Chapter 8

Corrective Adaptive Logics

Blaming all triviality on inconsistency seems a mistake. I shall present corrective adaptive logics that are not or not only inconsistency-adaptive. These allow for gluts or gaps (or both) with respect to all (as well as several) logical symbols. All basic such logics will be reviewed. I shall also outline variants as well as the combined adaptive logics built from the aforementioned corrective adaptive logics. A very different matter is the topic of Section 8.3: a particular ambiguity-adaptive logic. It allows for ambiguous non-logical symbols but handles them adaptively, minimizing ambiguities. A fascinating result, presented in Section 8.4, is that all these forms of adaptation may be combined. This leads to an adaptive logic that has an empty lower limit logic—nothing follows from any premise set, no logical or non-logical symbol has any definite meaning. Nevertheless, the adaptive logic is a useful and fascinating tool. A more general result, presented in Section 8.6, is a criterion for separating corrective adaptive logics in flip-flop logics and others.

8.1 Not Only Inconsistency-Adaptive Logics

Most contemporary (first-order) theories, whether mathematical or empirical, can be seen as having **CL** as their underlying logic. Most earlier theories may be sensibly interpreted similarly. Yet, as was explained in Section 2.1, some of these theories turned out to be trivial if taken literally. How should one proceed in such situations? The advice given in Section 2.1 was to interpret such theories as consistently as possible, and next to try removing the inconsistencies. But this is not the only possible way to proceed.

Classical logicians seem mesmerized by negation. Whenever a theory turns out to have no **CL**-models, they analyse the situation as an inconsistency: for some formula A, the theory requires that both A and $\neg A$ are true. Paraconsistent logicians seem equally mesmerized by negation. Whenever a theory turns out to have no **CL**-models, paraconsistent logicians argue that this shows that one needs models in which, for some A, both A and $\neg A$ are true. As announced, other approaches are possible.

We have seen, in Section 2.2, that the **CL**-clause for negation may be seen as consisting of the consistency requirement

$$\text{if } v_M(A) = 1 \text{ then } v_M(\neg A) = 0$$

which rules out negation gluts—for some A, both A and $\neg A$ are true—and the (negation-)completeness requirement

$$\text{if } v_M(A) = 0 \text{ then } v_M(\neg A) = 1$$

which rules out negation gaps—for some A, both A and $\neg A$ are false. Both classical logicians and paraconsistent logicians concentrate only on negation gluts. Classical logicians identify the triviality of a theory with the presence of negation gluts, whereas paraconsistent logicians stress that some theories display negation gluts without being trivial.

Consider the set $\{p, \neg\neg\neg p\}$. According to **CLuN**, this set has three kinds of models: (i) those in which p, $\neg p$, and $\neg\neg\neg p$ are true and $\neg\neg p$ is false, (ii) those in which p, $\neg\neg p$, and $\neg\neg\neg p$ are true and $\neg p$ is false, and (iii) those in which p, $\neg p$, $\neg\neg p$, and $\neg\neg\neg p$ are all true. If, however, one allows for negation gaps, there are models in which p and $\neg\neg\neg p$ are true, whereas $\neg p$ and $\neg\neg p$ are false. Such models 'explain' the problem just as well as the aforementioned **CLuN**-models. If the negation-completeness requirement is dropped, both $\neg p$ and $\neg\neg p$ may be false, which allows p and $\neg\neg\neg p$ to be true.

The logic which is a 'counterpart' of **CLuN** but allows for negation-gaps rather than negation gluts will be called **CLaN**—it is just like **CL** except that it allows for g*a*ps with respect to *n*egation. Its indeterministic semantics is obviously obtained by dropping the negation-completeness requirement from the **CL**-semantics. Its deterministic semantics and axiomatization will be spelled out below.

Consider a theory T that had **CL** as its underlying logic but turns out to be trivial. Suppose moreover that T has **CLaN**-models and hence that one may remove its triviality by replacing the underlying logic **CL** by **CLaN**. The result, call it T', is a negation-incomplete theory. By the same reasoning as was used in Section 2.1, T' is too weak in comparison to what T was intended to be. So we shall want to interpret the negation-incomplete T' as negation-complete as possible. In other words, we shall want to minimize the negation gaps. To do so, we have to go adaptive.

Going adaptive requires, according to the standard format, a lower limit logic, a set of abnormalities, and a strategy. The lower limit logic is obviously **CLaN** and the strategy is Minimal Abnormality or Reliability. What is the set of abnormalities? Clearly, the kind of formulas we want to consider as false unless the premises require them to be true. Clearly we want $A \vee \neg A$ to be *true* unless the premises require it to be false. However, we need formulas that will be considered as *false* unless the premises require them to be true. The presence of the classical logical symbols enables one to express this: the abnormalities will be the formulas of the form $\check{\neg}(A \vee \neg A)$.

If we need to use classical logical symbols anyway, there is a more transparent way to characterize the abnormalities. Consider a **CLaN**-model in which both A and $\neg A$ are false. Instead of saying that the model verifies $\check{\neg}(A \vee \neg A)$, we may just as well say that it verifies $\check{\neg}A \,\check{\wedge}\, \check{\neg}\neg A$. In **CLaN**, the standard negation has the same meaning as the classical negation. I use the classical negation in the present context in view of the convention from Section 4.3. Actually, the use of classical negation in the present context will prove very handy in the sequel of this section.

The formula $\check{\neg}A \,\check{\wedge}\, \check{\neg}\neg A$ nicely expresses what we mean by an abnormality in the present context: A is *false* in the model and $\neg A$ is also *false* in it. And there is another instructive reading: the model *verifies* $\check{\neg}A$ but *falsifies* $\neg A$. This clearly expresses a negation gap: the *classical* negation of A is verified but the *standard* negation of A is not. So the standard negation displays a gap. Of course, abnormalities have to be existentially closed for the predicative level. So we define $\Omega = \{\check{\exists}(\check{\neg}A \mathbin{\check{\wedge}} \check{\neg}\neg A) \mid A \in \mathcal{F}_s\}$.

It is instructive to check what becomes of the **CLuN**-abnormalities if the same transformation is applied to them. Before, the **CLuN**-abnormalities were defined as $\Omega = \{\exists(A \wedge \neg A) \mid A \in \mathcal{F}_s\}$. It is just as good to define them as $\Omega = \{\check{\exists}(\check{\neg}\check{\neg}A \mathbin{\check{\wedge}} \neg A) \mid A \in \mathcal{F}_s\}$. The form of these abnormalities clearly indicates a negation *glut*. Applied to models: the model falsifies the classical negation of A but nevertheless verifies the standard negation.

Let me reassure the suspicious reader that one obtains the same logics $\mathbf{CLuN}^r$ and $\mathbf{CLuN}^m$ if one defines $\Omega = \{\check{\exists}(\check{\neg}\check{\neg}A \mathbin{\check{\wedge}} \neg A) \mid A \in \mathcal{F}_s\}$. For example whenever a model verifies $\check{\exists}(\check{\neg}\check{\neg}A \mathbin{\check{\wedge}} \neg A)$ for some A, it verifies $\exists(A \wedge \neg A)$ for the same A; and *vice versa*. Whenever the first formula is derivable from a premise set for an A, so is the second formula for that A; and *vice versa*. The original formulation has the advantage that abnormalities are expressed in the standard language. What is attractive about the reformulation, however, is that we now have a unified way to characterize negation gluts and negation gaps and that this characterization is transparent. Moreover, this approach may be generalized to all logical symbols.

Consider another example, the premise set $\{p, q, \neg(p \wedge q)\}$. At first sight, handling this sets seems to require that one allows for inconsistencies, in other words for negation gluts. But suppose we have models with conjunction gaps: the classical conjunction of A and B is true, but their standard conjunction is false. So the abnormalities will have the form $\check{\exists}((A \mathbin{\check{\wedge}} B) \mathbin{\check{\wedge}} \check{\neg}(A \wedge B))$.[1] If a model of $\{p, q, \neg(p \wedge q)\}$ allows for conjunction gaps, it will verify p and q, and hence also $p \mathbin{\check{\wedge}} q$, but it may falsify $p \wedge q$, in which case it verifies $\check{\neg}(p \wedge q)$ as well as (if there are no gaps for the standard negation) $\neg(p \wedge q)$. In other words, the premise set $\{p, q, \neg(p \wedge q)\}$ does not require paraconsistent models. It has just as well models of logics that allow for conjunction gaps, even if they allow for no other gluts or gaps.

Some premise sets are even more amusing. Consider $\{p, r, \neg q \vee \neg r, (p \wedge r) \supset q\}$. This clearly has no **CL**-models. It has models if one allows for negation gluts, but also if one allows for conjunction gaps, or for disjunction gluts, or for implication gluts. In general, for every gap or glut with respect to any logical symbol, there are premise sets that have no **CL**-models but have models of the logic that allows just for such gluts or gaps.

I claimed that classical logicians and paraconsistent logicians are both mesmerized by negation gluts. There is an easy historical explanation for this: all gluts and gaps *surface* as inconsistencies if **CL** is applied to the premise set. Thus, if **CL** is applied to $\{p, \neg\neg\neg p\}$, one obtains the inconsistencies $p \wedge \neg p$ and $\neg p \wedge \neg\neg p$ (as well as all others of course). Similarly if **CL** is applied to $\{p, q, \neg(p \wedge q)\}$. The situation is exactly the same for any other gluts and gaps. In all cases an inconsistency surfaces when one applies **CL**.

That all gluts and gaps surface as inconsistencies makes it understandable why there was and is ample interest in paraconsistent logics, but much less

[1] I add the existential closure because a model verifying $\check{\exists}x((Px \mathbin{\check{\wedge}} Qx) \mathbin{\check{\wedge}} \check{\neg}(Px \wedge Qx))$ verifies a conjunction gap even if it does not verify any instance of that formula.

in logics that display other kinds of gluts or gaps (or both). Nevertheless, it seems to me that it is a mistake to concentrate on consistency only. Remember that the plot behind inconsistency-adaptive logics was to localize and isolate the *problems* displayed by a theory or premise set and to do so in order to remove those problems. Inconsistency-adaptive logics always identify disjunctions of inconsistencies as the problems. Suppose one chooses a logic **L** that allows for other kinds of gluts or gaps and that one applies an adaptive logic that has **L** as its lower limit. Other formulas may then be identified as the problems and often there is quite some choice, as in the case of $\{p, r, \neg q \vee \neg r, (p \wedge q) \supset q\}$. Although *Dab*-formulas will be derivable for every choice, the *Dab*-formulas will be different. So different problems have to be resolved if one wants to regain consistency, whence different consistent alternatives are suggested. From a purely logical point of view, it is sensible to consider all possibilities. Some choices of gluts or gaps may cause less 'problems' than others. Moreover, there may be extra-logical reasons to prefer certain consistent alternatives over others.

I shall now describe the basic logics that allow for gluts or gaps in comparison to **CL**. Combinations of different kinds of gluts or gaps will be considered thereafter, but it is easier to mention the combination of gluts and gaps of the same kind from the very beginning.

Devising the basic logics, one may proceed in a systematic way. All clauses of the **CL**-semantics concern a 'basic form': schematic letters for sentences, primitive predicative expressions, and the forms characterized by a metalinguistic formula that contains precisely one logical symbol, identity included. Each of these clauses may be split into two implicative clauses. The consequence of one of the implicative clauses states the condition that causes formulas of the form A to receive the valuation value 0. This implicative clause rules out a kind of gluts. The consequence of the other implicative clause states the condition that causes formulas of the form A to receive the valuation value 1. This rules out gaps of a particular kind. So obtaining the basic logics is straightforward.

Consider first gluts for a particular logical form A. Each of the logics described below allows for a single kind of gluts, and does not allow for any gaps. The indeterministic semantics is obtained by removing from the **CL**-semantics the implicative clause that has $v_M(\mathsf{A}) = 0$ as its implicatum. In order to illustrate the naming scheme, I shall list all glut variants, including gluts for sentential letters and for primitive predicative expressions.[2] In view of what precedes, the names of the logics are self-explanatory, except perhaps the use of "M" for material implication (because I need the "I" for identity) and the use of "X", the second letter of "existential" (because I need the "E" for equivalence).

[2]These cause trouble on which I shall comment later in the text.

logic	*removed* implicative clause
CLuS	where $A \in \mathcal{S}$, if $v(A) = 0$ then $v_M(A) = 0$
CLuP	if $\langle v(\alpha_1), \ldots, v(\alpha_r)\rangle \notin v(\pi^r)$ then $v_M(\pi^r\alpha_1 \ldots \alpha_r) = 0$
CLuI	if $v(\alpha) \neq v(\beta)$ then $v_M(\alpha = \beta) = 0$
CLuN	if $v_M(A) = 1$ then $v_M(\neg A) = 0$
CLuM	if $v_M(A) = 1$ and $v_M(B) = 0$, then $v_M(A \supset B) = 0$
CLuC	if $v_M(A) = 0$ or $v_M(B) = 0$, then $v_M(A \wedge B) = 0$
CLuD	if $v_M(A) = 0$ and $v_M(B) = 0$, then $v_M(A \vee B) = 0$
CLuE	if $v_M(A) \neq v_M(B)$, then $v_M(A \equiv B) = 0$
CLuU	if $\{v_M(A(\beta)) \mid \beta \in \mathcal{C} \cup \mathcal{O}\} \neq \{1\}$, then $v_M(\forall\alpha A(\alpha)) = 0$
CLuX	if $1 \notin \{v_M(A(\beta)) \mid \beta \in \mathcal{C} \cup \mathcal{O}\}$, then $v_M(\exists\alpha A(\alpha)) = 0$

Each of these logics has a deterministic semantics. This requires a clause of the form "$v_M(\mathsf{A}) = 1$ iff [condition]". This clause is obtained from the **CL**-semantics by or-ing the condition of the standard clause with the correct reference to the assignment value: "$v(\mathsf{A}) = 1$". I again list all the logics.

logic	replacing clause
CLuS	where $A \in \mathcal{S}$, $v_M(A) = 1$ iff $v(A) = 1$ or $v(A) = 1$
CLuP	$v_M(\pi^r\alpha_1 \ldots \alpha_r) = 1$ iff $\langle v(\alpha_1), \ldots, v(\alpha_r)\rangle \in v(\pi^r)$ or $v(\pi^r\alpha_1 \ldots \alpha_r) = 1$
CLuI	$v_M(\alpha = \beta) = 1$ iff $v(\alpha) = v(\beta)$ or $v(\alpha = \beta) = 1$
CLuN	$v_M(\neg A) = 1$ iff $v_M(A) = 0$ or $v(\neg A) = 1$
CLuM	$v_M(A \supset B) = 1$ iff $(v_M(A) = 0$ or $v_M(B) = 1)$ or $v(A \supset B) = 1$
CLuC	$v_M(A \wedge B) = 1$ iff $v_M(A) = 1$ and $v_M(B) = 1$ or $v(A \wedge B) = 1$
CLuD	$v_M(A \vee B) = 1$ iff $v_M(A) = 1$ or $v_M(B) = 1$ or $v(A \vee B) = 1$
CLuE	$v_M(A \equiv B) = 1$ iff $v_M(A) = v_M(B)$ or $v(A \equiv B) = 1$
CLuU	$v_M(\forall\alpha A(\alpha)) = 1$ iff $\{v_M(A(\beta)) \mid \beta \in \mathcal{C} \cup \mathcal{O}\} = \{1\}$ or $v(\forall\alpha A(\alpha)) = 1$
CLuX	$v_M(\exists\alpha A(\alpha)) = 1$ iff $1 \in \{v_M(A(\beta)) \mid \beta \in \mathcal{C} \cup \mathcal{O}\}$ or $v(\exists\alpha A(\alpha)) = 1$

Needless to say, all other clauses of the **CL**-semantics are retained.

Some readers may worry at this point. Is it really obvious that the indeterministic semantics defines the same logic as the deterministic semantics? It is. Please check the proof outline of Theorem 2.2.1. This is easily adjusted for any logic mentioned in the last table.

Nearly all glut-logics have nice adequate axiomatizations in $\mathcal{W}_s$. For **CLuC**, for example, it is sufficient to remove from the axiom system of **CL** the axioms A∧1 and A∧2, and to attach to A=2 the restriction "provided $A(\alpha) \in \mathcal{W}_s^p$", just as we did for **CLuN** in Section 2.2. However, as the reader will have seen, this way of proceeding gets us into trouble when we come to implication gluts. As, later on, I have to consider combinations of gluts and gaps, the trouble will spread.

A different road is possible, and it is instructive. Consider the axiom system of **CL**, replace in every axiom and rule every standard symbol by the corresponding classical symbol, and attach to axiom schema A=2 the restriction that $A \in \mathcal{W}_{\check{s}}$.[3] Call this axiom system **CLC**.[4] Next add, for every logical symbol, the axiom that gives the standard symbol the same meaning as the classical

[3]Remember that the members of $\mathcal{W}_{\check{s}}$ contain only classical symbols and no standard ones.

[4]This is an axiom system for **CL**. The restriction on A=2 causes no weakening because one may derive the original version of A=2 for all members of $\mathcal{W}_{\check{s}}$.

symbol—example: $\neg A \check{\equiv} \check{\neg} A$. So all standard symbols have their **CL**-meaning in **CLC**. To obtain an axiomatic system that allows for gluts with respect to a specific logical form A, remove the relevant equivalence and replace it by a glut-tolerating implication. I do not list all of them as they are all similar. Gluts with respect to sentential letters and primitive predicative formulas will be commented upon below.

logic	axiom
CLuI	$\alpha \check{=} \beta \check{\supset} \alpha = \beta$
CLuN	$\check{\neg} A \check{\supset} \neg A$
⋮	⋮
CLuX	$\check{\exists} \alpha A(\alpha) \check{\supset} \exists \alpha A(\alpha)$

So the matter is utterly simple. As the standard symbol may display gluts, the formula containing the standard symbol is logically implied by the formula containing the corresponding classical symbol, but not *vice versa*.

Note that these axiom systems agree with the convention from Section 2.5: no classical symbol occurs within the scope of a standard symbol. Note also the direct relation between the implicative glut-tolerating axiom and the relevant retained clause in the indeterministic semantics. Just as **CLuI** contains the axiom $\alpha \check{=} \beta \check{\supset} \alpha = \beta$ and not its converse, the indeterministic **CLuI**-semantics contains the clause "if $v(\alpha) = v(\beta)$ then $v_M(\alpha = \beta) = 1$". Note that the antecedent of the clause, $v(\alpha) = v(\beta)$, is the semantic definition of the antecedent of the axiom, $\alpha \check{=} \beta$.

As I promised, I now comment on the logics **CLuS** and **CLuP**. No axiomatic system for **CLuS** is provided by the previous paragraphs. There is no need to do so, as it is obvious from the deterministic semantics that **CLuS** is identical to **CL**. So I shall never refer to it again by the funny name **CLuS**.

For **CLuP** the matter is more complicated. Again, no axiomatic system for it is provided in the previous paragraphs. **CLuP** does have a decent axiomatization, but its peculiarities are even incompatible with **CLC**. To see this, it is sufficient to realize (i) that $v_M(\pi^r\alpha_1 \ldots \alpha_r)$ may be 1 because $v(\pi^r\alpha_1 \ldots \alpha_r) = 1$, even if $\langle v(\alpha_1), \ldots, v(\alpha_r)\rangle \notin v(\pi^r)$ and (ii) that $v(\alpha_1) = v(\beta)$ does not warrant that $v_M(\pi^r\beta\alpha_2 \ldots \alpha_r) = v_M(\pi^r\alpha_1\alpha_2 \ldots \alpha_r)$. The axiomatization of **CLuP** requires that one starts from the axiomatic system for **CL** in the language $\mathcal{L}_{\check{s}}$ but with A=2 *removed*. Next every standard symbol should be given the same meaning as the classical symbol. The result is an odd logic in which even the classical symbols do not have the right meaning. So no adaptive logic will be built on this logic.[5]

We are done with the basic logics for gluts and can move on to logics that allow for one kind of gaps. Their indeterministic semantics is obtained by removing from the **CL**-semantics the implicative clause that has $v_M(\mathsf{A}) = 1$ as its implicatum. All these logics will have a lower case "a", referring to the possibility of gaps, where their glut-counterparts have a lower case "u". By now, I suppose that the reader understood the plot and skip most of the logics,

[5]The attentive reader may have remarked that variants for **CLuS** and **CLuP** may be devised in which one explicitly distinguishes between the classical meaning of sentential letters and predicates, denoted for example as $\check{p}$ and $\check{P}a$, and the standard meaning of such entities, denoted by p and Pa. On the semantics, $\check{p} \check{\equiv} p$ and $\check{P}a \check{\supset} Pa$ are valid, but not the converse of the latter. I shall not pursue this road here.

except that I include gaps for sentential letters and for primitive predicative expressions—these will be commented upon below.

logic	*removed* implicative clause
CLaS	where $A \in \mathcal{S}$, if $v(A) = 1$ then $v_M(A) = 1$
CLaP	if $\langle v(\alpha_1), \ldots, v(\alpha_r)\rangle \in v(\pi^r)$ then $v_M(\pi^r\alpha_1 \ldots \alpha_r) = 1$
CLaI	if $v(\alpha) = v(\beta)$ then $v_M(\alpha = \beta) = 1$
CLaN	if $v_M(A) = 0$ then $v_M(\neg A) = 1$
⋮	⋮
CLaX	if $1 \in \{v_M(A(\beta)) \mid \beta \in \mathcal{C} \cup \mathcal{O}\}$, then $v_M(\exists\alpha A(\alpha)) = 1$

Each of these logics has a deterministic semantics, which requires a clause of the form "$v_M(\mathsf{A}) = 1$ iff [condition]". This clause is obtained from the **CL**-semantics by and-ing the condition of the standard clause with the correct reference to the assignment value: "$v(\mathsf{A}) = 1$".

logic	replacing clause
CLaS	where $A \in \mathcal{S}$, $v_M(A) = 1$ iff $v(A) = 1$ and $v(A) = 1$
CLaP	$v_M(\pi^r\alpha_1 \ldots \alpha_r) = 1$ iff $\langle v(\alpha_1), \ldots, v(\alpha_r)\rangle \in v(\pi^r)$ and $v(\pi^r\alpha_1 \ldots \alpha_r) = 1$
CLaI	$v_M(\alpha = \beta) = 1$ iff $v(\alpha) = v(\beta)$ and $v(\alpha = \beta) = 1$
CLaN	$v_M(\neg A) = 1$ iff $v_M(A) = 0$ and $v(\neg A) = 1$
⋮	⋮
CLaX	$v_M(\exists\alpha A(\alpha)) = 1$ iff $1 \in \{v_M(A(\beta)) \mid \beta \in \mathcal{C} \cup \mathcal{O}\}$ and $v(\exists\alpha A(\alpha)) = 1$

As for the glut-variants, all other clauses of the **CL**-semantics are retained.

The way in which gaps are realized is fully transparent. Consider $v_M(\neg A) = 1$. As this may be a negation glut, that $v_M(A) = 0$ is necessary but not sufficient. We need something more. The 'something more' is taken care of by requiring that moreover $v(\neg A) = 1$.

For the axiomatization, I shall again follow the road taken for the glut-allowing logics. Here are the axioms.

logic	axiom
CLaI	$\alpha = \beta \check{\supset} \alpha \check{=} \beta$
CLaN	$\neg A \check{\supset} \check{\neg} A$
⋮	⋮
CLaX	$\exists\alpha A(\alpha) \check{\supset} \check{\exists}\alpha A(\alpha)$

Again, the matter is utterly simple. As the standard symbol may display gaps (and no gluts), the formula containing the classical symbol is logically implied by the formula containing the corresponding standard symbol, but not *vice versa*. Again, all logical symbols for which no gaps are permitted are characterized by an axiom stating that a formula containing the standard symbol is classically equivalent to the corresponding expression containing the classical symbol.

Some will find the classical contraposition of the axioms more transparent, for example $\check{\neg}\alpha \check{=} \beta \check{\supset} \check{\neg}\alpha = \beta$ for **CLaI**. This also illustrates the direct connection between the axiom and the corresponding retained clause of the indeterministic semantics.

I still have to comment upon **CLaS** and **CLaP**. No axiomatic system for **CLaS** is provided above, and rightly so as it is obvious from the deterministic semantics that **CLaS** is identical to **CL**. So I shall never refer to it again by the funny name **CLaS**.

The logic **CLaP** is identical to **CLuP** and displays the same oddities. I shall not refer to it in the sequel because this logic cannot function as an adaptive logic in standard format.

Let us now move to the case where gluts and gaps for the same logical form are combined. The names of the logics contain a lower case "o" to indicate that *both* gluts and gaps are possible. For the indeterministic semantics, one removes both the clause preventing gluts and the clause preventing gaps. This means that one removes the **CL**-clause altogether.

logic	removed implicative clauses
CLoS	where $A \in \mathcal{S}$, if $v(A) = 0$ then $v_M(A) = 0$
	where $A \in \mathcal{S}$, if $v(A) = 1$ then $v_M(A) = 1$
CLoP	if $\langle v(\alpha_1), \ldots, v(\alpha_r)\rangle \notin v(\pi^r)$ then $v_M(\pi^r\alpha_1 \ldots \alpha_r) = 0$
	if $\langle v(\alpha_1), \ldots, v(\alpha_r)\rangle \in v(\pi^r)$ then $v_M(\pi^r\alpha_1 \ldots \alpha_r) = 1$
CLoI	if $v(\alpha) \neq v(\beta)$ then $v_M(\alpha = \beta) = 0$
	if $v(\alpha) = v(\beta)$ then $v_M(\alpha = \beta) = 1$
CLoN	if $v_M(A) = 1$ then $v_M(\neg A) = 0$
	if $v_M(A) = 0$ then $v_M(\neg A) = 1$
$\vdots$	$\vdots$
CLoX	if $1 \notin \{v_M(A(\beta)) \mid \beta \in \mathcal{C} \cup \mathcal{O}\}$, then $v_M(\exists\alpha A(\alpha)) = 0$
	if $1 \in \{v_M(A(\beta)) \mid \beta \in \mathcal{C} \cup \mathcal{O}\}$, then $v_M(\exists\alpha A(\alpha)) = 1$

The deterministic semantics is also simple: the truth-value of composing formulas play no role whatsoever.

logic	replacing clause
CLoS	where $A \in \mathcal{S}$, $v_M(A) = v(A)$
CLoP	$v_M(\pi^r\alpha_1 \ldots \alpha_r) = 1 = v(\pi^r\alpha_1 \ldots \alpha_r) = 1$
CLoI	$v_M(\alpha = \beta) = v(\alpha = \beta) = 1$
CLoN	$v_M(\neg A) = v(\neg A)$
$\vdots$	$\vdots$
CLoX	$v_M(\exists\alpha A(\alpha)) = v(\exists\alpha A(\alpha))$

The way to obtain the axiomatic system corresponds closely to the indeterministic semantics: one removes the axiom concerning the symbol, for example $\alpha \doteq \beta \equiv \alpha = \beta$ for **CLoI**. As a result, the standard identity does not occur in any axiom of **CLoI**, while all other standard symbols are identified with their classical counterparts. The logic **CLoS** is again identical to **CL**, whereas **CLoP** is the same logic as **CLuP** and **CLaP**.

It is obviously possible to formulate logics that allow for a combination of gluts and gaps for different symbols. We may form names for such logics by combining the qualifications that appear in the already used names. Thus **CLoNaM** allows for negation gluts, negation gaps, and implication gaps. To obtain, for example, the indeterministic semantics of **CLoNaM**, remove both implicative clauses on negation, as it was done for **CLoN**, and moreover remove the clause that prevents implication gaps. To obtain the deterministic semantics, one starts from the semantics for **CLoN** and replaces the implication clause by the implication clause from the **CLaM**-semantics. Similarly for the axiomatic systems.

Note that there is a logic that allows for gluts and gaps with respect to all logical symbols. Let us call it **CLo**. In this logic, no standard symbol is given a meaning. So if $\Gamma \in \mathcal{W}_s$, then $Cn^{\mathcal{L}_s}_{\mathbf{CLo}}(\Gamma) = \Gamma$. All this will seem of little interest, unless one remembers the reason to consider all these logics, which is to let them function as the lower limit of an adaptive logic. So let us have a look at the adaptive logics.

As announced, I shall disregard the logics that (attempt to) display gluts or gaps with respect to sentential letters or primitive predicative expressions. For the other logics, the matter is simple. I have already described the lower limits. To obtain adaptive logics in standard format, we need to combine those with either Reliability or Minimal Abnormality as well as with the right set of abnormalities. So all I have to describe are the sets of abnormalities and it was outlined before in which way these are obtained. So the adaptive logics allowing for one kind of gluts are the following.

LLL	set of abnormalities Ω
CLuI	$\{\check{\exists}(\check{\neg}\alpha \check{=} \beta \check{\wedge} \alpha = \beta) \mid \alpha, \beta \in \mathcal{C} \cup \mathcal{V}\}$
CLuN	$\{\check{\exists}(\check{\neg}\check{\neg}A \check{\wedge} \neg A) \mid A \in \mathcal{F}_s\}$
CLuM	$\{\check{\exists}(\check{\neg}(A \check{\supset} B) \check{\wedge} (A \supset B)) \mid A, B \in \mathcal{F}_s\}$
$\vdots$	$\vdots$
CLuX	$\{\check{\exists}(\check{\neg}\check{\exists}\alpha A(\alpha) \check{\wedge} \exists\alpha A(\alpha)) \mid A \in \mathcal{F}_s\}$

And here are the adaptive logics allowing for one kind of gaps.

LLL	set of abnormalities Ω
CLaI	$\{\check{\exists}(\alpha \check{=} \beta \check{\wedge} \check{\neg}\alpha = \beta) \mid \alpha, \beta \in \mathcal{C} \cup \mathcal{V}\}$
CLaN	$\{\check{\exists}(\check{\neg}A \check{\wedge} \check{\neg}\neg A) \mid A \in \mathcal{F}_s\}$
CLaM	$\{\check{\exists}((A \check{\supset} B) \check{\wedge} \check{\neg}(A \supset B)) \mid A, B \in \mathcal{F}_s\}$
$\vdots$	$\vdots$
CLaX	$\{\check{\exists}(\check{\exists}\alpha A(\alpha) \check{\wedge} \check{\neg}\exists\alpha A(\alpha)) \mid A \in \mathcal{F}_s\}$

If the lower limit logic allows for gluts as well as gaps with respect to the same logical symbol, the appropriate set of abnormalities is the union of two sets of abnormalities: that of the corresponding logic allowing for gluts and that of the corresponding logic allowing for gaps. Thus the appropriate set of abnormalities for **CLoI** is $\Omega = \{\check{\exists}(\check{\neg}\alpha \check{=} \beta \check{\wedge} \alpha = \beta) \mid \alpha, \beta \in \mathcal{C} \cup \mathcal{V}\} \cup \{\check{\exists}(\alpha \check{=} \beta \check{\wedge} \check{\neg}\alpha = \beta) \mid \alpha, \beta \in \mathcal{C} \cup \mathcal{V}\}$ and the appropriate set of abnormalities for **CLoX** is $\{\check{\exists}(\check{\neg}\check{\exists}\alpha A(\alpha) \check{\wedge} \exists\alpha A(\alpha)) \mid A \in \mathcal{F}_s\} \cup \{\check{\exists}(\check{\exists}\alpha A(\alpha) \check{\wedge} \check{\neg}\exists\alpha A(\alpha)) \mid A \in \mathcal{F}_s\}$.

Handling logics that combine gluts or gaps for different logical symbols is just as easy. The appropriate set of abnormalities is the union of the sets that contain those gluts and gaps. Thus the appropriate set of abnormalities for **CLoNaM** is $\Omega = \{\check{\exists}(\check{\neg}\check{\neg}A \check{\wedge} \neg A) \mid A \in \mathcal{F}_s\} \cup \{\check{\exists}(\check{\neg}A \check{\wedge} \check{\neg}\neg A) \mid A \in \mathcal{F}_s\} \cup \{\check{\exists}((A \check{\supset} B) \check{\wedge} \check{\neg}(A \supset B)) \mid A, B \in \mathcal{F}_s\}$.

The appropriate set of abnormalities for **CLo** is obviously the union of all sets of abnormalities mentioned (explicitly or implicitly) in the two preceding tables. Incidentally, one may also use this union for *all* corrective adaptive logics considered so far. Some abnormalities are logically impossible for certain lower limit logics, but these have no effect on the adaptive logic anyway—see Fact 5.9.8.

Let me summarize. In this section, the basic logics for handling gluts and gaps with respect to one logical symbol were defined, together with all logics that combine those gluts and gaps. For each of these logics, there is an 'appropriate set' of abnormalities. Combining such a logic with the appropriate set of abnormalities and with the Reliability or Minimal Abnormality strategy results in an adaptive logic in standard format. Note that $\mathbf{CLuN}^m$ and $\mathbf{CLuN}^r$ are such adaptive logics. There are many more and in view of the obvious naming schema, it is at once clear what is meant by $\mathbf{CLaI}^r$, $\mathbf{CLoNaM}^m$, or $\mathbf{CLo}^r$. These logics may be used as such, but may also serve other functions, as we shall see in the next section.

8.2 Variants and Combinations

This section contains further comments on the adaptive logics presented in the previous section. Three topics will be considered: variants of the lower limit logics, including variants analogous to the inconsistency-adaptive variants described in Sections 7.4 and 7.5, choosing among the adaptive logics from the previous section for handling a given premise set, and combining the adaptive logics. Some of the comments remain sketchy because I did not see the point of describing them in more detail. Either the matter is obvious, or the elaboration does not seem to engender any really new features.

The first topic concerns variants on the glut-logics and gap-logics. Four kinds of variants will be briefly considered. A first type concerns the rule of Replacement of Identicals. With the obvious exception of **CL**, no logic presented in the previous section validates this rule. However, all those logics have variants that validate Replacement of Identicals and leave the meaning of all other logical symbols unchanged. The reader may easily construct those variants by comparing **CLuN** with **CLuNs** from Section 7.2. For a different approach and some more variation, I refer the reader to [VBC0x].[6]

A very different kind of enrichment is related to the reduction of complex expressions containing gappy or glutty symbols to simpler such expressions. It is easy enough for the reader to devise the logic **CLaNs**, which relates to **CLaN** in the same way as **CLuNs** relates to **CLuN**; similarly for **CLoNs**. It is not difficult to find similar axiom schemas, and semantic clauses, for other logical symbols. Take implication. Among the obvious candidates, in which I use at once classical logical symbols for the sake of generality, are such equivalences as $(A \supset (A \supset B)) \check{\equiv} (A \supset B)$, $((A \vee B) \supset C) \check{\equiv} ((A \supset C) \check{\wedge} (B \supset C))$, and $(A \supset (B \wedge C)) \check{\equiv} ((A \supset B) \check{\wedge} (A \supset C))$. There is no need to spell all this out here.

The third kind of variant concerns the identification of abnormal objects. This is handled in a way fully parallel to what was said in Section 7.5, except that one starts from the abnormalities of the logic one wants to enrich, rather than from existentially quantified contradictions. The matter is completely straightforward. The fourth kind of variants is analogous to the enrichment discussed in Section 7.4. This too is a rather obvious exercise. The basic

[6]The central point of that paper is that all those logics can be faithfully embedded in **CL**, a fact which has dramatic consequences for the application of partial decision methods.

change needed is that the table for $\mathfrak{a}$-formulas and $\mathfrak{b}$-formulas has to be adjusted to the specific gluts or gaps of the enriched logic. If, for example, this is an adaptive logic that has **CLuM** as its lower limit, then $p \supset q$ will be a member of $\mathrm{sp}((p \wedge r) \supset q)$.

Let us move to the second topic: choosing among the adaptive logics from the previous section for handling a given premise set. I have commented upon this choice in the previous section. Here, my main aim is to show that the dynamic proofs may help one to pick the right choice. The idea is to start with a $\mathbf{CLo}^m$-proof. Let us consider a simple example: $\Gamma_1 = \{p, r, \neg q \vee \neg r, (p \wedge r) \supset q, \neg p \vee s\}$. I introduce the classical symbols step by step in order to make the proof fully transparent.

1	p	Premise	$\emptyset$	
2	r	Premise	$\emptyset$	
3	$\neg q \vee \neg r$	Premise	$\emptyset$	
4	$(p \wedge r) \supset q$	Premise	$\emptyset$	
5	$\neg p \vee s$	Premise	$\emptyset$	
6	$\neg q \check{\vee} \neg r$	3; RC	$\{\check{\neg}(\neg q \check{\vee} \neg r) \check{\wedge} (\neg q \vee \neg r)\}$	$\checkmark^{16}$
7	$\check{\neg}\neg r$	2; RC	$\{\check{\neg}\check{\neg} r \check{\wedge} \neg r\}$	$\checkmark^{16}$
8	$\neg q$	6, 7; RU	$\{\check{\neg}(\neg q \check{\vee} \neg r) \check{\wedge} (\neg q \vee \neg r), \check{\neg}\check{\neg} r \check{\wedge} \neg r\}$	$\checkmark^{16}$
9	$\neg p \check{\vee} s$	5; RC	$\{\check{\neg}(\neg p \check{\vee} s) \check{\wedge} (\neg p \vee s)\}$	
10	$\check{\neg}\neg p$	1; RC	$\{\check{\neg}\check{\neg} p \check{\wedge} \neg p\}$	
11	s	9, 10; RU	$\{\check{\neg}(\neg p \check{\vee} s) \check{\wedge} (\neg p \vee s), \check{\neg}\check{\neg} p \check{\wedge} \neg p\}$	
12	$p \wedge r$	1, 2; RC	$\{(p \check{\wedge} r) \check{\wedge} \check{\neg}(p \wedge r)\}$	$\checkmark^{16}$
13	$(p \wedge r) \check{\supset} q$	4; RC	$\{\check{\neg}((p \wedge r) \check{\supset} q) \check{\wedge} ((p \wedge r) \supset q)\}$	$\checkmark^{16}$
14	q	12, 13; RU	$\{\check{\neg}((p \wedge r) \check{\supset} q) \check{\wedge} ((p \wedge r) \supset q),$ $(p \check{\wedge} r) \check{\wedge} \check{\neg}(p \wedge r)\}$	$\checkmark^{16}$
15	$\check{\neg} q$	8; RC	$\{\check{\neg}(\neg q \check{\vee} \neg r) \check{\wedge} (\neg q \vee \neg r), \check{\neg}\check{\neg} r \check{\wedge} \neg r,$ $\check{\neg}\check{\neg} q \check{\wedge} \neg q\}$	$\checkmark^{16}$
16	$(\check{\neg}((p \wedge r) \check{\supset} q) \check{\wedge} ((p \wedge r) \supset q)) \check{\vee} ((p \check{\wedge} r) \check{\wedge} \check{\neg}(p \wedge r)) \check{\vee}$ $(\check{\neg}(\neg q \check{\vee} \neg r) \check{\wedge} (\neg q \vee \neg r)) \check{\vee} (\check{\neg}\check{\neg} r \check{\wedge} \neg r) \check{\vee} (\check{\neg}\check{\neg} q \check{\wedge} \neg q)$	14, 15; RD	$\emptyset$	

The proof is constructed in such a way that a single abnormality is added to the condition of every line at which RC is applied. These abnormalities are a disjunction glut at lines 6 and 9, a negation glut at lines 7, 10 and 15, a conjunction gap at line 12, and an implication glut at line 13. At line 16 I use the derived rule RD, which was introduced in Section 4.4.

The last example proof provides us with an analysis of the situation: the trouble is caused by the conditional $\mathbf{CLo}^m$-derivability of both q and $\check{\neg} q$. Moreover, it is obvious which gluts and gaps cause the conditional derivability of q and $\check{\neg} q$. By choosing a lower limit which is stronger than **CLo**, and the set of abnormalities suitable for this lower limit, it is possible to obtain a stronger final consequence set.

The above $\mathbf{CLo}^m$-proof is easily transformed to a proof in terms of any of the adaptive logics mentioned in the previous paragraph. To illustrate this, and to illustrate at once the point from the previous paragraph, consider first the familiar adaptive logic $\mathbf{CLuN}^m$. The difference between the $\mathbf{CLo}^m$-proof and the $\mathbf{CLuN}^m$-proof is simply that all gluts and gaps are removed from the conditions of the lines as well as from the only *Dab*-formula derived in the proof. Here is the so obtained proof.

1	p	Premise	$\emptyset$	
2	r	Premise	$\emptyset$	
3	$\neg q \vee \neg r$	Premise	$\emptyset$	
4	$(p \wedge r) \supset q$	Premise	$\emptyset$	
5	$\neg p \vee s$	Premise	$\emptyset$	
6	$\neg q \check{\vee} \neg r$	3; RU	$\emptyset$	
7	$\check{\neg}\neg r$	2; RC	$\{\check{\neg}\check{\neg} r \check{\wedge} \neg r\}$	$\checkmark^{16}$
8	$\neg q$	6, 7; RU	$\{\check{\neg}\check{\neg} r \check{\wedge} \neg r\}$	$\checkmark^{16}$
9	$\neg p \check{\vee} s$	5; RU	$\emptyset$	
10	$\check{\neg}\neg p$	1; RC	$\{\check{\neg}\check{\neg} p \check{\wedge} \neg p\}$	
11	s	9, 10; RU	$\{\check{\neg}\check{\neg} p \check{\wedge} \neg p\}$	
12	$p \wedge r$	1, 2; RU	$\emptyset$	
13	$(p \wedge r) \check{\supset} q$	4; RU	$\emptyset$	
14	q	12, 13; RU	$\emptyset$	
15	$\check{\neg} q$	8; RC	$\{\check{\neg}\check{\neg} r \check{\wedge} \neg r, \check{\neg}\check{\neg} q \check{\wedge} \neg q\}$	$\checkmark^{16}$
16	$(\check{\neg}\check{\neg} r \check{\wedge} \neg r) \check{\vee} (\check{\neg}\check{\neg} q \check{\wedge} \neg q)$	14, 15; RD	$\emptyset$	

It is useful to check the way in which the present proof is a transformation of the preceding one. To maximally retain the parallelism, I did not remove the lines at which classical disjunction and classical implication are introduced. These are useless but cause no harm. Apart from the announced deletion of certain formulas from the conditions and the *Dab*-formula, the only change is that RC is replaced by RU where no $\mathbf{CLuN}^m$-abnormality is introduced. Note that the occurrence of a classical contradiction still leads to the *Dab*-formula 16.

There is a gain in the last example proof in comparison to the $\mathbf{CLo}^m$-proof: q is finally derivable. It is easy enough to choose an adaptive logic from the previous section that provides us with the opposite gain: that $\neg q$ as well as $\check{\neg} q$ are finally derivable. Moreover, the $\mathbf{CLo}^m$-proof shows us the way. One possibility is to allow only for conjunction gaps, in other words, to choose the adaptive logic $\mathbf{CLaC}^m$. The proof then goes as follows.

1	p	Premise	$\emptyset$	
2	r	Premise	$\emptyset$	
3	$\neg q \vee \neg r$	Premise	$\emptyset$	
4	$(p \wedge r) \supset q$	Premise	$\emptyset$	
5	$\neg p \vee s$	Premise	$\emptyset$	
6	$\neg q \check{\vee} \neg r$	3; RU	$\emptyset$	
7	$\check{\neg}\neg r$	2; RU	$\emptyset$	
8	$\neg q$	6, 7; RU	$\emptyset$	
9	$\neg p \check{\vee} s$	5; RU	$\emptyset$	
10	$\check{\neg}\neg p$	1; RU	$\emptyset$	
11	s	9, 10; RU	$\emptyset$	
12	$p \wedge r$	1, 2; RC	$\{(p \check{\wedge} r) \check{\wedge} \check{\neg}(p \wedge r)\}$	$\checkmark^{16}$
13	$(p \wedge r) \check{\supset} q$	4; RU	$\emptyset$	
14	q	12, 13; RU	$\{(p \check{\wedge} r) \check{\wedge} \check{\neg}(p \wedge r)\}$	$\checkmark^{16}$
15	$\check{\neg} q$	8; RU	$\emptyset$	
16	$(p \check{\wedge} r) \check{\wedge} \check{\neg}(p \wedge r)$	14, 15; RD	$\emptyset$	

Nearly the same effect is obtained by choosing $\mathbf{CLuM}^m$, which allows only for implication gluts. In that proof, $\check{\neg}((p \wedge r) \check{\supset} q) \check{\wedge} ((p \wedge r) \supset q)$ is the formula of line 16 and the singleton comprising this formula is the condition of lines 13 and 14, whence these lines are marked.

In the next to last proof, $\mathbf{CLuN}^m$ gives one q as an unconditional consequence. This is also the case if one chooses the logic $\mathbf{CLuD}^m$, which allows for disjunction gluts only. Moreover, the $\mathbf{CLo}^m$-proof reveals that this is a secure choice. Indeed, allowing for disjunction gluts causes $\check{\neg} q$ not to be a final consequence of the premise set. So this avoids triviality.

What happens if one chooses the adaptive logic $\mathbf{CLaN}^m$? All conditions become empty, so q and $\check{\neg} q$ are derived unconditionally and RD cannot be applied. Put differently, the formula of line 16 is turned into the empty string by removing all 'abnormalities' that are not negation gaps. However, as we derived a classical inconsistency, q and $\check{\neg} q$, and we derived it on the empty condition, we obtain triviality. So $\mathbf{CLaN}^m$ does not lead to a minimally abnormal 'interpretation' of Γ_1.

It should be clear by now that $\mathbf{CLo}^m$ proofs offer an instrument for obtaining the minimally abnormal interpretations of premise sets. Suppose that *Dab*-formulas are only introduced by RD, as I advised some paragraphs ago. If no *Dab*-formulas are derived in the $\mathbf{CLo}^m$ proof, the premise set is apparently normal.[7] If that is so, its interpretation in terms of **CL** is normal. If *Dab*-formulas are derived, a minimally abnormal interpretation of the premises is obtained by choosing a lower limit that does not turn any *Dab*-formula into the empty string. Note that some of these lower limit logics may combine different gluts and gaps. The matter is completely straightforward. We can read off the minimally abnormal interpretations from the $\mathbf{CLo}^m$ proof. In sum, constructing proofs in $\mathbf{CLo}^m$ (or $\mathbf{CLo}^r$) offers an analysis that allows one to decide which adaptive logics from the previous section may be applied to handle a given premise set, and which may not because they assign a trivial consequence set to the premise set. The analysis also reveals which adaptive logics offer a richer consequence set than others.

The logic $\mathbf{CLo}^m$ is also interesting in itself for a theoretical reason. Indeed, in this logic, the meaning of all standard logical symbols is *contingent*: the meaning of an occurrence of a standard symbol, and these are the only ones that should occur in the premises and the (main) conclusion, depends fully on the premise set. To put it in a pompous way: $\mathbf{CLo}^m$ provides one with a formal hermeneutics—but see Section 8.4 for a more impressive result in this respect.

The story does not end here. Until now I have considered logics from the previous section and have illustrated the way in which they lead to different non-trivial but inconsistent 'interpretations' of an inconsistent theory. However, the logics from the previous section may, in a specific sense, also be combined in the sense of Chapter 6. I shall illustrate that this leads to further non-trivial but inconsistent 'interpretations' of an inconsistent theory. This approach requires some clarification before we start.

Let us consider the premise set $\Gamma_2 = \{p, r, (p \vee q) \supset s, (p \vee t) \supset \neg r, (p \wedge r) \supset \neg s, (p \wedge s) \supset t\}$. I shall not write out the $\mathbf{CLo}^m$-proof, but if one writes it out, one readily sees that Γ_2 can be interpreted non-trivially by allowing for disjunction gaps as well as for conjunction gaps. The $\mathbf{CLo}^m$-proof moreover reveals that it may be interesting to first eliminate the disjunction gaps and

[7] I write "apparently" because the judgement concerns only the present stage of the $\mathbf{CLo}^m$ proof.

next the conjunction gaps, something which typically may be realized by a combined adaptive logic. The question is what this combined logic precisely looks like.

The combination $Cn^{\mathcal{L}_s}_{\mathbf{CLaC}^m}(Cn^{\mathcal{L}_S}_{\mathbf{CLaD}^m}(\Gamma_2))$ would not have the desired effect. One the one hand, every conjunction of members of $Cn^{\mathcal{L}_S}_{\mathbf{CLaD}^m}(\Gamma_2)$ is itself a member of that set because the standard conjunction behaves like the classical conjunction in **CLaD**. So closing $Cn^{\mathcal{L}_S}_{\mathbf{CLaD}^m}(\Gamma_2)$ under $\mathbf{CLaC}^m$ does not add any conjunctions. On the other hand, the standard *disjunction* behaves like the classical disjunction in **CLaC**. This means that if $A \in \Gamma$ and hence $A \in Cn^{\mathcal{L}_S}_{\mathbf{CLaD}^m}(\Gamma_2)$, then $A \vee B \in Cn^{\mathcal{L}_s}_{\mathbf{CLaC}^m}(Cn^{\mathcal{L}_S}_{\mathbf{CLaD}^m}(\Gamma_2))$ for every B. This may very well cause triviality. The reader may easily verify this by reinterpreting the subsequent proof from Γ_2 as a proof for $Cn^{\mathcal{L}_s}_{\mathbf{CLaC}^m}(Cn^{\mathcal{L}_S}_{\mathbf{CLaD}^m}(\Gamma_2))$.[8]

What we need is rather obvious. We want to superimpose two simple adaptive logics that allow for disjunction gaps as well as for conjunction gaps, but we want first to minimize the set of disjunction gaps and only thereafter the set of conjunction gaps. So, following the naming scheme from the previous section, we first need an adaptive logic composed of the lower limit logic **CLaDaC**, the set of abnormalities $\Omega = \{\check{\exists}((A \check{\vee} B) \check{\wedge} \check{\neg}(A \vee B)) \mid A, B \in \mathcal{F}_s\})$, comprising the disjunction gaps, and say Minimal Abnormality. One might call this logic $\mathbf{CLaDaC}^m_{aD}$. Next, we want to close the consequence set of this logic by an adaptive logic composed of the lower limit logic **CLaDaC**, the set of abnormalities $\Omega = \{\check{\exists}((A \check{\wedge} B) \check{\wedge} \check{\neg}(A \wedge B)) \mid A, B \in \mathcal{F}_s\}$, comprising the conjunction gaps, and Minimal Abnormality. One might call this logic $\mathbf{CLaDaC}^m_{aC}$.

Let us move to the proof in this combined logic. All logical symbols have their classical meaning with the exception of disjunction and conjunction. The reader should remember from Chapter 6 that the first round of marking proceeds in terms of the minimal *Dab*-formulas that have disjunction gaps as their disjuncts and are derived on the empty condition, whereas the second round proceeds in terms of the minimal *Dab*-formulas that have conjunction gaps as their disjuncts and are derived at an unmarked line the condition of which may contain disjunction gaps. I try to make the proof more transparent by first deriving the required disjunctions, applying $\mathbf{CLaDaC}^m_{aD}$, and only thereafter deriving the required conjunctions by applying $\mathbf{CLaDaC}^m_{aC}$.

1	p	Premise	$\emptyset$	
2	r	Premise	$\emptyset$	
3	$(p \vee q) \supset s$	Premise	$\emptyset$	
4	$(p \vee t) \supset \neg r$	Premise	$\emptyset$	
5	$(p \wedge r) \supset \neg s$	Premise	$\emptyset$	
6	$(p \wedge s) \supset t$	Premise	$\emptyset$	
7	$p \vee q$	1; RC	$\{(p \check{\vee} q) \check{\wedge} \check{\neg}(p \vee q)\}$	
8	s	3, 7; RU	$\{(p \check{\vee} q) \check{\wedge} \check{\neg}(p \vee q)\}$	
9	$p \vee t$	1; RC	$\{(p \check{\vee} t) \check{\wedge} \check{\neg}(p \vee t)\}$	$\checkmark^{11}$
10	$\neg r$	4, 9; RU	$\{(p \check{\vee} t) \check{\wedge} \check{\neg}(p \vee t)\}$	$\checkmark^{11}$
11	$(p \check{\vee} t) \check{\wedge} \check{\neg}(p \vee t)$	2, 10; RD	$\emptyset$	
12	$p \wedge r$	1, 2; RC	$\{(p \check{\wedge} r) \check{\wedge} \check{\neg}(p \wedge r)\}$	$\checkmark^{14}$
13	$\neg s$	5, 12; RU	$\{(p \check{\wedge} r) \check{\wedge} \check{\neg}(p \wedge r)\}$	$\checkmark^{14}$

[8]The disjunction $p \vee t$ is **CLaC**-derivable from p and hence is derivable on the empty condition in the so reinterpreted proof. But then so are both r and $\neg r$, whence triviality results.

14	$(p \check{\wedge} r) \check{\wedge} \check{\neg}(p \wedge r)$	8, 13; RD	$\{(p \check{\vee} q) \check{\wedge} \check{\neg}(p \vee q)\}$
15	$p \wedge s$	1, 8; RU	$\{(p \check{\vee} q) \check{\wedge} \check{\neg}(p \vee q), (p \check{\wedge} s) \check{\wedge} \check{\neg}(p \wedge s)\}$
16	t	6, 15; RU	$\{(p \check{\vee} q) \check{\wedge} \check{\neg}(p \vee q), (p \check{\wedge} s) \check{\wedge} \check{\neg}(p \wedge s)\}$

On line 14, the general form of rule RD is applied. The set of consequences of the combined logic can be 'summarized' as $\{p, r, s, t, \neg(p \wedge r), \neg(p \vee t)\}$. Note that I write classical negation in the abnormalities in the proof to be coherent with the rest of this chapter, but that the standard negation has the same meaning. The same result cannot be obtained by any of the logics described in the previous section. By using the superposition combination, $(p \check{\vee} q) \check{\wedge} \check{\neg}(p \vee q)$ is not a disjunct of a minimal *Dab*-consequence of the premises, whereas it is for example in $\mathbf{CLuCaD}^m$.

There may be specific logical or extra-logical reasons to 'interpret' Γ_2 in terms of the combined adaptive logic. As mentioned before, such reasons may become apparent, or the interpretation may be seen as a sensible alternative, in view of a $\mathbf{CLo}^m$-proof from Γ_2. Obviously, this is only an example. However, the example shows the pattern to be followed: select the abnormalities one needs or wants to allow for; choose a lower limit logic that allows for precisely these abnormalities and combine them with the chosen strategy; finally, choose an ordering of the abnormalities and superimpose the simple adaptive logics in that order.

The upper limit logic of all simple adaptive logics presented in this chapter is **CL**. So these logics, and all the combined adaptive logics built from them, assign the same consequence set as **CL** to all premise sets that have **CL**-models. While this is an interesting feature in itself, the interest of the diversity of the logics lies with premise sets that have no **CL**-models.

8.3 Ambiguity-Adaptive Logics

In [Van97], Guido Vanackere presented the first ambiguity-adaptive logic. The underlying idea is simple but ingenious. The inconsistency of a text may derive from the ambiguity of its non-logical symbols. To take these possible ambiguities into account, one *indexes* all occurrences of non-logical symbols. An ambiguity-adaptive logic interprets a set of premises as unambiguous as possible. This is realized by presupposing that two occurrences of a non-logical symbol have the same meaning unless and until proven otherwise.

While the idea is simple and attractive, elaborating the technical details requires hard work. Most published papers on ambiguity-adaptive logics evade some unsolved problems. There is a reason why the matter is confusing. The languages underlying ambiguity-adaptive logics may serve diverse, unexpected, and attractive purposes. All purposes require a monotonic logic that is close to **CL**, but many purposes demand that the logic deviate from **CL** in one or other detail, and each purpose requires a different deviation. I now spell out a systematic and sensible variant of ambiguity logic.

In the language $\mathcal{L}_s$, the sets of schematic letters[9] for non-logical symbols are $\mathcal{S}$, $\mathcal{C}$, $\mathcal{V}$, and $\mathcal{P}^r$ (for each rank $r \in \mathbb{N}$). Let us replace each of these sets with a set of indexed letters, which comprise the letters from the original sets with a

[9]The name "letter" is slightly misleading. Most schematic letters are actually strings composed from a finite sequence of symbols.

superscripted index $i \in \mathbb{N}$ attached to them. Thus $\mathcal{S}^I = \mathcal{S} \cup \{\lambda^i \mid \lambda \in \mathcal{S}; i \in \mathbb{N}\}$, and similarly for $\mathcal{C}^I$, $\mathcal{V}^I$, and $\mathcal{P}^{rI}$. The resulting sets are still denumerable. From these sets we define a language $\mathcal{L}^I_s$, with $\mathcal{F}^I_s$ as its set of formulas and $\mathcal{W}^I_s$ as its set of closed formulas. The language $\mathcal{L}^I_s$ is exactly as one expects, *except that the quantifiers still range over the variables of* $\mathcal{L}_s$. The reasons for this convention will be explained later on.

Next, we define a logic **CLI** over this language. The logic is almost identical to **CL**, except for the way in which quantified formulas are handled. To phrase the semantics, we need to add an indexed set $\mathcal{O}^I$ of pseudo-constants, which is defined from $\mathcal{O}$ in the same way as $\mathcal{C}^I$ is defined from $\mathcal{C}$. The resulting pseudo-language $\mathcal{L}^I_{\mathcal{O}}$ has $\mathcal{W}^I_{\mathcal{O}}$ as its set of closed formulas. A **CLI**-model $M = \langle D, v \rangle$, in which D is a set and v is an assignment function. The function v is like for **CL**, except that the indexed sets are interpreted.

C1 $v\colon \mathcal{W}^I_{\mathcal{O}} \rightarrow \{0,1\}$
C2 $v\colon \mathcal{C}^I \cup \mathcal{O}^I \rightarrow D$ (where $D = \{v(\alpha) \mid \alpha \in \mathcal{C}^I \cup \mathcal{O}^I\}$)
C3 $v\colon \mathcal{P}^{rI} \rightarrow \wp(D^r)$

The valuation function $v_M\colon \mathcal{W}^I_{\mathcal{O}} \rightarrow \{0,1\}$ determined by M is defined as follows:

C$\mathcal{S}^I$ where $A \in \mathcal{S}^I$, $v_M(A) = 1$ iff $v(A) = 1$
C$\mathcal{P}^{rI}$ where $\pi^r \in \mathcal{P}^{rI}$ and $\alpha_1 \ldots \alpha_r \in \mathcal{C}^I \cup \mathcal{O}^I$,
$v_M(\pi^r\alpha_1 \ldots \alpha_r) = 1$ iff $\langle v(\alpha_1), \ldots, v(\alpha_r)\rangle \in v(\pi^r)$
C= $v_M(\alpha = \beta) = 1$ iff $v(\alpha) = v(\beta)$
C¬ $v_M(\neg A) = 1$ iff $v_M(A) = 0$
C⊃ $v_M(A \supset B) = 1$ iff $v_M(A) = 0$ or $v_M(B) = 1$
C∧ $v_M(A \wedge B) = 1$ iff $v_M(A) = 1$ and $v_M(B) = 1$
C∨ $v_M(A \vee B) = 1$ iff $v_M(A) = 1$ or $v_M(B) = 1$
C≡ $v_M(A \equiv B) = 1$ iff $v_M(A) = v_M(B)$
C$\forall^I$ $v_M(\forall\alpha A(\alpha^{i_1}, \ldots, \alpha^{i_n})) = 1$ iff $\{v_M(A(\beta^{i_1}, \ldots, \beta^{i_n})) \mid \beta \in \mathcal{C} \cup \mathcal{O}\} = \{1\}$
C$\exists^I$ $v_M(\exists\alpha A(\alpha^{i_1}, \ldots, \alpha^{i_n})) = 1$ iff $1 \in \{v_M(A(\beta^{i_1}, \ldots, \beta^{i_n})) \mid \beta \in \mathcal{C} \cup \mathcal{O}\}$

$M \Vdash A$ iff $v_M(A) = 1$, which defines $\vDash_{\mathbf{CLI}} A$ as well as $\Gamma \vDash_{\mathbf{CLI}} A$.

The clauses C∀ and C∃ deserve some clarification. Note that the quantifiers range over α and that the α^{i_j} are indexed occurrences of this variable in A. As was agreed before, quantifiers range over members of $\mathcal{V}$ whereas the variables that occur in members of $\mathcal{F}^I_{\mathcal{O}}$ are members of $\mathcal{V}^I$. Thus $M \Vdash \forall x(P^1x^1 \supset Q^1x^2)$ holds iff $M \Vdash P^1\alpha^1 \supset Q^1\alpha^2$ holds for all $\alpha \in \mathcal{C} \cup \mathcal{O}$. Similarly, $M \Vdash \exists x(P^1x^1 \wedge Q^1x^2)$ holds iff $M \Vdash P^1\alpha^1 \wedge Q^1\alpha^2$ holds for some $\alpha \in \mathcal{C} \cup \mathcal{O}$.

The behaviour of the quantifiers causes a connection between variables that differ only from each other in their index, because the same quantifiers bind them all. The quantifiers also connect indexed variables to the constants with the same indices. Thus, among the semantic consequences of $\forall x(P^1x^1 \supset Q^1x^2)$ are $P^1a^1 \supset Q^1a^2$ as well as $P^1b^1 \supset Q^1b^2$, but not, for example, $P^1a^1 \supset Q^1b^2$ or $P^1a^1 \supset Q^1a^3$. We shall see later that this peculiar logic is tailored in order to suit the ambiguity-adaptive logic of which it is the lower limit.

I leave it as an easy exercise for the reader to spell out an axiomatic system for **CLI**. Hint: take the **CL**-axiomatization from Section 1.7, letting the metavariables range over indexed entities; next adjust A∀ to $\forall\alpha A(\alpha^{i_1}, \ldots, \alpha^{i_n})) \supset A(\beta^{i_1}, \ldots, \beta^{i_n})$, and adjust A∃, R∀, and R∃ similarly.

The idea of (non-adaptive) ambiguity logics is that, where $\Gamma \subseteq \mathcal{W}_s$ and $A \in \mathcal{W}_s$, $\Gamma \vdash A$ iff a certain translation of A is a **CLI**-consequence of a certain translation of Γ. The presumably unexpected handling of the quantifiers will be easier understood after I presented the translation. Let $\Gamma^\dagger$ be obtained from Γ by adding superscripted indices from an $I \subset \mathbb{N}$ to all non-logical symbols in Γ in such a way that every index occurs at most once. Next, let $A^\ddagger$ be obtained from A by adding superscripted indices from $\mathbb{N} - I$ to all non-logical symbols in A in such a way that every index occurs at most once.[10] The ambiguity logic **CLA**, defined over the language $\mathcal{L}_s$, is defined by

$$\Gamma \vdash_{\mathbf{CLA}} A \text{ iff } \Gamma^\dagger \vdash_{\mathbf{CLI}} A^\ddagger\,.$$

In order to define **CLA**, we need only a certain fragment of **CLI**. For every premise set Γ and conclusion A, $\Gamma^\dagger \cup \{A^\ddagger\}$ is a set of members of $\mathcal{W}_s^I$ that has a very specific property: all non-logical symbols are indexed and no two occurrences of the same non-logical symbol have the same index. One of the effects of this is that there are no Γ and A for which $\Gamma^\dagger \vdash_{\mathbf{CLI}} A^\ddagger$, whereas there obviously are Γ and A for which $\Gamma \vdash_{\mathbf{CLI}} A$, for example $p^1 \wedge q^2 \vdash_{\mathbf{CLI}} p^1$.

At this point, the handling of the quantifiers should be more transparent. We have $A \wedge B \nvdash_{\mathbf{CLA}} A$. For example, $p \wedge q \nvdash_{\mathbf{CLA}} p$ because $p^1 \wedge q^2 \nvdash_{\mathbf{CLI}} p^3$—in some **CLI**-models $v(p^1) = v(q^2) = 1$ and $v(p^3) = 0$. But consider $\forall x\, x = x \wedge q \vdash_{\mathbf{CLA}} \forall x\, x = x$. If the quantifiers ranged over the indexed variables, this would come out true because $\forall x^1\, x^1 = x^1 \wedge q^2 \vDash_{\mathbf{CLI}} \forall x^3\, x^3 = x^3$.[11] But then quantified statements would behave oddly, because they would form classical exceptions in the ambiguity logic.

Let us take a closer look at this. The point is actually related to theorems of logic. Thus $\nvdash_{\mathbf{CLA}} p \vee \neg p$ because $\nvdash_{\mathbf{CLI}} p^1 \vee \neg p^2$. In general, **CLA** does not have any theorems at the propositional level. Note that the absence of theorems derives from the *translation*, not from **CLI**, which obviously has all the right theorems, for example $\vdash_{\mathbf{CLI}} p^1 \vee \neg p^1$. When one moves to the predicative level, **CL**-theorems turn out to be non-theorems of **CLA**. For example $\nvdash_{\mathbf{CLA}} a = a$ because $\nvDash_{\mathbf{CLI}} a^1 = a^2$ and $\nvdash_{\mathbf{CLA}} \forall x Px \supset Pa$ because $\nvdash_{\mathbf{CLI}} \forall x P^1x^2 \supset P^3a^4$—note that, even if the quantifiers ranged over the indexed variables, we would still have $\nvdash_{\mathbf{CLI}} \forall x^2 P^1x^2 \supset P^3a^4$. However, if the quantifiers ranged over indexed variables, we would have $\vdash_{\mathbf{CLA}} \forall x\, x = x$ because $\vdash_{\mathbf{CLI}} \forall x^1\, x^1 = x^1$—remember that $\forall x^1\, x^1 = x^2$ is not a closed formula. So this would reintroduce logical theorems at a unique specific point, which would be an anomaly. Actually, letting the quantifiers range over the original variables causes no trouble, as the **CLI**-semantics reveals. Indeed, there are **CLI**-models that verify $\forall x\, x^1 = x^2$, and there are that do not, just as we want it. So $\nvdash_{\mathbf{CLA}} \forall x\, x = x$. Similarly $\forall x\, x = x \nvdash_{\mathbf{CLA}} \forall x\, x = x$ because $\forall x\, x^1 = x^2 \nvdash_{\mathbf{CLI}} \forall x\, x^3 = x^4$.

The reader may think that another approach is equally sensible: to let the quantifiers range over indexed variables while multiplying the quantifiers where this is necessary to obtain closed formulas. Thus the translation of $\forall x\, x = x$ would be, for example, $\forall x^1 \forall x^2\, x^1 = x^2$. This, however, would not work. Indeed, from this formula, one might first obtain $\forall x^2\, a^1 = x^2$ and next $a^1 = b^2$, which

[10] Other ways of indexing are equally adequate. As explained below in the text, every two occurrences of the same symbol in $\Gamma \cup \{A\}$ should have different indices and no individual variable should have the same index as an individual constant.

[11] If the quantifiers range over indexed variables, $\forall x^1\, x^1 = x^2$ is not a closed formula.

would blur the difference between two very different formulas, $\forall x\, x = x$ and $\forall x \forall y\, x = y$. By letting the quantifiers range over the non-indexed variables, we guarantee that all indexed occurrences of the same variable are instantiated at the same time.

The logic **CLA** is intriguing. Nothing is valid in it, nothing is derivable from any premise set. Post-modernists should be pleased. Sensible people, however, will regard **CLA** as a lower limit logic, and will try to minimize abnormalities. They will admit that some texts (or premise sets) force one to consider some non-logical terms as ambiguous,[12] but they will also stress that non-logical terms have to be considered unambiguous "unless and until proven otherwise". In other words, they will go adaptive.

It is not difficult to see what going adaptive comes to. The lower limit logic will be **CLI** and the strategy either Reliability or Minimal Abnormality. We need a set of abnormalities containing three kinds of formulas: ambiguities pertaining respectively to sentential letters, to individual constants and variables, and to predicative letters. In order to save some space in the examples proofs, I shall introduce abbreviations for each of these kinds of abnormalities. Ambiguities for sentential letters have the form $\neg(A^i \equiv A^j)$, with $A \in \mathcal{S}$ and $i, j \in \mathbb{N}$.[13] These will be abbreviated as $A^{i \cdot j}$, for example $p^{5 \cdot 8}$ abbreviates $\neg(p^5 \equiv p^8)$. Ambiguities for individual constants and variables will have the form $\exists \neg\, \alpha^i = \alpha^j$, with $\alpha \in \mathcal{C} \cup \mathcal{V}$ and $i, j \in \mathbb{N}$. These will be abbreviated as $\alpha^{i \cdot j}$, for example $a^{6 \cdot 7}$ abbreviates $\neg\, a^6 = a^7$ and $x^{4 \cdot 8}$ abbreviates $\exists x \neg\, x^4 = x^8$. Finally, ambiguities for predicative letters have the form $\exists \neg(\pi^i \alpha_1 \ldots \alpha_r \equiv \pi^j \alpha_1 \ldots \alpha_r)$, with $\pi \in \mathcal{P}^r$, $i, j \in \mathbb{N}$, and $\alpha_1 \ldots \alpha_r \in \mathcal{V}$. These will be abbreviated as $\pi^{i \cdot j} \alpha_1 \ldots \alpha_r$, for example, where $P^{3 \cdot 5} x^1$ abbreviates $\exists x \neg(P^3 x^1 \equiv P^5 x^1)$ and, where $R \in \mathcal{P}^3$, $R^{2 \cdot 8} a^1 x^1 b^2$ abbreviates $\exists x \neg(R^2 a^1 x^1 b^2 \equiv R^8 a^1 x^1 b^2)$.[14]

The meaning of the abnormalities requires hardly any clarification: different occurrences of a symbol have different meanings. This is straightforward for sentential letters, individual constants and individual variables. There is a difference, however. Occurrences of the same constant may have different denotations. So it is possible that $\neg\, a^1 = a^2$, $\neg\, a^1 = a^3$, and $\neg\, a^2 = a^3$, and so on for any number of occurrences of the same constant. The matter is different for propositional letters. As there are (on the present approach) only two truth-values, 0 and 1, the occurrence of p^1, p^2 and p^3 necessarily leads to $p^1 \equiv p^2$, to $p^3 \equiv p^1$, or to $p^3 \equiv p^2$. The case of predicative letters is slightly more sophisticated. If both $P^1 a^2$ and $\neg P^3 a^2$ hold true, the object denoted by a^2 belongs to the extension of P^1 but not to that of P^3. In other words, P^1 and P^3 differ in extension with respect to the object denoted by a^2. If moreover both $P^1 a^4$ and $\neg P^3 a^4$ hold true, there is a further ambiguity: P^1 and P^3 also differ in extension with respect to the object denoted by a^4. This is the reason why abnormalities pertaining to predicates require a more complex abbreviation than the other abnormalities.

It is time to formally state the set of abnormalities. I shall do this in terms

[12] I obviously suppose here that the *logical* symbols have a unique and stable meaning.

[13] If the intention is to combine ambiguity logics with logics from Sections 8.1 or 8.2, the abnormalities are better phrased with the help of classical logical symbols.

[14] The use of ambiguities in the variables is illustrated by $\forall x(P^1 x^2 \equiv P^3 x^4) \vdash_{\mathbf{CLI}} \exists x \neg(x^2 = x^4) \vee \exists x \neg(P^1 x^2 \equiv P^3 x^2)$. Incidentally, $\neg(p^1 \equiv p^2)$ and $\neg(p^2 \equiv p^1)$ are officially considered as different (but equivalent) abnormalities. Similarly $p^{1 \cdot 2}$ and $p^{2 \cdot 1}$ are officially seen as abbreviations of different formulas. Both decisions are obviously purely conventional.

of the introduced abbreviations: $\Omega = \{A^{i \cdot j} \mid A \in \mathcal{S}; i,j \in \mathbb{N}; i \neq j\} \cup \{\alpha^{i \cdot j} \mid \alpha \in \mathcal{C} \cup \mathcal{V}; i,j \in \mathbb{N}; i \neq j\} \cup \{\pi^{i \cdot j}\alpha_1 \ldots \alpha_r \mid \pi \in \mathcal{P}^r; i,j \in \mathbb{N}; \alpha_1 \ldots \alpha_r \in \mathcal{C}^I \cup \mathcal{V}^I; i \neq j\}$. When reading this, remember that all logical symbols have their classical meaning. The adaptive logics $\mathbf{CLI}^m$ and $\mathbf{CLI}^r$ are now fully defined.

In terms of $\mathbf{CLI}^m$, we define the logic $\mathbf{CLA}^{\mathsf{m}}$:

$$\Gamma \vdash_{\mathbf{CLA}^{\mathsf{m}}} A \text{ iff } \Gamma^{\dagger} \vdash_{\mathbf{CLI}^m} A^{\ddagger} \,,$$

and similarly for $\mathbf{CLA}^{\mathsf{r}}$. I write the superscripts of $\mathbf{CLA}^{\mathsf{m}}$ and $\mathbf{CLA}^{\mathsf{r}}$ in a different type to indicate that these logics are not themselves adaptive logics in standard format, but are characterized in terms of such logics.

Let us consider some example proofs. The premise set $\Gamma_3 = \{\forall x(Px \supset Qx), Pa\}$ is normal. So the $\mathbf{CLA}^{\mathsf{m}}$-consequence set (and $\mathbf{CLA}^{\mathsf{r}}$-consequence set) of Γ_3 is identical to its $\mathbf{CL}$-consequence set, as the reader expected. Here is an example proof of $Qa \in Cn_{\mathbf{CLA}^{\mathsf{m}}}(\Gamma_3)$. This comes to $\forall x(P^1x^2 \supset Q^3x^4), P^5a^6 \vdash_{\mathbf{CLI}^m} Q^7a^8$.

1	$\forall x(P^1x^2 \supset Q^3x^4)$	Prem	$\emptyset$
2	P^5a^6	Prem	$\emptyset$
3	$P^1a^2 \supset Q^3a^4$	1; RU	$\emptyset$
4	P^1a^2	2; RC	$\{P^{1\cdot5}a^6, a^{2\cdot6}\}$
5	Q^3a^4	3, 4; RU	$\{P^{1\cdot5}a^6, a^{2\cdot6}\}$
6	Q^7a^8	5; RC	$\{P^{1\cdot5}a^6, a^{2\cdot6}, Q^{3\cdot7}a^4, a^{4\cdot8}\}$

As $\{\forall x(P^1x^2 \supset Q^3x^4), P^5a^6\}$ is normal with respect to $\mathbf{CLI}^m$, no *Dab*-formula is derivable from it, whence no line is marked in any extension of the proof.

Some readers may find the proof a bit fast. Here is the trick, applied to the transition from 2 to 4. The condition of line 2.1 is the negation of the formula of that line. So the line results from the $\mathbf{CLI}$-theorem $(P^5a^6 \equiv P^1a^6) \vee \neg(P^5a^6 \equiv P^1a^6)$. Similarly for line 2.3, which results from the $\mathbf{CLI}$-theorem $a^2 = a^6 \vee \neg a^2 = a^6$.

2	P^5a^6	Prem	$\emptyset$
2.1	$P^5a^6 \equiv P^1a^6$	RC	$\{P^{1\cdot5}a^6\}$
2.2	P^1a^6	2, 2.1; RU	$\{P^{1\cdot5}a^6\}$
2.3	$a^2 = a^6$	RC	$\{a^{2\cdot6}\}$
4	P^1a^2	2.2, 2.3; RU	$\{P^{1\cdot5}a^6, a^{2\cdot6}\}$

If predicative expressions are ambiguous, the ambiguity can lie with a predicate, an individual constant, or a variable. This often leads to a disjunction of such abnormalities. For example $P^1a^2, \neg P^3a^4 \vdash_{\mathbf{CLI}} \neg a^2 = a^4 \vee \neg(P^1a^4 \equiv P^3a^4)$. This will be illustrated in the next example proof.

It is instructive to consider a further example: $\Gamma_4 = \{\forall x(Px \supset Qx), Pa, \neg Qa, Pb\}$. Its translation is, for example, $\{\forall x(P^1x^2 \supset Q^3x^4), P^5a^6, \neg Q^7a^8, P^9b^{10}\}$. Let us check wether $\Gamma_4 \vdash_{\mathbf{CLA}^{\mathsf{m}}} Qa$ and $\Gamma_4 \vdash_{\mathbf{CLA}^{\mathsf{m}}} Qb$. As the indices 1–10 occur in the translation of Γ_4, the indexed conclusions will be, for example, $Q^{11}a^{12}$ and $Q^{11}b^{12}$ respectively.

1	$\forall x(P^1x^2 \supset Q^3x^4)$	Prem	$\emptyset$
2	P^5a^6	Prem	$\emptyset$
3	$\neg Q^7a^8$	Prem	$\emptyset$
4	P^9b^{10}	Prem	$\emptyset$

5	$P^1a^2 \supset Q^3a^4$	1; RU	$\emptyset$	
6	P^1a^2	2; RC	$\{P^{1\cdot 5}a^6, a^{2\cdot 6}\}$	$\checkmark^{10}$
7	Q^3a^4	5, 6; RU	$\{P^{1\cdot 5}a^6, a^{2\cdot 6}\}$	$\checkmark^{10}$
8	$Q^{11}a^{12}$	7; RC	$\{P^{1\cdot 5}a^6, a^{2\cdot 6}, Q^{3\cdot 11}a^4, a^{4\cdot 12}\}$	$\checkmark^{10}$
9	$\neg Q^3a^4$	3; RC	$\{Q^{7\cdot 3}a^8, a^{8\cdot 4}\}$	$\checkmark^{10}$
10	$P^{1\cdot 5}a^6 \vee a^{2\cdot 6} \vee Q^{7\cdot 3}a^8 \vee a^{8\cdot 4}$	7, 9; RD	$\emptyset$	

Apart from 10, many other *Dab*-formulas are derivable from the proof. For any suitable i and j, Q^ia^j is derivable from Q^3a^4 on the condition $\{P^{1\cdot 5}a^6, a^{2\cdot 6}, Q^{3\cdot i}a^4, a^{4\cdot j}\}$ and $\neg Q^ia^j$ is derivable from Q^7a^8 on the condition $\{Q^{7\cdot i}a^8, a^{8\cdot j}\}$. So the disjunction of members of both conditions is **CLI**-derivable on the empty condition. In whichever way one proceeds, the line at which $Q^{11}a^{12}$ is derived will be marked; $Q^{11}a^{12}$ is not a final $\mathbf{CLI}^m$-consequence of $\{P^{1\cdot 5}a^6, a^{2\cdot 6}, Q^{3\cdot i}a^4, a^{4\cdot j}\}$ and $\Gamma_4 \nvdash_{\mathbf{CLA^m}} Qa$.

The situation is obviously very different for $Q^{11}b^{12}$. Let us have a look at the continuation of the previous proof.

11	$P^1b^2 \supset Q^3b^4$	1; RU	$\emptyset$
12	P^1b^2	4; RC	$\{P^{1\cdot 9}b^{10}, b^{2\cdot 10}\}$
13	Q^3b^4	11, 12; RU	$\{P^{1\cdot 9}b^{10}, b^{2\cdot 10}\}$
14	$Q^{11}b^{12}$	13; RC	$\{P^{1\cdot 9}b^{10}, b^{2\cdot 10}, Q^{3\cdot 11}b^4, b^{4\cdot 12}\}$

None of these lines will be marked in any extension of the proof. The reason is that the conditions of the lines contain only abnormalities that explicitly mention b, whereas no such abnormality is **CLI**-derivable from $\{\forall x(P^1x^2 \supset Q^3x^4), P^5a^6, \neg Q^7a^8, P^9b^{10}\}$. So $Q^{11}b^{12}$ is a final **CLI**-consequence of the translated premise set and $\Gamma_4 \vdash_{\mathbf{CLA^m}} Qb$.

Some readers may wonder why the proofs contain no examples of abnormalities that pertain to variables. This is partly a matter of style. For example, the lines 11–14 of the last proof may just as well be replaced by the following lines in which I also proceed a bit faster.

11	$\forall x(P^9x^{10} \supset Q^{11}x^{12})$	1; RC	$\{P^{1\cdot 9}x^2, x^{2\cdot 10}, Q^{3\cdot 11}x^4, x^{4\cdot 12}\}$
12	$P^9b^{10} \supset Q^{11}b^{12}$	11; RC	$\{P^{1\cdot 9}x^2, x^{2\cdot 10}, Q^{3\cdot 11}x^4, x^{4\cdot 12}\}$
13	$Q^{11}b^{12}$	4, 12; RC	$\{P^{1\cdot 9}x^2, x^{2\cdot 10}, Q^{3\cdot 11}x^4, x^{4\cdot 12}\}$

In other cases, for example in order to establish $\forall x(Px \supset Qx), \forall x(Qx \supset Rx) \vdash_{\mathbf{CLA^m}} \forall x(Px \supset Rx)$, abnormalities pertaining to variables are unavoidable, unless of course when dummy constants are introduced.

Before leaving the matter, two points are worth some attention. The first concerns my promise to clarify the translation, the second concerns variants for the present ambiguity-adaptive logics.

The translation is actually a simple matter. When describing it, I required (in footnote 10) that no two occurrences of the same symbol receive the same index *and* that no individual constant receives the same index as an individual variable. The first requirement is obvious. That two occurrences of the same symbol receive the same index amounts to declaring them to have the same meaning. If ambiguities may be around, there is no logical justification for doing so. The second requirement may be easily explained. Consider the premise set $\{\forall x\, Px, \neg Pa\}$ and note that Pa is derivable from the first premise. If, for example, the first premise is translated as $\forall x\, P^1x^2$, then P^1a^2 is a **CLI**-consequence of it. So there either is an ambiguity in P or there is an ambiguity in a. But suppose that the premise set were translated as $\{\forall x\, P^1x^1, \neg P^2a^1\}$—this translation fulfils the first requirement but not the second. As P^1a^1 is a **CLI**-consequence of this, so is the abnormality $P^{1\cdot 2}a^1$. But this is obviously mistaken because it locates the ambiguity definitely in P, neglecting the possible ambiguity in a.

Let us now move to variants. Actually, **CLI** and similar logics contain a very rich potential—see for example [Bat02b] and [Batar] for applications that have nothing to do with ambiguity-adaptive logic. However, also the ambiguity-adaptive logics deserve further attention. A striking point concerns ambiguities in sentential letters. As we have seen before, if there are three occurrences of the same sentential letter, at least two of them 'have the same meaning'. This is so because having the same meaning is expressed by equivalence, there are only two truth values, and equivalence is truth-functional. However, it is obvious that the same sentential letter (or the same sentence in a natural language) may be used with more than two different meanings. This suggests that one tries to dig deeper into meaning. The meaning of a linguistic element may be seen as composed from different elements. Some bunches of such elements may actually be realistic, in that they occur in the language, whereas others are not. Moreover, it is well-known that speakers often want to express something close to, but slightly different from, a given realistic bunch and still use the same word or phrase. An approach that allows for digging deeper into meaning is available along these lines. Some work has been done on it. I cannot report on it here, but address the reader to some relevant papers: [D'H02], [D'H01], [Urbnt] cite: Stephan / mail zoeken

Before leaving the matter, an important proviso should be mentioned. Much so-called ambiguity arises from the fact that many predicates are *vague*. Vagueness obviously cannot be adequately handled by means of **CLA**—*pace* [VKV03]. See [VvdWvG08] for a decent proposal to upgrade fuzzy *logics* adaptively. anders formuleren

8.4 Adaptive Zero Logic

In the previous sections, we met two extremely weak logics. The first was **CLo**, in which no standard logical symbol has any specific meaning. We have seen that $A \in \mathcal{W}_s$ is **CLo**-derivable from a premise set $\Gamma \subseteq \mathcal{W}_s$ iff A is a member of Γ. The second, even weaker logic, was **CLA**, in which every occurrence of a non-logical symbol may have a meaning that is unrelated to any other such occurrence. Recall that no $A \in \mathcal{W}_s$ is **CLA**-derivable from any premise set $\Gamma \subseteq \mathcal{W}_s$. It is not difficult to combine the weaknesses of both logics. I shall do so and call the result **CL∅**, in words *zero logic*. In zero logic, no symbol has a fixed meaning. While zero logic in itself is utterly useless, it may function as the lower limit of a very useful adaptive logic. The idea of zero logic and the related adaptive logic was first presented in [Bat99d]. The paper is a bit clumsy at some points and uses terminology that has now been replaced.

Defining **CL∅** is easy. For the semantics, replace all standard logical symbols in the **CLI**-semantics by their classical counterparts and do not add anything for the standard logical symbols. Let this logic be called **CL∅I**. For its axiomatization, replace the standard logical symbols in the axiom system of **CLI** by their classical counterparts (and do not add anything for the standard logical symbols). From **CL∅I**, define **CL∅** by

$$\Gamma \vdash_{\mathbf{CL}\emptyset} A \text{ iff } \Gamma^\dagger \vdash_{\mathbf{CL}\emptyset\mathbf{I}} A^\ddagger\,.$$

in which † and ‡ are as in Section 8.3. The logic **CL∅** is really useless. Even the difference between logical and non-logical symbols is blurred. To be more precise, the difference is obviously neat in the metalanguage, but nothing *within* the logic reveals it. This is really the logic that suits the post-modernist. It also shows that post-modernism, in its extreme form, is not viable. If, in a text, any occurrence of any symbol can have whatever meaning, then nothing sensible can be said about the text. Presumably **CL∅** is the logic present in our brains before we start to learn our mother tongue. Only as this learning proceeds, we start connecting words to entities in the world (things, actions, processes) or to representations of such entities, and we start connecting logical terms to operators. In doing so, we are forced to turn the connection into a probabilistic and contextual one. I now move back to logic, but I shall have to return to this point later.

NOG NAZIEN

The most straightforward adaptive logics that have **CL∅I** as their lower limit logic combine it with Reliability or Minimal abnormality and with a specific set of abnormalities. This set is the union of two subsets: (i) the set containing all formulas expressing gluts and gaps (as mentioned in the table at the end of Section 8.1), and (ii) the abnormalities of $\mathbf{CLI}^m$, duly phrased in terms of classical logical symbols. This gives us $\mathbf{CL\emptyset I}^m$ and $\mathbf{CL\emptyset I}^r$. From these we define

$$\Gamma \vdash_{\mathbf{CL\emptyset^m}} A \text{ iff } \Gamma^\dagger \vdash_{\mathbf{CL\emptyset I}^m} A^\ddagger \,.$$

and similarly for $\mathbf{CL\emptyset}^{\mathsf{r}}$.

I shall not present any example proofs in $\mathbf{CL\emptyset I}^m$. These are easy enough in view of what was said in Sections 8.1 and 8.3. It is more important to comment on the use of adaptive zero logic.

Every symbol, logical or non-logical, has a contingent meaning in $\mathbf{CL\emptyset I}^m$. This means that the meaning of a specific occurrence of a symbol will depend on the premises. Of course, there are presuppositions, laid down by the abnormalities. Thus logical symbols are supposed to have their classical meaning, unless and until proven otherwise. Different occurrences of non-logical terms are supposed to have the *same* meaning, unless and until proven otherwise—the fact that our logic is defined within a language *schema* causes these meanings to be left unspecified.

If applied to abnormal premise sets, $\mathbf{CL\emptyset I}^m$ is a marvellous instrument of analysis. It locates each and every possible explanation of the abnormality. The idea here is as explained in Section 8.2, except that the present analysis is richer: ambiguities in the non-logical terms are also considered. The analysis will give rise to different abnormal but non-trivial theories, obtained by blaming one kind of abnormality rather than another, or by blaming the abnormalities in a certain order (combined adaptive logics).

If applied to a normal premise set, $\mathbf{CL\emptyset I}^m$ delivers the full **CL**-consequence set. This is fully the merit of the adaptivity of the logic, because the lower limit logic does not assign any meaning to any symbol. The lower limit logic prescribes literally nothing about any symbol. In a sense, settling the meaning of symbols has become an empirical matter.

The last statements from the previous paragraph should be qualified. It obviously makes a difference which precise set of abnormalities is selected, because this defines the normal interpretation of the symbols. A first choice that underlies $\mathbf{CL\emptyset I}^m$ is that the upper limit logic is **CL**. Some will want to replace this by a different 'standard of deduction'. Next, the selected abnormalities are the plain ones, bare gluts and bare gaps for the logical symbols and plain ambiguity for the non-logical symbols. For the logical symbols, this may be modified to, for example, the abnormalities described in Section 7.4, which require combined logics.

By all means, the present results suggest a formal approach to the interpretation of texts. The logic $\mathbf{CL\emptyset I}^m$ forms a skeleton that may be given some flesh. What should be added is basically a set of suitable suppositions about the actual meaning of certain symbols, logical and non-logical symbols alike, and contextual features should be taken into account. This is not the place to expand upon the topic, but it seemed worth pointing out this possible line of research. The reader will also note the connection with argumentation. Most contributions to that domain are on the non-formal side and close to natural language. $\mathbf{CL\emptyset I}^m$ provides an approach on the formal side and close to formal languages. It seems to me that both approaches may work towards each other—see (the old) [Bat96] for some first ideas on this.

8.5 Strength of Paraconsistency and Ambiguity

I have argued that each of the logics considered in this chapter lead, with respect to some premise sets, to a different maximal consistent interpretation. Obviously, most of the logics trivialize some premise sets that have no **CL**-models. Consider all logics from Sections 8.1 and 8.2. Whether the logic is adaptive or not, the consequence set of $\{p, \neg p\}$ is trivial unless negation is paraconsistent. In this sense paraconsistency has a special status: it provides models for *all* premise sets that have no **CL**-models.

Incidentally, many of the logics from Sections 8.1 and 8.2 are extensions of **CL**. In many of those logics, some standard symbols have the same meaning as the corresponding **CL**-symbols and other **CL**-symbols may be defined. This is fairly obvious for most of the logics. Slightly unexpected might be that $\sim A =_{df} A \supset \neg A$ defines classical negation within **CLaN**, **CLaNs**, and so on.

Ambiguity logics share the strength of paraconsistent logics. Every $\Gamma \subseteq \mathcal{W}_s$, even if it has no **CL**-models, has **CLA**-models.[15] Some paraconsistent logics may even be defined in terms of ambiguity logics—I have shown in [Bat02b] that this holds for **LP** and it is not impossible that a similar result holds for all paraconsistent logics. Note that this is a technical point. A philosophical point is that, even if all paraconsistent logics can be characterized in terms of ambiguity logics, the interpretation of both types of logics is nevertheless different. The question as to the precise meaning of negation should not be confused with the question whether ambiguities occur in non-logical symbols. In this respect, the philosophical tenet of David Lewis in [Lew82] is mistaken. That a given text (or premise set) may be interpreted both ways is altogether a different matter.

What should be concluded from the strength of paraconsistency and ambiguity? Not much as I see it. These approaches offer a road to a maximally non-trivial interpretation of every premise set. However, if another logic provides also such a road for a given premise set, the latter road may be just as sensible. Logics should offer ways for handling the **CL**-triviality of a theory T.

[15]Obviously not every $\Gamma \subseteq \mathcal{W}_s^I$ has **CLI**-models.

Which maximally non-trivial interpretation of T will turn out most interesting will always depend on a non-logical considerations. As early as 1964, Nicholas Rescher remarked in [Res64, p. 37]: "And while the *recognition* of ambiguity does fall within the province of logic, its *resolution* is inevitably an extralogical matter." This holds for every cause of triviality.

8.6 Flip-flop Criteria

That some adaptive logics are flip-flops is annoying. Whenever one develops a new adaptive logic, one has to show that it is a flip-flop, in case we want one, or that it is not a flip-flop, in case it is not intended to be one. Especially the latter task often requires a thorough study of the logic and next finding a metatheoretic proof. So it is fortunate that a simple semantic criterion was discovered. The criterion applies to simple adaptive logics. If the matter is known for these, it is easy enough to figure out whether their combinations are flip-flops. In its present form the criterion applies only to corrective adaptive logics, but that is the difficult bit anyway.

The approach requires some preparation, viz. some alternative formulations of the **CL**-semantics. The semantics from Section 1.7 will be called the *clausal semantics*. Let us turn it into a *tabular semantics* by leaving the assignment function unchanged, replacing the ten clauses specifying the valuation function by the following ten tables—the last two are amalgamated.

Where $A \in \mathcal{S}$:

$v(A)$	A
1	1
0	0

Where $\alpha_1, \ldots, \alpha_n \in \mathcal{C} \cup \mathcal{O}$ and $\pi \in \mathcal{P}^n$:

$\langle v(\alpha_1), \ldots, v(\alpha_n)\rangle, v(\pi)$	$\pi\alpha_1 \ldots \alpha_n$
$\in$	1
$\notin$	0

Where $\alpha, \beta \in \mathcal{C} \cup \mathcal{O}$:

$v(\alpha), v(\beta)$	$\alpha = \beta$
$=$	1
$\neq$	0

	$\neg$
1	0
0	1

$\wedge$	1	0
1	1	0
0	0	0

$\vee$	1	0
1	1	1
0	1	0

$\supset$	1	0
1	1	0
0	1	1

$\equiv$	1	0
1	1	0
0	0	1

$\{v_M(A(\alpha)) \mid \alpha \in \mathcal{C} \cup \mathcal{O}\}$	$\forall\alpha(A(\alpha))$	$\exists\alpha(A(\alpha))$
$\{1\}$	1	1
$\{0, 1\}$	0	1
$\{0\}$	0	0

We have seen that some logics display gluts or gaps or both. For them, I shall articulate a semantics in which the valuation has the form $v_M : \mathcal{W}_\mathcal{O} \to \{t, u, a, f\}$, in which the values intuitively stand for true, glut, gap, and false. Let us call this the *tuaf semantics*. A tuaf semantics may be two-valued, three valued, or four valued, depending on the number of values needed. Another important piece of information is that t and u are the designated values. So a model M verifies A iff $v_M(A) \in \{t, u\}$. This settles at once the semantic consequence relation $\Gamma \models A$.

In some logics no gluts occur, or no gaps. So there only two or three of the values will be used. For example, the tuaf semantics for **CL** is boringly isomorphic to **CL**'s tabular semantics: every 1 is replaced by t and every 0 by f—why this is so will soon become clear. I spell out this semantics for future reference.[16]

Where $A \in \mathcal{S}$:

$v(A)$	A
1	t
0	f

Where $\alpha_1, \ldots, \alpha_n \in \mathcal{C} \cup \mathcal{O}$ and $\pi \in \mathcal{P}^n$:

$\langle v(\alpha_1), \ldots, v(\alpha_n)\rangle, v(\pi)$	$\pi\alpha_1 \ldots \alpha_n$
$\in$	t
$\notin$	f

Where $\alpha, \beta \in \mathcal{C} \cup \mathcal{O}$:

$v(\alpha), v(\beta)$	$\alpha = \beta$
$=$	t
$\neq$	f

	$\neg$
t	f
f	t

$\wedge$	t	f
t	t	f
f	f	f

$\vee$	t	f
t	t	t
f	t	f

$\supset$	t	f
t	t	f
f	t	t

$\equiv$	t	f
t	t	f
f	f	t

$\{v_M(A(\alpha)) \mid \alpha \in \mathcal{C} \cup \mathcal{O}\}$	$\forall\alpha(A(\alpha))$	$\exists\alpha(A(\alpha))$
$\{t\}$	t	t
$\{f, t\}$	f	t
$\{f\}$	f	f

The matter gets interesting when we move to logics that allow for gluts or gaps. Let us start with **CLuN**. Its tabular semantics looks exactly as for **CL**, except that the table for negation is replaced. I spell out the matter very explicitly to avoid confusion.

$v_M(A)$	$v(\neg A)$	$v_M(\neg A)$
1	0	0
1	1	1
0	(any)	1

The "(any)" indicates that it has no effect on the value of $v_M(\neg A)$ whether the value of $v(\neg A)$ is 0 or 1.

In order to obtain the tuaf semantics for **CLuN**, we proceed in a special way. Let me first present the tuaf semantics and next explain.

Where $A \in \mathcal{S}$:

$v(A)$	A
1	t
0	f

[16] I use the same notation, $v_M(A)$, for the valuation function in all three kinds of semantics and I shall do so for all logics. The matter is always disambiguated by the context.

Where $\alpha_1, \ldots, \alpha_n \in \mathcal{C} \cup \mathcal{O}$ and $\pi \in \mathcal{P}^n$:

$\langle v(\alpha_1), \ldots, v(\alpha_n)\rangle, v(\pi)$	$\pi\alpha_1 \ldots \alpha_n$
$\in$	t
$\notin$	f

Where $\alpha, \beta \in \mathcal{C} \cup \mathcal{O}$:

$v(\alpha), v(\beta)$	$\alpha = \beta$
$=$	t
$\neq$	f

$v_M(A)$	$v(\neg A)$	$v_M(\neg A)$
t	0	f
t	1	u
u	0	f
u	1	u
f	(any)	t

$\wedge$	t	u	f
t	t	t	f
u	t	t	f
f	f	f	f

$\vee$	t	u	f
t	t	t	t
u	t	t	t
f	t	t	f

$\supset$	t	u	f
t	t	t	f
u	t	t	f
f	t	t	t

$\equiv$	t	u	f
t	t	t	f
u	t	t	f
f	f	f	t

$\{v_M(A(\alpha)) \mid \alpha \in \mathcal{C} \cup \mathcal{O}\}$	$\forall\alpha(A(\alpha))$	$\exists\alpha(A(\alpha))$
$\subseteq \{t, u\}$	t	t
$= \{f\}$	f	f
(other)	f	t

The assignment function, which is the same for all semantics considered, has values in $\{0, 1\}$. As the value u is introduced by the table for negation, this value has to occur in all tables in which the input-entries are valuation values. The "(any)" has the same meaning as in the tabular semantics. The "(other)" obviously means that the set $\{v_M(A(\alpha)) \mid \alpha \in \mathcal{C} \cup \mathcal{O}\}$ contains at least one f and at least one t or u.

Where precisely does the tuaf semantics assign the value u? In the *tabular* semantics for **CLuN**, every table defines, for some form A, $v_M(A)$ as a function of the valuation value of subformulas of A and possibly of the assignment value of A. So it is easy to check whether *at that point* the valuation function of **CLuN**, or of any other logic **L** allowing for gluts or gaps, agrees with **CL**. If both valuation functions assign a 1 at that point, the tuaf semantics assigns a t; if both valuation functions assign a 0 at that point, the tuaf semantics assigns a f; if the **CL**-valuation assigns a 1 and the **L**-valuation assigns a 0, the tuaf semantics assigns an a because this is a gap; if the **CL**-valuation assigns a 0 and the **L**-valuation assigns a 1, the tuaf semantics assigns an u because this is a glut. Let us call this the *tuaf criterion*—this is a criterion for constructing three-valued and four-valued logics, not the promised criterion for locating flip-flops. The reader should check that the tuaf semantics for **CLuN** assigns a u at two specific points in accordance with the convention just sketched. At those points, the tabular assignment function assigns a 1 to $v_M(A)$. So the tabular assignment function for **CL** assigns a 0 to $v_M(\neg A)$, but the tabular assignment function for **CLuN** assigns a 1 to $v_M(\neg A)$.

The words "glut" and "gap" have been used with several meanings in the literature. In [vW96], for example, Georg Henrik von Wright says that there is an overlap (rather than glut) when a formula is true together with its negation and that there is a gap if a formula is false together with its negation. In the same place, von Wright calls a formula false iff its negation is true.[17] So he also says that there is a glut (or overlap) if a formula is both true and false, and a gap if it is neither. The tuaf criterion from the previous paragraph is in line with the way in which I used the words glut and gap in Section 8.1. There is, however, an important proviso, which was implicit there but became explicit here: whenever a formula has the value u, this formula constitutes a glut, but not the other way around. The point is that the values u and a are only assigned at points where gluts or gaps *originate*. This will become absolutely obvious when we look at **CLuNs**, which we shall do right now. The clausal semantics is presented in Section 7.2. I shall need the equivalence classes defined there in the sequel. The tabular semantics is just like that for **CL**, except that the table for negation is replaced by the following tables.

Where $A \in \mathcal{W}^p_{\mathcal{O}}$:

$v_M(A)$	$\{v(\neg B) \mid B \in [\![A]\!]\}$	$v_M(\neg A)$
1	$= \{0\}$	0
1	$\neq \{0\}$	1
0	(any)	1

A	$\neg\neg A$
$A \wedge \neg B$	$\neg(A \supset B)$
$\neg A \vee \neg B$	$\neg(A \wedge B)$
$\neg A \wedge \neg B$	$\neg(A \vee B)$
$(A \vee B) \wedge (\neg A \vee \neg B)$	$\neg(A \equiv B)$
$\exists\alpha\neg A(\alpha)$	$\neg\forall\alpha A(\alpha)$
$\forall\alpha\neg A(\alpha)$	$\neg\exists\alpha\neg A(\alpha)$
1	1
0	0

The lower table is obviously a summary of seven tables, each stating that the formula in the right column has the same value as the formula in the left column. Of course, the fascinating bit is the tuaf semantics.

Where $A \in \mathcal{S}$:

$v(A)$	A
1	t
0	f

Where $\alpha_1, \ldots, \alpha_n \in \mathcal{C} \cup \mathcal{O}$ and $\pi \in \mathcal{P}^n$:

$\langle v(\alpha_1), \ldots, v(\alpha_n)\rangle, v(\pi)$	$\pi\alpha_1 \ldots \alpha_n$
$\in$	t
$\notin$	f

Where $\alpha, \beta \in \mathcal{C} \cup \mathcal{O}$:

$v(\alpha), v(\beta)$	$\alpha = \beta$
$=$	t
$\neq$	f

[17]Remember that I follow a completely different convention in keeping the metalanguage classical everywhere.

Where $A \in \mathcal{W}^p_{\mathcal{O}}$:

$v_M(A)$	$\{v(\neg B) \mid B \in [\![A]\!]\}$	$v_M(\neg A)$
t	$= \{0\}$	f
t	$\neq \{0\}$	u
f	(any)	t

A	$\neg\neg A$
$A \wedge \neg B$	$\neg(A \supset B)$
$\neg A \vee \neg B$	$\neg(A \wedge B)$
$\neg A \wedge \neg B$	$\neg(A \vee B)$
$(A \vee B) \wedge (\neg A \vee \neg B)$	$\neg(A \equiv B)$
$\exists\alpha\neg A(\alpha)$	$\neg\forall\alpha A(\alpha)$
$\forall\alpha\neg A(\alpha)$	$\neg\exists\alpha\neg A(\alpha)$
t	t
u	t
f	f

$\wedge$	t	u	f
t	t	t	f
u	t	t	f
f	f	f	f

$\vee$	t	u	f
t	t	t	t
u	t	t	t
f	t	t	f

$\supset$	t	u	f
t	t	t	f
u	t	t	f
f	t	t	t

$\equiv$	t	u	f
t	t	t	f
u	t	t	f
f	f	f	t

$\{v_M(A(\alpha)) \mid \alpha \in \mathcal{C} \cup \mathcal{O}\}$	$\forall\alpha(A(\alpha))$	$\exists\alpha(A(\alpha))$
$\subseteq \{t, u\}$	t	t
$= \{f\}$	f	f
(other)	f	t

There is only one 'output value' u in all these tables. The reader is prayed to check that this agrees with the tuaf criterion. Here are some hints. Primitive formulas never receive the value u. Formulas of which the central symbol is not a negation cannot receive a u because, in the tabular semantics, they receive exactly the formulas that the **CL**-semantics would assign in view of the values of the subformulas. Finally, consider the table for the negation of complex formulas. Among the input-entries of this table is u. Please note that the only formulas in the left column that may have a valuation value u are those of the form A. These may start with a negation and hence also have the form $\neg C$. But even if the formula of the form A has the value u, the formula of the corresponding form $\neg\neg A$ needs the value t. Indeed, if, in the **CL**-semantics, A has the valuation value 1, then so has $\neg\neg A$, and the **CLuNs**-semantics agrees completely with this.

The point is a bit tricky, so allow me to expand. Some readers may feel there is something wrong here. If one knows, they will argue, that $v_M(A) = u$, then one knows that the set of formulas verified by M has no **CL**-model. So in which sense does the **CLuNs**-model agree with **CL**-semantics? This is quite all right and it is the reason why the tuaf criterion refers to the tabular semantics, which is two-valued. All one knows in the tabular semantics is that this A, which also has the form $\neg c$, receives the value 1. That $\neg\neg A$ also receives the value 1 agrees with the **CL**-semantics.

I mentioned before that the tuaf semantics introduces values u and a where the gluts or gaps originate. Please check this. If $v_M(Pa) = v_M(\neg Pa) = 1$ in the clausal or tabular **CLuNs**-semantics, the tuaf semantics settles for $v_M(\neg Pa) = u$. Obviously, if $v_M(Pa) = v_M(\neg Pa) = v_M(Qb) = 1$ in the clausal or tabular **CLuNs**-semantics, then $v_M(Pa \wedge Qb) = v_M(\neg Pa \vee \neg Qb) = 1$ in the same semantic system, and hence also $v_M(Pa \wedge Qb) = v_M(\neg(Pa \wedge Qb)) = 1$. However, the tuaf semantics settles for $v_M(\neg(Pa \wedge Qb)) = t$. And this is precisely as we want it: the glut does not originate with $\neg(Pa \wedge Qb)$; it originates with $\neg Pa$.

A comparison with the tuaf semantics for **CLuN** is enlightening. In **CLuN** *every* inconsistency receives the value u because the truth of no negative formula, however complex, results from its composing parts. If both A and $\neg A$ have the valuation value 1, the glut starts right with $\neg A$. This is why I said, some pages ago, that **CLuN** maximally isolates inconsistencies. As far as **CLuN** is concerned, if $v_M(p \wedge \neg p) = 1$ then it is possible that $v_M(\neg(p \wedge \neg p)) = 1$ as well, but nothing requires this and, if it holds, it forms a novel inconsistency.

A very instructive illustration is the tuaf semantics for the logic with the terrifying name **CLuCoDaM**, in words, the logic that allows for conjunction gluts, for disjunction gluts as well as for disjunction gaps, and for implication gaps. As Replacement of Identicals is invalid, we will not have the complication present in **CLuNs**. Moreover, I skip the tabular semantics. The reader may very easily construct it in case the tuaf semantics would not be obvious at once. The tables for $A \in \mathcal{S}$, $\pi\alpha_1 \ldots \alpha_n$ and $\alpha = \beta$ are exactly as for **CLuN** and **CLuNs** and are not repeated.

	$\neg$
t	f
u	f
a	t
f	t

$\equiv$	t	u	a	f
t	t	t	f	f
u	t	t	f	f
a	f	f	t	t
f	f	f	t	t

$v(A \wedge B) = 1:$

$\wedge$	t	u	a	f
t	t	t	u	u
u	t	t	u	u
a	u	u	u	u
f	u	u	u	u

$v(A \vee B) = 1:$

$\vee$	t	u	a	f
t	t	t	t	t
u	t	t	t	t
a	t	t	u	u
f	t	t	u	u

$v(A \supset B) = 1:$

$\supset$	t	u	a	f
t	t	t	f	f
u	t	t	f	f
a	t	t	t	t
f	t	t	t	t

$v(A \wedge B) = 0:$

$\wedge$	t	u	a	f
t	t	t	f	f
u	t	t	f	f
a	f	f	f	f
f	f	f	f	f

$v(A \vee B) = 0:$

$\vee$	t	u	a	f
t	a	a	a	a
u	a	a	a	a
a	a	a	f	f
f	a	a	f	f

$v(A \supset B) = 0:$

$\supset$	t	u	a	f
t	a	a	f	f
u	a	a	f	f
a	a	a	a	a
f	a	a	a	a

$\{v_M(A(\alpha)) \mid \alpha \in \mathcal{C} \cup \mathcal{O}\}$	$\forall\alpha(A(\alpha))$	$\exists\alpha(A(\alpha))$
$\subseteq \{t, u\}$	t	t
$\subseteq \{a, f\}$	f	f
(other)	f	t

This semantics illustrates a variety of cases. As there are no gluts or gaps with respect to negation, equivalence, and the quantifiers, the output values are all

t and f. For conjunction there are only gluts. So if $v(A \wedge B) = 0$, one obtains the normal table; if $v(A \wedge B) = 1$ every f in the normal table is replaced by a u. Implication and disjunction illustrate the other cases.

Before, finally, coming to the flip-flop criterion, the reader who is inexperienced with many-valued logics should be warned that a tuaf semantics forms a rather unusual three-valued or four-valued system. Actually, I do not know of any published semantics that behaves according to the tuaf criterion. Often values like u and a, values differing from simple truth or falsehood, are assigned to primitive formulas in order to indicate that formulas composed from them behave in an unusual way. Thus in many three-valued semantics for paraconsistent logics, A has the value u, or another value different from truth and falsehood, iff both A and $\neg A$ are verified by the model. Often the unusual values are also spread. In the same paraconsistent semantic systems, it often holds that $v_M(A \wedge B) = u$ iff $v_M(A) = v_M(B) = u$. So people not familiar with many-valued logics should not use their insights from the tuaf semantics when studying many-valued logics from the literature.

Now to the flip-flop criterion. The tuaf criterion warrants that a value u or a is only assigned to a formula A (i) if A causes a glut, respectively gap, to obtain and (ii) if this glut or gap does not result from the fact that subformulas of A are verified or falsified by the models. In other words, all formulas that in a model M obtain the value u, respectively a, obtain the value f, respectively t, in another model M that agrees with M on all proper subformulas of A. Intuitively this means that the glut or gap is avoidable.

This is worth a further comment because it is related to the recursiveness of a semantics. The assignment function assigns values to all non-logical symbols of the language (and possibly to some other things as well). The valuation function assigns values to formulas in a certain order, relying on the assignment. The order in which valuation values are assigned depends on a certain complexity function. All formulas are given a certain degree of complexity. A semantics is recursive iff the valuation value of any formula A is itself a function of the assignment values of formulas that are at most as complex as A and of the valuation values of formulas that are *less* complex than A. Note that there are many complexity functions[18]—I refer to [VBC0x] for some theory on complexity functions and their effect. So not every semantics needs to assign valuation values in the same order.

Incidentally, there is nothing wrong with a non-recursive semantics, provided it can be turned into a recursive one. All deterministic semantics presented in this book are recursive. The indeterministic semantic systems in this book are officially not recursive, but I have shown an easy way to turn them into semantic systems that are deterministic and recursive, viz. by letting the assignment function assign values to all formulas and by referring to the assignment function where gluts or gaps originate. Indeterministic semantics occurred already in my [Bat80] and probably in earlier papers by others. For an interesting study and application of indeterministic semantics, I refer to work by Arnon Avron and associates, [Avr05, AK05, ABNK07].

[18]A very simple one defines the complexity of A as the number of logical symbols different from identity that occur in A. This function is not suitable for the **CLuNs**-semantics because, for example, the value of $v_M(\neg(A \wedge B))$ depends on the value of $v_M(\neg A \vee \neg B))$. It is instructive to formulate a suitable function. Here is a hint: $c(A) = 1$ if $A \in \mathcal{S}$, $c(A \wedge B) = c(A \vee B) = c(A) + c(B) + 1$, $c(\neg A) = c(A) \times 2$, and so on—beware of equivalence.

Returning to the tuaf criterion, it warrants that formulas that receive the value u or a do not receive this value because of the valuation value of less complex formulas, but because of a direct interference of the assignment. If these gluts and gaps are located, then all problems (with respect to the upper limit logic **CL**) are located. If a classical disjunction of gluts and gaps is derivable from the premises,[19] then so is a classical disjunction of gluts caused by formulas receiving the value u and of gaps caused by formulas receiving the value s. The formulas that may possibly receive the value u or a are formulas of a certain possibly restricted form. In the **CLuN**-semantics and in the **CLuCoDaM**-semantics, the logical forms are unrestricted. In the **CLuNs**-semantics, the logical form, viz. $\exists(A \wedge \neg A)$ is restricted to $A \in \mathcal{W}^p_{\mathcal{O}}$.

The abnormalities of any corrective adaptive logic are classical conjunctions of formulas. Let us opt for the formulation used in this chapter: the classical conjunction of (i) a classical expression, respectively its classical negation, and (ii) the classical negation of the corresponding standard expression, respectively the corresponding standard expression. We are now in a position to formulate the *flip-flop criterion* in terms of the tuaf semantics. To avoid clutter in the formulation, let $A(\overline{\alpha})$ be a formula in which $n \geq 1$ members of $\mathcal{V}$ occur free and let $A(\overline{\beta})$ be the result of systematically replacing in $A(\overline{\alpha})$ every member of $\mathcal{V}$ by a member of $\mathcal{C} \cup \mathcal{O}$ which do not occur in $A(\overline{\alpha})$.[20]

Flip-flop criterion: Where **AL** is defined by **LLL**, Ω, and Reliability or Minimal Abnormality, **AL** is not a flip-flop logic if, for every $A \in \Omega$ and for every tuaf **LLL**-model M, one of the following holds: (i) $M \Vdash A$ iff $v_M(A) = u$, (ii) A has the form $\check{\neg} B$ and $M \Vdash A$ iff $v_M(B) = a$, (iii) A has the form $B \check{\wedge} C$ and $M \Vdash A$ iff $v_M(C) = u$, (iv) A has the form $B \check{\wedge} \check{\neg} C$ and $M \Vdash A$ iff $v_M(C) = a$, (v) A has the form $\check{\exists}(B \check{\wedge} C(\overline{\alpha}))$ and $M \Vdash A$ iff $v_M(C(\overline{\beta})) = u$ for some $\overline{\beta}$, (vi) A has the form $\check{\exists}(B \check{\wedge} \check{\neg} C(\overline{\alpha}))$ and $M \Vdash A$ iff $v_M(C(\overline{\beta})) = a$ for some $\overline{\beta}$.

Note that the flip-flop criterion is an implication, not an equivalence. It states that some adaptive logics are not flip-flops, not that some are.

Applying the flip-flop criterion is easy: one articulates the tuaf semantics for the lower limit logic **LLL**, identifies the possibly restricted logical forms that may receive the value u or a, and checks whether all members of the set of abnormalities have the required form. Moreover, it is obvious that the flip-flop criterion is correct. If a classical disjunction of these abnormalities, $Dab(\Delta)$, is **LLL**-derivable from a premise set Γ, then $Dab(\Delta)$ is not **LLL**-derivable from a less complex[21] *Dab*-consequence of the premise set. In other words, if an abnormality A is falsified by every minimal abnormal (or by every reliable) **LLL**-model of Γ, then A is not a disjunct of any minimal *Dab*-consequence of Γ. So the adaptive logic is not a flip-flop.

That the flip-flop criterion is only an implication is not much of a hindrance. If it is compatible with the criterion that an adaptive logic is a flip-flop, then

[19] The "derivable" obviously means derivable by the logic characterized by the tuaf semantics.

[20] To keep the criterion as simple as possible, I suppose that, if the abnormalities are (possibly existentially quantified) conjunctions, then these conjunctions are classical (or have the classical meaning) and the conjuncts occur in a certain order. The supposition agrees with all abnormalities mentioned in this book.

[21] Obviously "complex" here refers to the complexity function which underlies the recursive character of the semantics.

it is not difficult to figure out by means of the tuaf **LLL**-semantics whether the **LLL**-derivability of a *Dab*-formula from Γ causes every abnormality to be a disjunct of a minimal *Dab*-consequence of Γ. This basically always proceeds in the way illustrated by the proof of Theorem 7.3.1.

Let us consider some applications. The tuaf semantics for **CLuNs** shows that the adaptive logics $\mathbf{CLuNs}^r$ and $\mathbf{CLuNs}^m$ are not flip-flops. I leave it to the reader to verify that the flip-flop criterion applies. It follows immediately that $\mathbf{LP}^r$ and $\mathbf{LP}^m$ are not flip-flops either.

A more interesting case is presented by the $\mathbf{C}_n$ logics because a complication is involved. The congruence requirement may be handled by first defining a pre-valuation, which looks just like a tuaf semantics itself, and next defining a valuation from the pre-valuation. Here is the tuaf semantics for $\mathbf{C}_1$.

The assignment function is again the general one, as for example in the **CL**-semantics from Section 1.7. The pre-valuation $v_M : \mathcal{W}_\mathcal{O} \rightarrow \{t, u, f\}$ is characterized by the following tables—the tables for $A \in \mathcal{S}$, $\pi\alpha_1 \ldots \alpha_n$ and $\alpha = \beta$ are exactly as for the tuaf valuation of **CLuN** and of **CLuNs** and are not repeated.

Where $A \in \mathcal{W}^p_\mathcal{O}$:

$v_M(A)$	$v(\neg A)$	$v_M(\neg A)$
t	0	f
t	1	u
f	(any)	t

Where $\dagger \in \{\vee, \wedge, \supset\}$ and $A \dagger B$ has not the form $C \wedge \neg C$:[22]

$v_M(A \dagger B)$	$v_M(A^{(1)})$	$v_M(B^{(1)})$	$v(\neg(A \dagger B))$	$v_M(\neg(A \dagger B))$
t	t	t	(any)	f
t	(other)		0	f
t	(other)		1	u
f	(any)		(any)	t

Where $\mathsf{Q} \in \{\forall, \exists\}$:

$v_M(\mathsf{Q}\alpha A(\alpha))$	$\{v_M(A(\beta)^{(1)}) \mid \beta \in \mathcal{C} \cup \mathcal{O}\}$	$v(\neg \mathsf{Q}\alpha A(\alpha))$	$v_M(\neg \mathsf{Q}\alpha A(\alpha))$
t	$= \{t\}$	(any)	f
t	$\neq \{t\}$	0	f
t	$\neq \{t\}$	1	u
f	(any)	(any)	t

The other tables apply to all members of $\mathcal{W}_\mathcal{O}$:

$v_M(\neg A)$	$v(\neg\neg A)$	$v_M(\neg\neg A)$
t	(any)	f
u	0	f
u	1	u
f	(any)	t

$\neg A$	$A^{(1)}$
t	t
u	f
f	t

$\wedge$	t	u	f
t	t	t	f
u	t	t	f
f	f	f	f

$\vee$	t	u	f
t	t	t	t
u	t	t	t
f	t	t	f

$\supset$	t	u	f
t	t	t	f
u	t	t	f
f	t	t	t

$\equiv$	t	u	f
t	t	t	f
u	t	t	f
f	f	f	t

[22] Remember that $A^{(1)}$ abbreviates $\neg(C \wedge \neg C)$.

$\{v_M(A(\alpha)) \mid \alpha \in \mathcal{C} \cup \mathcal{O}\}$	$\forall\alpha A(\alpha)$	$\exists\alpha A(\alpha)$
$\in \wp\{t,u\}$	t	t
$= \{f\}$	f	f
(other)	f	t

Let $f(A)$ be the obtained by first deleting all vacuous quantifiers in A and then systematically replacing all variables in the result by the first variables of the alphabet in alphabetical order. Next, we define the valuation values V_M in terms of the pre-valuation values v_M by $V_M(A) = v_M(f(A))$.

Alternatively, a pre-valuation v_M is called a valuation iff $v_M(A) = v_M(B)$ whenever $A \equiv^c B$.

Transforming the above semantics to logic $\mathbf{C}_n$ (for any $n < \omega$) is an easy exercise left to the reader—the formulation of the tables for $\mathbf{C}_1$ and the plot described in the previous paragraph indicate the road. For $\mathbf{C}_{\overline{\omega}}$, one replaces the tables for negation by the left and middle table below, and adds the table to the right below for classical negation:

$v_M(A)$	$v(\neg A)$	$v_M(\neg A)$
t	0	f
t	1	u
f	(any)	t

$v_M(\neg A)$	$v(\neg\neg A)$	$v_M(\neg\neg A)$
t	(any)	f
u	0	f
u	1	u
f	(any)	t

A	$\check{\neg} A$
t	f
u	f
f	t

The reader is prayed to check that the logics ${\mathbf{C}_n}^r$ and ${\mathbf{C}_n}^m$ are not flip-flops. This is particularly interesting because many abnormalities are $\mathbf{C}_n$-derivable from other abnormalities, which was a reason to suspect them to be flip-flops. However, the flip-flop criterion shows that, in the $\mathbf{C}_n$ logics, no inconsistency is derivable from an inconsistency that is less complex according to the complexity function underlying the semantics.

Incidentally, an indeterministic tuaf semantics is often more transparent than its deterministic counterpart. As it does not refer to the valuation, we get less clutter in the heads of the tables. So let me display the relevant tables, viz. negation tables, for $\mathbf{C}_1$.

Where $A \in \mathcal{W}^p_{\mathcal{O}}$:

A	$\neg A$
t	$[f,u]$
f	t

Where where $\dagger \in \{\vee, \wedge, \supset\}$ and $A \dagger B$ has not the form $C \wedge \neg C$:

$A * B$	$A^{(1)}$	$B^{(1)}$	$\neg(A * B)$
t	t	t	f
t	(other)		$[f,u]$
f	(any)		t

Where $\mathsf{Q} \in \{\forall, \exists\}$:

$\mathsf{Q}\alpha A(\alpha)$	$\{v_M(A(\beta)^{(1)}) \mid \beta \in \mathcal{C} \cup \mathcal{O}\}$	$\neg\mathsf{Q}\alpha A(\alpha)$
t	$\{t\}$	f
t	(other)	$[f,u]$
f	(any)	t

The other tables apply to all members of $\mathcal{W}_\mathcal{O}$:

$\neg A$	$\neg\neg A$
t	f
u	$[f, u]$
f	t

$\neg A$	$A^{(1)}$
t	t
u	f
f	t

The expression $[f, u]$ indicates that the value may be f or u—this is an indeterministic semantics. Note that the 'normal' value, the one that agrees with **CL** at this point, is f. So if the value is u, it 'drops from the sky' as far as the indeterministic semantics is concerned—in the deterministic semantics, the assignment function interferes at this point. The 'dropping from the sky' holds for the semantics only; a premise set may require that some values are u in its models. For other logics, a premise set may require some values to be a. The metaphor is helpful, however, because it highlights that the values u and a occur at points where an abnormality is generated.

One may wonder whether it is possible to express the abnormal part of a model by referring to the set of formulas that have a valuation value u or a. The answer is yes, but there is a proviso because u or a may be the valuation value of a pseudo-formula. Let P be a predicate of rank 1 and o_1 and o_2 pseudo-constants. Consider a **CLuN**-model. Suppose that Po_1 has the valuation value u. It follows that the abnormality $\exists x(Px \wedge \neg Px)$, respectively $\check{\exists} x(\check{\neg}\check{\neg} Px \check{\wedge} \neg Px)$ is verified by the model. Suppose that also Po_2 has the valuation value u. This is another pseudo-formula that obtains the value u, but obviously no new abnormality results.[23]

Before leaving this section, a warning is in place. In its present form, the flip-flop criterion applies only to corrective adaptive logics and on the condition that a tuaf semantics can be formulated for the lower limit logic. So, for example, it cannot be applied to ampliative adaptive logics. However, as **CL** is the lower limit of these logics, it is rather easy to determine whether an ampliative adaptive logic is a flip-flop. So there is no real urge for a criterion.

[23] There is an interesting relation, but again not an utterly simple one, with the set of abnormalities introduced in Section 7.5.