedge \phi' \mid \phi' \in \Phi(\Gamma) - \phi \}$. And hence, because, in view of 6.4, $\bigwedge \pi$ is a consequence of $\bigwedge \phi$, the following holds¹¹:

$$\begin{aligned} &\text{For some } \phi \in \Phi(\Gamma) \text{ and some } \pi \in \Pi(C_\Gamma(G)): \\ &\Gamma \vdash_{\mathbf{LLL}} \bigwedge \pi \vee \bigvee \{ \bigwedge \phi' \mid \phi' \in \Phi(\Gamma) - \phi \}. \end{aligned} \quad (6.5)$$

Hence:

$$\begin{aligned} &\text{For some } \phi \in \Phi(\Gamma): \\ &\Gamma \vdash_{\mathbf{LLL}} \bigvee \{ \bigwedge \pi \mid \pi \in \Pi(C_\Gamma(G)) \} \vee \bigvee \{ \bigwedge \phi' \mid \phi' \in \Phi(\Gamma) - \phi \} \end{aligned} \quad (6.6)$$

and with lemma 6.4:

$$\begin{aligned} &\text{For some } \phi \in \Phi(\Gamma): \\ &\Gamma \vdash_{\mathbf{LLL}} \bigwedge \{ \bigvee \theta \mid \theta \in C_\Gamma(G) \} \vee \bigwedge \{ \bigvee \phi' \mid \phi' \in \Pi(\Phi(\Gamma) - \phi) \}. \end{aligned} \quad (6.7)$$

Since ϕ is a minimal choice set of $\{ \Delta_1, \dots, \Delta_n \}$ and $\Delta_1, \dots, \Delta_n$ are minimal *Dab*-consequences of Γ , there is a choice set Δ of $\Phi(\Gamma) - \phi$, such that $\Gamma \not\vdash_{\mathbf{LLL}} Dab(\Delta)$, 6.7 leads to:

$$\begin{aligned} &\text{For some } \phi \in \Phi(\Gamma), \text{ and some } \Delta \in \Pi(\Phi(\Gamma) - \phi): \\ &\Gamma \vdash_{\mathbf{LLL}} \bigwedge \{ \bigvee \theta \mid \theta \in C_\Gamma(G) \} \vee Dab(\Delta) \text{ and } \Gamma \not\vdash_{\mathbf{LLL}} Dab(\Delta). \end{aligned} \quad (6.8)$$

¹¹If the set of all minimal *Dab*-formulas derivable from Γ is infinite, the conjunction of the members of this set is of course also infinite and thus not a well formed formula. However, the case that is investigated here, is the case that the procedure has terminated. One can easily proof that the procedure would not have terminated if the mentioned set would have been infinite.

But then,

For some finite $\Delta \subset \Omega$:

$$\Gamma \vdash_{\mathbf{LLL}} \bigwedge \{ \forall \theta | \theta \in C_{\Gamma}(G) \} \vee Dab(\Delta) \text{ and } \Gamma \not\vdash_{\mathbf{LLL}} Dab(\Delta). \quad (6.9)$$

If, for some formula A , $\Gamma \vdash_{\mathbf{LLL}} \bigwedge \{ \forall \theta | \theta \in C_{\Gamma}(G) \} \vee A$ and $\Gamma \not\vdash_{\mathbf{LLL}} A$ then also for all $\Theta \subseteq C_{\Gamma}(G)$ $\Gamma \vdash_{\mathbf{LLL}} \bigwedge \{ \forall \theta | \theta \in \Theta \} \vee A$ and $\Gamma \not\vdash_{\mathbf{LLL}} A$

For all $\Theta \subseteq C_{\Gamma}(G)$, there is a $\Delta \subset \Omega$:

$$\Gamma \vdash_{\mathbf{LLL}} \bigwedge \{ \forall \theta | \theta \in \Theta \} \vee Dab(\Delta) \text{ and } \Gamma \not\vdash_{\mathbf{LLL}} Dab(\Delta). \quad (6.10)$$

This is clearly in contradiction with hypothesis 6.1. ■

The negative part of theorem 6.7 still needs to be proven: if the final derivability procedure stops with a negative answer, the candidate conclusion is not a conclusion of the premises.

LEMMA 6.6

When G is A-marked after phase 1 in the final derivability procedure for Minimal Abnormality, $\Gamma \not\vdash_{\mathbf{AL}^m} G$

PROOF. Let $\Upsilon = \langle \Delta_1, \dots, \Delta_n \rangle$ be the finite set of all minimal conditions on which G is derived. $\Gamma \vdash_{\mathbf{LLL}} G \vee \bigwedge \{ Dab(\Delta) | \Delta \in \Upsilon \}$ and $\Gamma \vdash_{\mathbf{LLL}} \bigwedge \{ Dab(\Delta) | \Delta \in \Upsilon \}$ or $(\Gamma \vdash_{\mathbf{LLL}} \bigwedge \{ Dab(\Delta) | \Delta \in \Upsilon \} \vee Dab(\Lambda))$ and $\Gamma \not\vdash_{\mathbf{LLL}} Dab(\Lambda)$ for some $\Lambda \subset \Omega$.

In the first case ($\Gamma \vdash_{\mathbf{LLL}} \bigwedge \{ Dab(\Delta) | \Delta \in \Upsilon \}$), every conditional derivation of Γ , say on condition Θ , can be marked in view of the derivable formula $Dab(\Theta)$, because Θ must be a superset of a set in Υ and $\Gamma \vdash_{\mathbf{LLL}} Dab(\Delta)$ for any $\Delta \in \Upsilon$. In the second case, there is a series of minimal Dab -formulas \mathbf{LLL} -derivable from Γ : $\langle \Delta'_1, \dots, \Delta'_n \rangle$ with $\Delta'_i \subseteq \Delta_i \cup \Lambda$ and $\Delta'_i \not\subseteq \Lambda$ (otherwise $\Gamma \vdash_{\mathbf{LLL}} Dab(\Lambda)$, which was not the case) for all $i \leq n$. So there is a $\phi \in \Phi(\Gamma)$ such that $\{A_1, \dots, A_n\} \subseteq \phi$ and $A_i \in \Delta_i$ for all $i \leq n$. This set ϕ will have an element in common with every condition in Υ . All lines on which G can be derived, will be marked. ■

THEOREM 6.7

If the procedure $\text{PAm}(\Gamma, G)$ procedure terminates, for some $\Theta_1 \subset \Omega, \Theta_2 \subset \Omega, \dots$, and $\Theta_n \subset \Omega$, $G \vee \bigwedge (\bigvee \Theta_1, \bigvee \Theta_2, \dots, \bigvee \Theta_n)$ is derived on a line in the generated proof that is not A-marked iff $\Gamma \vdash_{\mathbf{AL}^m} G$.

PROOF. Immediate in view of the two preceding lemmas. ■

THEOREM 6.8

If the procedure $\text{PAr}(\Gamma, G)$ terminates, for some $\Theta \subset \Omega$, G^{Θ} is derived on a line in the generated proof that is not A-marked iff $\Gamma \vdash_{\mathbf{AL}^r} G$.

PROOF. The procedure results essentially in the same proof as the one in [10]. So theorem 6.10 is correct in view of the following theorems from [10].

Theorem 4 from [10]. If Γ is finite, every prospective proof for $\Gamma \vdash_{\mathbf{ACL}_{\mathbf{uN1}}} G$ terminates.

Theorem 8 from [10]. For all *finite* Γ and for all G , the procedure forms a decision method for $\Gamma \vdash_{\mathbf{ACLuN1}} G$.

■

THEOREM 6.9

If total procedures are available for respectively lower limit logic and conditional derivability of an adaptive logic \mathbf{AL} , and $\text{MinCon}_{\mathbf{AL}}(\Gamma, G)$ is finite, then the procedures $\text{PAr}(\Gamma, G)$ and $\text{PAm}(\Gamma, G)$ that use these procedures are total.

THEOREM 6.10

If Γ is finite, then the procedures $\text{PAr}(\Gamma, G)$ and $\text{PAm}(\Gamma, G)$ for the positive fragments of respectively \mathbf{CLuN}^r and \mathbf{CLuN}^m are total.

7 Conclusion

In this paper I have presented a proof procedure for the actual adaptive consequence relation: final derivability. I have done this for \mathbf{CLuN}^m and \mathbf{CLuN}^r , and thanks to a modular approach, these results are immediately generalizable to all adaptive logics with a lower limit logic with a classically behaving disjunction and conjunction and for which a proof procedure for conditional derivability is devisable.

The procedure for Minimal Abnormality puts the Minimal Abnormality strategy in a different light. The strategy was often seen as much more complicated than the Reliability strategy. To people who are not very familiar with adaptive logics, the marking definition seemed rather abstract. The procedure I have presented shows that there is not such a big difference between the two strategies, and that the resulting Minimal Abnormality proofs are not more complicated or unrealistic than the Reliability proofs. The Minimal Abnormality strategy is an elegant formal tool, that gives in some contexts more accurate results than the marking definition for Reliability.

The proofs generated by the procedures have a realistic character. Especially in two aspects. First, the goal directed proof method defines a realistic problem solving method (already for classical logic). The combination of logical analysis and conditional assumptions typical for the goal directed proofs is often observed in human deductive argumentation, and is a natural way to solve logical problems. A logical and philosophical elaboration of the concept *logical analysis* in this connection is forthcoming.

Secondly, also the actual proof procedures for final derivability in section 5 have a realistic character. In both the Reliability procedure and the Minimal Abnormality procedure, one can observe an interesting manner to cope with the typical adaptive dynamics. In the first permissive or careless phase of the procedure one tries to derive the goal on the assumption that everything (that is needed for this derivation) is normal, i.e. on the assumption that some conjunction of disjunctions of abnormalities is not derivable from the premises. For example, in the case of inconsistency adaptive logics, this means trying to obtain the goal by applying classical logic rules on the condition that the relevant subformulas are not inconsistent in view of the premises.

In the next, sceptic phase, the procedure tries to refute these careless reasoning steps, by deriving the conjunction of disjunctions of abnormalities that was taken to be not derivable in the first phase. When a refutation is derived in a careful sense, one should look for other ways to derive the goal. Hence, one returns to the

permissive phase. If no careful refutation is found within the sceptic phase, one still has the possibility that the refutation can be derived in a conditional (careless) way. The critical attitude is also allowed to be careless. This implies of course that the criticism itself can be refuted as well. So, one needs to go through a last phase.

The last phase consists of the attempt to criticize the criticism in a careful way. If this succeeds, the criticism is useless and the sceptic phase has to look for another attempt to criticize the careless derivations from the first phase. The goal is derived if no sensible criticism can be found to the careless derivation of the goal. Or: the goal is derived if every criticism to its derivation can be carefully criticized. This does not sound weird at all from a dialectical point of view.

I have shown that the procedure is decisive for the propositional fragment of the logics \mathbf{CLuN}^m or \mathbf{CLuN}^r with finite premise sets. Moreover, adaptive logics that have a decidable lower limit logic (such as predicative \mathbf{CLuN}^m with maximally unary predicates and finite premise sets), and a finite list of problem relevant abnormalities are decidable as well. So, for an important proportion of the realistic problem solving contexts there is an algorithm that will, within finite time, give an answer.

But a lot of adaptive logic problems are undecidable. Also in these cases the procedure is quite important. It enables the reasoner to obtain a provisional solution to the problem as well as more insights in the premises. These insights are immediately relevant to the solution of the problem. This contrasts sharply with non goal directed, random reasoning from premises: the reasoner gains information, but has no guarantee at all that this is of any use. Relevant information is crucial in view of the creation of alternative interesting statements and in view of other reasoning methods towards the solution (intuitive guessing, doing new empirical research, considering conceptual and/or (methodo)logical changes).

Standard adaptive proofs do not serve as a demonstration for the final derivability of formulas from the premise set. The reasoner can only obtain finality of conviction about the final derivability of formulas derived in the proof, by a reasoning at the metalevel. So, it is important to have a generally applicable procedure that (if it returns an answer) can serve as a means to decide whether lines of a proof are stably marked or unmarked. If the procedures in this paper terminate, they do provide the reasoner with finality of conviction about the final derivability of formulas from premises, because of the general reasoning at the metalevel that proves the correctness of the presented procedures for all premises and conclusions.

Moreover, thanks to the typical proof format, the interesting information that is gained during the problem solving process is also explicitly available in the actual output of the procedure. Both if the answer of the procedure is positive and if it is negative, a proof is obtained in which, in most cases, the goal is the formula element of some (possibly marked) lines of the proof. So, on these lines the goal is derived on two types of conditions: sets of usual formulas and sets of abnormalities. Those conditions are very useful. If the procedure has terminated and failed to derive the conclusion, the failed proof can serve as the basis for further research that can change the premise set. For example, some parts of inconsistent theories can be dropped in order to remove some inconsistencies. Also, if the procedure was not decisive (it did not terminate) after a reasonable time, one can still affirm the goal provisionally. Several lines on which this goal is derived tell us how this affirmation can be falsified.

The presented procedure narrows the gap between the abstract logical rules of

dynamic proofs and the actual ability of the reasoner to solve problems by means of it. It is a rational method to cope with defeasible inferences in every day and scientific contexts.¹²

References

- [1] Atocha Aliseda. *Seeking Explanations: Abduction in Logic, Philosophy of Science and Artificial Intelligence*. Phd thesis, Institute for Logic, Language and Computation (ILLC), University of Amsterdam, The Netherlands, 1997.
- [2] Grigoris Antoniou. *Nonmonotonic Reasoning*. MIT Press, Cambridge, Mass., 1996.
- [3] Diderik Batens. Inconsistency-adaptive logics and the foundation of non-monotonic logics. *Logique et Analyse*, 145:57–94, 1994. Appeared 1996.
- [4] Diderik Batens. Inconsistency-adaptive logics. In Orłowska [36], pages 445–472.
- [5] Diderik Batens. Towards the unification of inconsistency handling mechanisms. *Logic and Logical Philosophy*, 8:5–31, 2000. Appeared 2002.
- [6] Diderik Batens. A dynamic characterization of the pure logic of relevant implication. *Journal of Philosophical Logic*, 30:267–280, 2001.
- [7] Diderik Batens. A strengthening of the Rescher–Manor consequence relations. *Logique et Analyse*, 183–184:289–313, 2003. Appeared 2005.
- [8] Diderik Batens. The need for adaptive logics in epistemology. In Dov Gabbay, S. Rahman, J. Symons, and J. P. Van Bendegem, editors, *Logic, Epistemology and the Unity of Science*, pages 459–485. Kluwer Academic Publishers, Dordrecht, 2004.
- [9] Diderik Batens. On a logic of induction. In Festa et al. [24], pages 221–242. (Contains uncorrected proofs; see [11] for correct version.)
- [10] Diderik Batens. A procedural criterion for final derivability in inconsistency-adaptive logics. *Journal of Applied Logic*, 3:221–250, 2005.
- [11] Diderik Batens. On a logic of induction. *L&PS – Logic & Philosophy of Science*, IV(1):3–32, 2006. (Corrected version of [9].)
- [12] Diderik Batens. A universal logic approach to adaptive logics. *Logica Universalis*, 1:221–242, 2007.
- [13] Diderik Batens and Lieven Haesaert. On classical adaptive logics of induction. *Logique et Analyse*, 173–175:255–290, 2001. Appeared 2003.
- [14] Diderik Batens and Joke Meheus. The adaptive logic of compatibility. *Studia Logica*, 66:327–348, 2000.
- [15] Diderik Batens, Joke Meheus, Dagmar Provijn, and Liza Verhoeven. Some adaptive logics for diagnosis. *Logic and Logical Philosophy*, 11/12:39–65, 2003.
- [16] Diderik Batens and Dagmar Provijn. Pushing the search paths in the proofs. A study in proof heuristics. *Logique et Analyse*, 173–175:113–134, 2001. Appeared 2003.
- [17] Diderik Batens and Timothy Vermeir. Direct dynamic proofs for the Rescher–Manor consequence relations: The flat case. *Journal of Applied Non-Classical Logics*, 12:63–84, 2002.
- [18] Salem Benferhat, Didier Dubois, and Henri Prade. Some syntactic approaches to the handling of inconsistent knowledge bases: A comparative study. Part 1: The flat case. *Studia Logica*, 58:17–45, 1997.
- [19] Salem Benferhat, Didier Dubois, and Henri Prade. Some syntactic approaches to the handling of inconsistent knowledge bases: A comparative study. Part 2: The prioritized case. In Orłowska [36], pages 473–511.
- [20] Gerhard Brewka. *Nonmonotonic Reasoning: Logical Foundations of Commonsense*. Cambridge University Press, Cambridge, Mass., 1991.
- [21] Newton C.A. da Costa, Otávio Bueno, and Steven French. The logic of pragmatic truth. *Journal of Philosophical Logic*, 27:603–620, 1998.
- [22] Kristof De Clercq. Two new strategies for inconsistency-adaptive logics. *Logic and Logical Philosophy*, 8:65–80, 2000. Appeared 2002.

¹²Research for this paper was supported by subventions from Ghent University and from the Fund for Scientific Research – Flanders. I am indebted to Diderik Batens for comments on a former draft.

24 A Proof Procedure for Adaptive Logics

- [23] Leen De Vreese and Erik Weber. Applications of the adaptive logic for causal discovery. *Logique et Analyse*, 185–188:33–51, 2004. Appeared 2005.
- [24] Roberto Festa, Atocha Aliseda, and Jeanne Peijnenburg, editors. *Confirmation, Empirical Progress, and Truth Approximation. Essays in Debate with Theo Kuipers. Vol. 1*, volume 83 of *Poznan Studies in the Philosophy of the Sciences and the Humanities*. Rodopi, Amsterdam/New York, 2005.
- [25] Leon Horsten and Philip Welch. The undecidability of propositional adaptive logic. *Synthese*, to appear, 2007.
- [26] Theo A. F. Kuipers. *From Instrumentalism to Constructive Realism. On some Relations Between Confirmation, Empirical Progress, and Truth Approximation*, volume 287 of *Synthese Library*. Kluwer, Dordrecht, 2000.
- [27] Witold Luskaszewicz. *Non-Monotonic Reasoning. Formalization of Commonsense Reasoning*. Ellis Horwood, New York, 1990.
- [28] Joke Meheus. Erotetic arguments from inconsistent premises. *Logique et Analyse*, 165–166:49–80, 1999. Appeared 2002.
- [29] Joke Meheus. Adaptive logics for question evocation. *Logique et Analyse*, 173–175:135–164, 2001. Appeared 2003.
- [30] Joke Meheus. An adaptive logic for pragmatic truth. In Walter A. Carnielli, Marcelo E. Coniglio, and Itala M. Loffredo D’Ottaviano, editors, *Paraconsistency. The Logical Way to the Inconsistent*, pages 167–185. Marcel Dekker, New York, 2002.
- [31] Joke Meheus. Paraconsistent compatibility. *Logique et Analyse*, 183–184:251–287, 2003. Appeared 2005.
- [32] Joke Meheus. Empirical progress and ampliative adaptive logics. In Festa et al. [24], pages 193–217.
- [33] Joke Meheus and Diderik Batens. A formal logic for abductive reasoning. *Logic Journal of the IGPL*, 14:221–236, 2006.
- [34] Joke Meheus, Liza Verhoeven, Maarten Van Dyck, and Dagmar Provijn. Ampliative adaptive logics and the foundation of logic-based approaches to abduction. In Lorenzo Magnani, Nancy J. Nersessian, and Claudio Pizzi, editors, *Logical and Computational Aspects of Model-Based Reasoning*, pages 39–71. Kluwer, Dordrecht, 2002.
- [35] Irene Mikenberg, Newton C. A. da Costa, and Rolando Chuaqui. Pragmatic truth and approximation to truth. *Journal of Symbolic Logic*, 51:201–221, 1986.
- [36] Ewa Orłowska, editor. *Logic at Work. Essays Dedicated to the Memory of Helena Rasiowa*. Physica Verlag (Springer), Heidelberg, New York, 1999.
- [37] Dagmar Provijn. *Prospectieve dynamiek. Filosofische en technische onderbouwing van doelgerichte bewijzen en bewijsheuristieken*. PhD thesis, Universiteit Gent (Belgium), 2005. Unpublished PhD thesis.
- [38] Nicholas Rescher and Ruth Manor. On inference from inconsistent premises. *Theory and Decision*, 1:179–217, 1970.
- [39] Kurt Schütte. *Beweistheorie*. Springer, Berlin, 1960.
- [40] Raymond M. Smullyan. *First Order Logic*. Dover, New York, 1995. Original edition: Springer, 1968.
- [41] Stephan van der Waart van Gulik and Peter Verdée. Modeling sorites reasoning with adaptive fuzzy logic. Forthcoming, 200x.
- [42] Maarten Van Dyck. Causal discovery using adaptive logics. Towards a more realistic heuristics for human causal learning. *Logique et Analyse*, 185–188:5–32, 2004. Appeared 2005.
- [43] Bart Van Kerckhove and Guido Vanackere. Vagueness-adaptive logic: A pragmatical approach to sorites paradoxes. *Studia Logica*, To appear.
- [44] Guido Vanackere. Ambiguity-adaptive logic. *Logique et Analyse*, 159:261–280, 1997. Appeared 1999.
- [45] Guido Vanackere. Minimizing ambiguity and paraconsistency. *Logique et Analyse*, 165–166:139–160, 1999. Appeared 2002.
- [46] Guido Vanackere. Preferences as inconsistency-resolvers: the inconsistency-adaptive logic **PRL**. *Logic and Logical Philosophy*, 8:47–63, 2000. Appeared 2002.

- [47] Guido Vanackere. **HL2**. An inconsistency-adaptive and inconsistency-resolving logic for general statements that might have exceptions. *Journal of Applied Non-Classical Logics*, 10:317–338, 2000.
- [48] Guido Vanackere. The role of ambiguities in the construction of collective theories. *Logique et Analyse*, 173–174–175:189–214, 2001. Appeared 2003.
- [49] Peter Verdée. Adaptive logics using the minimal abnormality strategy are Π_1^1 -complex. Forthcoming, 200x.
- [50] Peter Verdée and Stephan van der Waart van Gulik. A generic framework for adaptive vague logics. Forthcoming, 200x.
- [51] Liza Verhoeven. All premisses are equal, but some are more equal than others. *Logique et Analyse*, 173–174–175:165–188, 2001. Appeared 2003.
- [52] Liza Verhoeven. Proof theories for some prioritized consequence relations. *Logique et Analyse*, 183–184:325–344, 2003. appeared 2005.
- [53] Andrzej Wiśniewski. *The Posing of Questions. Logical Foundations of Erotetic Inferences*. Kluwer, Dordrecht, 1995.

Received June 1, 2010