Logic of Information Knowledge

Alexei Yu. Muravitsky
Department of Computer Science
Courant Institute of Mathematical Sciences
251 Mercer Street
New York, New York 10012
Internet: `mrvtskya@acf4.nyu.edu`

Abstract

We share with some philosophers the view that a state of knowledge, being a part of the real world, can bring contradiction into it. Such an ontological reading of knowledge is very important when one deals with information knowledge, which arises as the content of the computer's memory when the computer is placed into changeable information environment ("information flow"), and "must" be able to tolerate any information (not excluding contradictions) from the computer's users. Continuing research begun in [KM 93], we consider in length one kind of Scott-continuous operations introduced there. Each such operation $[A \to B]$, where A and B are formulas in a propositional language, called a rule, moves the computer to a "minimal" state of knowledge, in which B is true, if in a current state A is true. Note that the notion of rule is used here in an information-transforming sense, rather than in the ordinary truth-sound sense. We distinguish between global and local rules and show that these notions are decidable. Also, we define a modal epistemic logic as a tool for the prediction of possible evolution of the system's knowledge and establish decidability of this logic.

1 Introduction

The question of the toleration of inconsistency has a long tradition, often as "admission, or insistence, that some statement is both true and false, in a context where not everything is accepted or some things are rejected" (cf. [PR 89]). However, formal elaborations of this approach have been mostly *epistemological*, considering, on the one hand, the "inherently consistent" real world and, on the other hand, admitting the possibility of its inconsistent description. In other words, "the real world can tolerate inconsistency of the reasoner's beliefs, since the latter may not be grounded in the reality" (cf. [KL 89]). Nevertheless, some philosophers have attempted to advocate the view that the state of knowledge being a part of the real world can bring contradiction into it (cf. [RB 79]). Such an *ontological* reading of knowledge is very important when one deals with *information knowledge*. The latter is the content of the computer's memory, which "must" be able to tolerate any information (not excluding contradictions) from the computer's users. Thus there is no need to distinguish knowledge from belief when working with information knowledge, because the computer believes all it knows, and knows all its beliefs.

It is clear that information knowledge is never complete. Although much recent work in artificial intelligence has developed new formal techniques for working with incomplete knowledge, the means for changing the computer's knowledge associated with the new approach, as in the case of classical logic, have been oriented towards consistent knowledge bases (cf. [Gin 87]). The crucial point here, however, is that the consistency orientation inherent to nonmonotonic logic, in particular to the logic of default reasoning, is not necessary and in practice not even possible when dealing with information knowledge.

Now the question arises: Which kinds of operations should we use to change information in a knowledge base that admits contradiction? This certainly depends on how knowledge has been represented. In 1975 N.Belnap suggested (though implicitly; cf. [Bel 75, Bel 76]) to consider information knowledge as a data type employing Dana Scott's axiomatic definition of the notion of data type (cf. [Sco 71]). According to this approach we can only admit continuous functions in the Scott topology (see [Sco 72, GHKLMS 80]) to change a state of the knowledge base. This point of view was extensively developed in [KM 93]. (See [KM 90] as a preliminary abstract).

We suppose that the knowledge with which we deal is expressed in the formal propositional language, that is, it can be expressed by truth-valuable propositions (comp. [Isr 93]). In the suggested approach, a *declarative* dynamic knowledge base consists of a finite number of finite pairwise incomparable *setups* that in entirety form *a minimal (epistemic) state* — a current actual epistemic state of the computer. The collection of minimal states forms a lattice (AFE) that is *an effective basis* of the lattice AGE of all *generalized* epistemic states — the field of activities of the continuous operations, to which we, following Scott's approach, would like to limit ourselves. Thereby, an intelligent system is thought of as including a collection of continuous operations acting on AGE as a topological space to modify the computer's epistemic states as elements in AFE. Moreover, those operations are supposed to be *coordinated* with the basis AFE in the sense, that any result of such an operation from points in AFE belongs to AFE (*Scott Principle*; cf. [KM 93]).

Among three kinds of the Scott-continuous operations on AGE introduced in [KM 93], one is of particular importance. Each such operation corresponds to an expression $A \to B$ called *a rule* (or a *conditional*), where A and B are formulas in our language, and moves the computer to a minimal state of knowledge, in which B is true, if in a current state A is true. Note that the notion of rule is used here, however, rather in an information-transforming sense, than in the ordinary truth-sound sense.

We distinguish *global* (*proper* and *improper*) and *local* rules. The former helps separate the rules which comprise explicit *imperative* knowledge of the system from those which are a part of its procedural knowledge. Local rules, in turn, concern the question how a particular rule $A \to B$ acts in a particular state of knowledge ε.

Finally, we consider the possibility that the computer knows about its own state of knowledge and how it can evolve. The simple propositional language is apparently not enough to answer that question. Therefore, we attract a language of *modal epistemic logic* and solve the problem of decidability of the class of epistemic formulas valid in every minimal epistemic state.

2 Preliminaries

According to our intention to admit contradictory information into a knowledge base, we (following [Bel 75, Bel 76]) introduce the set $\Im$ of four truth values: $\boldsymbol{f}$ (falsehood), $\boldsymbol{t}$ (truth), $\bot$ (unknown), $\top$ (both truth and falsehood), partially ordered by the relation $\sqsubseteq$ as the "information lattice" $\mathsf{A4}$ $(=<\Im;\sqcap,\sqcup>)$ (see Fig. 1). Formulas of the adopted propositional language are built up of atomic ones that constitute a set Var $(=\{p_1,p_2,\ldots,\})$ of propositional variables with connectives: $\wedge$ (conjunction), $\vee$ (disjunction) and $\neg$ (negation). We denote formulas via A, B, C,... and a variable via π, possibly with indices. The truth value of a formula A in a setup is defined with respect to Belnap's "logical lattice" $\mathsf{L4}$ $(=<\Im;\wedge,\vee>)$ (see Fig. 1) that is partially ordered by the relation $\leq$ according to the following rules:

$$s(B \wedge C) = s(B) \wedge s(C),$$

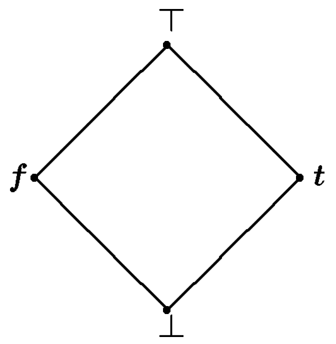
$$s(B \vee C) = s(B) \vee s(C),$$

$$s(\neg B) = \neg s(B),$$

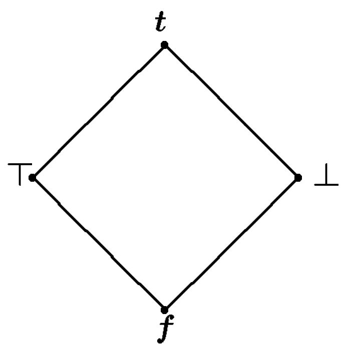
where s is a *setup*, that is a mapping $s : \mathsf{Var} \to \Im$, and the operation $\neg$ is defined by means of the following conditions: $\neg\boldsymbol{t} = \boldsymbol{f}$, $\neg\boldsymbol{f} = \boldsymbol{t}$ and $\neg\tau = \tau$, if $\tau \in \{\bot, \top\}$ (cf. [Bel 75, Bel 76]).

Figure 1: Lattices $\mathsf{A4}$ and $\mathsf{L4}$.

A4



L4



Lattices $\mathsf{A4}$ and $\mathsf{L4}$ took together and with the operation $\neg$ form the simplest nontrivial bilattice in the sense of [Gin 88].

Proposition 1 *The operation $\sqcap$ on the lattice $\mathsf{A4}$ is monotonic with respect to the ordering $\leq$ on the lattice $\mathsf{L4}$, that is,*

$$a_1 \leq b_1 \text{ and } a_2 \leq b_2 \text{ implies } a_1 \sqcap a_2 \leq b_1 \sqcap b_2$$

for any $a_1, a_2, b_1, b_2 \in \Im$.

Proof. Suppose the contrary, that is, there are $a_1, a_2, b_1, b_2 \in \Im$ such that

$$a_1 \leq b_1,\ a_2 \leq b_2 \text{ and } a_1 \sqcap a_2 \not\leq b_1 \sqcap b_2.$$

Let us consider the following cases.

Case: $a_1 \sqcap a_2 = \top$ and $b_1 \sqcap b_2 = \bot$.
Then $a_1 = a_2 = \top$, $b_1, b_2 \in \{\top, \boldsymbol{t}\}$ and, hence, $b_1 \sqcap b_2 \in \{\top, \boldsymbol{t}\}$.

Case: $a_1 \sqcap a_2 = \bot$ and $b_1 \sqcap b_2 = \top$.
Then $b_1 = b_2 = \top$ and, hence, $a_1, a_2 \in \{\boldsymbol{f}, \top\}$. Therefore, $a_1 \sqcap a_2 \in \{\boldsymbol{f}, \top\}$.

Case: $a_1 \sqcap a_2 = \top$ and $b_1 \sqcap b_2 = \boldsymbol{f}$.
Then $a_1 = a_2 = \top$ and, hence, $b_1, b_2 \in \{\top, \boldsymbol{t}\}$. Therefore, $b_1 \sqcap b_2 \in \{\top, \boldsymbol{t}\}$.

Case: $a_1 \sqcap a_2 = \bot$ and $b_1 \sqcap b_2 = \boldsymbol{f}$.
Consider two subcases:

a) $a_1 = b_1 = \boldsymbol{f}$ and $a_2, b_2 \in \{\boldsymbol{f}, \top\}$, b) $a_2 = b_2 = \boldsymbol{f}$ and $a_1, b_1 \in \{\boldsymbol{f}, \top\}$.

In both subcases we receive $a_1 \sqcap a_2 = \boldsymbol{f}$.

Case: $a_1 \sqcap a_2 = \boldsymbol{t}$ and $b_1 \sqcap b_2 = \boldsymbol{f}$.

Again, we have the same subcases as in previous case. And again, we receive $a_1 \sqcap a_2 = \boldsymbol{f}$.

Case: $a_1 \sqcap a_2 = \boldsymbol{t}$ and $b_1 \sqcap b_2 = \top$.

Then $b_1 = b_2 = \top$ and $a_1, a_2 \in \{\boldsymbol{f}, \top\}$. Therefore, $a_1 \sqcap a_2 \in \{\boldsymbol{f}, \top\}$.

Finally, case: $a_1 \sqcap a_2 = \boldsymbol{t}$ and $b_1 \sqcap b_2 = \bot$.

Then $a_1, a_2 \in \{\top, \boldsymbol{t}\}$ and, hence, $b_1, b_2 \in \{\top, \boldsymbol{t}\}$. Therefore, $b_1 \sqcap b_2 \in \{\top, \boldsymbol{t}\}$.

All cases lead to contradictions.

The reader can find motivations of the following definitions in [Bel 75, Bel 76] and [KM 93].

All the setups form the lattice AS with the order:

$$s \leq s_1 \text{ if and only if } s(\pi) \sqsubseteq s_1(\pi) \text{ for every } \pi \in \mathsf{Var}.$$

A setup s is *finite* if the set $\{\pi \mid s(\pi) \neq \bot\}$ is finite. An *epistemic state* is a set of setups. We will often use epistemic states of the form $Tset(A)$ and $Fset(A)$ for any formula A, defined as follows:

$$Tset(A) \stackrel{\text{def}}{=} \{s \mid s \in \mathsf{AS}, \boldsymbol{t} \sqsubseteq s(A), V(s) \subseteq V(A)\},$$

$$Fset(A) \stackrel{\text{def}}{=} \{s \mid s \in \mathsf{AS}, \boldsymbol{f} \sqsubseteq s(A), V(s) \subseteq V(A)\},$$

where $V(s) = \{\pi \mid s(\pi) \neq \bot\}$ and $V(A)$ is the set of variables in A. An *assignment* of formula A in epistemic state ε is defined as follows:

$$\varepsilon(A) \stackrel{\text{def}}{=} \sqcap\{s(A) \mid s \in \varepsilon\}.$$

A finite set of finite setups is called a *finite* (epistemic) *state*. Notice that $Tset(A)$ is a finite epistemic state for any formula A. For any finite epistemic state ε, the set $m(\varepsilon)$ of minimal elements in ε is nonempty, because of Descending Chain Condition. The computer's knowledge is supposed to be represented by means of *minimal* (epistemic) *states* — nonempty finite sets of incomparable finite setups. Thus, a finite state is minimal if and only if $m(\varepsilon) = \varepsilon$. All the minimal states form the lattice AFE with order defined as follows:

$$\varepsilon' \leq \varepsilon \text{ if and only if for any } s \in \varepsilon \text{ there is } s' \in \varepsilon' \text{ such that } s' \leq s.$$

We will consider so defined relation as a relation on the set of all epistemic states as well. Though in this case, it does not determine a lattice. For the purpose of introducing knowledge revision procedures we define the lattice AGE of *generalized* (epistemic) *states*, that is, the collections $\overline{\varepsilon}$ of the form $\{\varepsilon' \,|\varepsilon' \equiv \varepsilon\}$ for every epistemic state ε, where

$$\varepsilon' \equiv \varepsilon \text{ means } \varepsilon'(A) = \varepsilon(A) \text{ for any formula } A,$$

with the order:

$$\overline{\varepsilon} \leq \overline{\varepsilon_1} \text{ if and only if } \varepsilon(A) \sqsubseteq \varepsilon_1(A) \text{ for every formula } A.$$

It is obvious that if $\varepsilon = \{s\}$, where $s(\pi) = \top$ for every $\pi \in \mathsf{Var}$, then $\overline{\varepsilon}$ is *the unit* in the lattice AGE.

We call an operation $F : \mathsf{AGE} \to \mathsf{AGE}$ *continuous* (more precisely, *Scott-continuous*) if F is a continuous operation with respect to the Scott topology [Sco 72, GHKLMS 80] on AGE. An element $\overline{\varepsilon}$ in AGE is called a *fixed point of the operation* F if $F(\overline{\varepsilon}) = \overline{\varepsilon}$. Let $\mathcal{U} \subseteq \mathsf{AGE}$. An operation F is called $\mathcal{U}$*-stable* if every element in $\mathcal{U}$ is a fixed point of F.

Proposition 2 *Let F be a continuous operation. Then F is AGE-stable if and only if it is AFE-stable.*

Proof. Let $\overline{\varepsilon_0} \in \mathsf{AGE}$. According to the Theorem 6.4 and the embedding $\varepsilon \mapsto \overline{\varepsilon}$ from the proof of the Theorem 4.3 both from [KM 93] we know that

$$\overline{\varepsilon_0} = \sqcup\{\varepsilon \,|\varepsilon \in \mathsf{AFE}, \varepsilon \ll \overline{\varepsilon_0}\}.$$

Then by means of the Scott's limit theorem (the Proposition 2.5 in [Sco 72] or the Proposition II-2.1 in [GHKLMS 80]),

$$F(\overline{\varepsilon_0}) = \sqcup\{F(\varepsilon)\,|\varepsilon \in \mathsf{AFE}, \varepsilon \ll \overline{\varepsilon_0}\} = \overline{\varepsilon_0}.$$

We will further deal with restrictions, more precisely, with down-restrictions. Therefore, we recall that the *down-restriction* of a setup s over a set of variables V (V*-down-restriction*, for short) is the setup $s^{V\perp}$ defined as follows:

$$s^{V\perp} \overset{\text{def}}{=} \begin{cases} s(\pi) & \text{for } \pi \in V \\ \perp & \text{for } \pi \notin V. \end{cases}$$

Then, we define down-restrictions of epistemic states, ordinary and generalized, in the following way:

$$\varepsilon^{V\perp} \stackrel{\text{def}}{=} \left\{ s^{V\perp} | s \in \varepsilon \right\} \text{ and } \overline{\varepsilon}^{V\perp} \stackrel{\text{def}}{=} \overline{\varepsilon^{V\perp}}$$

(cf. [KM 93]).

3 Global Rules

The sequence $A \to B$, where A and B are formulas, is called a (*global*) *rule*. With every rule $A \to B$, we associate the operation $[A \to B]$ on the lattice AGE, which is Scott-continuous on AGE and coordinated with the basis AFE in the sense that all results of the operation from points in AFE belongs to AFE (see the precise definition and details in [KM 93]; comp. [Bel 75]). Informally, the operation $[A \to B]$ changes a current state of the computer with respect to the condition **if** A **then** B, giving, thereby, a procedural interpretation of entailment.

Being built into the intelligent system, the global rule acts as the reasoning device of the system. It does so properly if this rule itself is not a part of the system's procedural knowledge. However, if the rule $A \to B$ is represented by the operation $[A \to B]$ which is AFE-stable, it is hardly worthwhile to include such superfluous information.

Thus, we call the rule $A \to B$ *proper* (or *explicit*) if the corresponding operation $[A \to B]$ is not AFE-stable, or not AGE-stable which is equivalent to the former in view of the Proposition 2. Otherwise, we call the rule *improper* (or *implicit*). In other words, an implicit rule generates an operation which never changes the system's epistemic state. We could say that all the implicit rules constitute as a whole the system's *internal logic* [1].

The following theorem is a criterion which gives deciding procedures for the class of proper global rules.

Theorem 1 *For any formulas A and B the following conditions are equivalent:*

i) $\varepsilon(A) \leq \varepsilon(B)$ *for every* $\varepsilon \in \mathsf{AFE}$*;*

[1] Conception of logic as reasoning device appears to have first been expressed in modern time by J.Łukasiewics in his seminars in the 1920s (cf. [Pra 65], p. 98). The importance of this point of view for artificial intelligence has been reiteratively emphasized by David Israel (cf., for example, [Isr 93]).

ii) $\overline{\varepsilon}(A) \leq \overline{\varepsilon}(B)$ *for every* $\overline{\varepsilon} \in \mathsf{AGE}$*;*

iii) $[A \rightarrow B]$ *is improper;*

iv) $s(A) \leq s(B)$ *for every* $s \in \mathsf{AS}$*;*

v) $\vdash_{E_{fde}} A \rightarrow B$*;*
here $\leq$ *means the order on* $\mathsf{L4}$*.*

Proof. $i) \Rightarrow ii)$ follows immediately from the following lemma.

Lemma 1.1 *For every epistemic state* ε *and any list of formulas* $A_1, \ldots, A_n$*, there is a minimal state* ε_1*, i.e.* $\varepsilon_1 \in \mathsf{AFE}$*, such that* $\varepsilon_1(A_i) = \varepsilon(A_i)$ *for every* $i = 1, \ldots, n$*.*

Proof. Let V mean $V(A_1) \cup \ldots \cup V(A_n)$. Then $\varepsilon^{V\perp}$ is a finite epistemic state. Denote $m(\varepsilon^{V\perp})$ via ε_1. By Proposition 4 in [KM 93], we have

$$\varepsilon(A_i) = \varepsilon^{V\perp}(A_i) \text{ for } i = 1, \ldots, n.$$

And by Theorem 3.1 in [KM 93], we have $\varepsilon_1 \equiv \varepsilon^{V\perp}$. It is obvious that $\varepsilon_1 \in \mathsf{AFE}$.

$ii) \Rightarrow i)$ follows from the Theorem 4.3 in [KM 93].

$i) \Leftrightarrow iv)$ follows from the Proposition 1 and the Proposition 4 in [KM 93].

$i) \Rightarrow iii)$ Assume that $i)$ is true. Then, so is $iv)$. Let $\varepsilon \in \mathsf{AFE}$. Then $[A \rightarrow B]'(\varepsilon)$ is a finite state, where

$$[A \rightarrow B]'(\varepsilon) \stackrel{\text{def}}{=} \cup \Big\{ [A \rightarrow B]'(s) \,|\, s \in \varepsilon \Big\},$$

and

$$[A \rightarrow B]'(s) \stackrel{\text{def}}{=} \begin{cases} \{s \sqcup s' \,|\, s' \in m(Tset(B))\} & \text{if } \boldsymbol{t} \sqsubseteq s(A) \\ \{s\} & \text{otherwise} \end{cases} \tag{1}$$

(cf. [KM 93]).
Using the Theorem 4.3, the Theorem 7.7 and the Theorem 3.1 all in [KM 93], we conclude that $[A \rightarrow B](\varepsilon) = m\Big([A \rightarrow B]'(\varepsilon)\Big)$. Now we shall show that

$$m\Big([A \rightarrow B]'(\varepsilon)\Big) = \varepsilon. \tag{2}$$

Let $s \in \varepsilon$. Consider two cases.

Case 1: $\boldsymbol{t} \sqsubseteq s\,(A)$.
Since $s\,(A) \leq s\,(B)$, by means $v)$ we have $s\,(B) \in \{\boldsymbol{t}, \top\}$, i.e. $\boldsymbol{t} \sqsubseteq s\,(B)$. By the Proposition 4 in [KM 93], $s^{V(B)\perp} \in Tset\,(B)$. And since $s \sqcup s^{V(B)\perp} = s$. Therefore, by definition $s \in [A \rightarrow B]'(\varepsilon)$. Hence, $m\Big([A \rightarrow B]'(\varepsilon)\Big) \leq \varepsilon$.

Case 2: $\boldsymbol{t} \not\sqsubseteq s\,(A)$.
By definition we have $[A \rightarrow B]'(s) = \{s\}$. Again, we received $m\Big([A \rightarrow B]'(\varepsilon)\Big) \leq \varepsilon$.

Conversely, it is obvious that $\varepsilon \leq m\Big([A \rightarrow B]'(\varepsilon)\Big)$. So, the equation (2) is proved.

$iii) \Rightarrow iv)$. We shall previously prove two lemmas.
Define for any $\tau \in \Im$,

$$\tau^* \stackrel{\text{def}}{=} \begin{cases} \tau & \text{if } \tau \in \{\boldsymbol{t}, \boldsymbol{f}\} \\ \perp & \text{if } \tau = \top \\ \top & \text{if } \tau = \perp \end{cases}$$

Lemma 1.2 *The operation $*$ commutes with operations $\wedge$, $\vee$ and $\neg$, that is,*

$$(\tau_1 \wedge \tau_2)^* = \tau_1^* \wedge \tau_2^*,\ (\tau_1 \vee \tau_2)^* = \tau_1^* \vee \tau_2^* \text{ and } (\neg\tau)^* = \neg\tau^*$$

hold for any $\tau_1, \tau_2 \in \Im$.

Proof is extracted from the following two tables. We should merely remember about idempotency and commutativity of operations $\wedge$ and $\vee$ in $\mathsf{L4}$.

τ_1	τ_2	τ_1^*	τ_2^*	$\tau_1 \wedge \tau_2$	$\tau_1 \vee \tau_2$	$\tau_1^* \wedge \tau_2^*$	$\tau_1^* \vee \tau_2^*$	$(\tau_1 \wedge \tau_2)^*$	$(\tau_1 \vee \tau_2)^*$
$\boldsymbol{f}$	$\boldsymbol{t}$	$\boldsymbol{f}$	$\boldsymbol{t}$	$\boldsymbol{f}$	$\boldsymbol{t}$	$\boldsymbol{f}$	$\boldsymbol{t}$	$\boldsymbol{f}$	$\boldsymbol{t}$
$\boldsymbol{f}$	$\bot$	$\boldsymbol{f}$	$\top$	$\boldsymbol{f}$	$\bot$	$\boldsymbol{f}$	$\top$	$\boldsymbol{f}$	$\top$
$\boldsymbol{f}$	$\top$	$\boldsymbol{f}$	$\bot$	$\boldsymbol{f}$	$\top$	$\boldsymbol{f}$	$\bot$	$\boldsymbol{f}$	$\bot$
$\boldsymbol{t}$	$\bot$	$\boldsymbol{t}$	$\top$	$\bot$	$\boldsymbol{t}$	$\top$	$\boldsymbol{t}$	$\top$	$\boldsymbol{t}$
$\boldsymbol{t}$	$\top$	$\boldsymbol{t}$	$\bot$	$\top$	$\boldsymbol{t}$	$\bot$	$\boldsymbol{t}$	$\bot$	$\boldsymbol{t}$
$\bot$	$\top$	$\top$	$\bot$	$\boldsymbol{f}$	$\boldsymbol{t}$	$\boldsymbol{f}$	$\boldsymbol{t}$	$\boldsymbol{f}$	$\boldsymbol{t}$

τ	τ^*	$\neg\tau$	$\neg\tau^*$	$(\neg\tau)^*$
$\boldsymbol{t}$	$\boldsymbol{t}$	$\boldsymbol{f}$	$\boldsymbol{f}$	$\boldsymbol{f}$
$\boldsymbol{f}$	$\boldsymbol{f}$	$\boldsymbol{t}$	$\boldsymbol{t}$	$\boldsymbol{t}$
$\bot$	$\top$	$\bot$	$\top$	$\top$
$\top$	$\bot$	$\top$	$\bot$	$\bot$

Let us define $s^*(\pi)$ as meaning $(s(\pi))^*$ for any $\pi \in \mathsf{Var}$.

Lemma 1.3 *For any formula A and setup s, the equation $s^*(A) = (s(A))^*$ holds.*

Proof (by induction on the length of A) follows from the Lemma 1.2.

Now suppose that $s(A) \not\leq s(B)$. By virtue of Proposition 4 in [KM 93], we may count that $V(s) \subseteq V(A) \cup V(B)$. Let ε be $\{s\}$, and consider two cases.

Case 1: $s(A) \in \{\top, \boldsymbol{t}\}$ and $s(B) \in \{\boldsymbol{f}, \bot\}$.
Then $[A \to B]'(\varepsilon) = [A \to B]'(s) = \{s \sqcup s' \,|s' \in Tset(B)\}$. If it was the case that $s \in [A \to B]'(s)$, then there would be $s' \in m(Tset(B))$ such that $s' \leq s$. It would follow $\boldsymbol{t} \sqsubseteq s'(B) \sqsubseteq s(B)$. A contradiction. Thus, $s \notin [A \to B]'(\varepsilon)$ and, hence, $s \notin m\Big([A \to B]'(\varepsilon)\Big)$, i.e. $\varepsilon \neq [A \to B](\varepsilon)$.

Case 2: $s(A) \in \{\bot, \boldsymbol{t}\}$ and $s(B) \in \{\boldsymbol{f}, \top\}$.
Then in virtue of the Lemma (1.3), $s^*(A) \in \{\top, \boldsymbol{t}\}$ and $s^*(B) \in \{\boldsymbol{f}, \bot\}$. Consider the setup s' defined as $(s^*)^{V(A) \cup V(B) \bot}$. According to the Proposition 4 in [KM 93], $s'(A) \in \{\top, \boldsymbol{t}\}$ and $s'(B) \in \{\boldsymbol{f}, \bot\}$, and, moreover, $V(s') \subseteq V(A) \cup V(B)$. Thus, we reduce to the case 1.

$iv) \Leftrightarrow v)$. This equivalence is proved in [AB 75]. Also, see [Bel 75].

The proof of the Theorem 1 is completed.

4 Local Rules

Introducing new rules poses another question concerning the modification of the system's imperative knowledge. Indeed, we could want to know if some rule, say $A \rightarrow B$, does or does not lead to a change of the minimal state ε. From the logical point of view, we could say that A *implies* B *in a state* ε (symbolically, $A \stackrel{\varepsilon}{\rightarrow} B$ is true), if $[A \rightarrow B](\varepsilon) = \varepsilon$. Such an implication may stop to be true after the system has received new information as shown in the following example [2].

Example *Let s be a setup such that $s(\pi) = \boldsymbol{t}$ if and only if $\pi = p_1$ or $\pi = p_2$; otherwise $s(\pi) = \bot$. Furthermore, let ε be $\{s\}$. Then, though $p_1 \stackrel{\varepsilon}{\rightarrow} p_2$, but not $p_1 \stackrel{[\neg p_1](\varepsilon)}{\rightarrow} p_2$.*

Proof. By definition we have

$$[p_1 \rightarrow p_2]'(\varepsilon) = [p_1 \rightarrow p_2]'(s) = \{s \sqcup s' \,|s' \in m(Tset(p_2))\}$$

Let s_1 be a setup such that $s_1(p_2) = \boldsymbol{t}$ and $s_1(\pi) = \bot$ for $\pi \neq p_2$. It is easy to observe that $m(Tset(p_2)) = \{s_1\}$. Thus,

$$[p_1 \rightarrow p_2]'(\varepsilon) = \{s \sqcup s_1\} = \{s\}$$

Hence, $[p_1 \rightarrow p_2](\varepsilon) = \varepsilon$, i.e. $p_1 \stackrel{\varepsilon}{\rightarrow} p_2$.

On the other hand, let ε_1 be $[\neg p_1](\varepsilon)$. We know from [KM 93] that

$$\varepsilon_1 = \{s\} \sqcup m(Fset(p_1)) = \{s\} \sqcup \{s_2\} = \{s_3\},$$

[2]We write below $[A]$ for A^+ in [KM 93] for any formula A.

where $s_2(\pi) = \begin{cases} \boldsymbol{f} & \text{if } \pi = p_1 \\ \bot & \text{otherwise} \end{cases}$, and $s_3(\pi) = \begin{cases} \top & \text{if } \pi = p_1 \\ \boldsymbol{t} & \text{if } \pi = p_2 \\ \bot & \text{otherwise} \end{cases}$.

Continuing, we receive:

$$[p_1 \to p_2]'(\varepsilon_1) = [p_1 \to p_2]'(s_3) = \{s_3 \sqcup s' \,|\, s' \in m(Tset\,(p_2))\} = \{s_3 \sqcup s_1\} = \{s_4\},$$

where $s_4(\pi) = \begin{cases} \top & \text{if } \pi = p_1 \\ \boldsymbol{t} & \text{if } \pi = p_2 \\ \bot & \text{otherwise.} \end{cases}$ So, $s_3 \notin [p_1 \to p_2]'(\varepsilon_3)$. Hence, $p_1 \overset{\varepsilon_1}{\to} p_2$ does not hold.

From the procedural point of view, $A \overset{\varepsilon}{\to} B$ $(=[A \to B](\varepsilon))$ is an operation dependent on three parameters. We call it a *local* rule. If $A \to B$ is a proper rule, one can address whether or not it leads to an increase in a current state of the computer's knowledge, that is, whether $A \overset{\varepsilon}{\to} B$ holds. We will further seek to find a criterion to effectively decide this problem.

For any epistemic state ε and formula A, we denote:

$$\varepsilon_A^+ \overset{\text{def}}{=} \{s \,|\, s \in \varepsilon,\, \boldsymbol{t} \sqsubseteq s\,(A)\} \text{ and } \varepsilon_A \overset{\text{def}}{=} \begin{cases} \varepsilon_A^+ & \text{if } \varepsilon_A^+ \neq \emptyset \\ \text{the unit in } \mathsf{AGE} & \text{otherwise.} \end{cases}$$

Theorem 2 *Let ε be a minimal state (i.e. $\varepsilon \in \mathsf{AFE}$). Then the following conditions are equivalent:*

i) $[A \to B](\varepsilon) = \varepsilon$;

ii) $\varepsilon = m\Big(\cup\Big\{[A \to B]'(s) \,|\, s \in \varepsilon\Big\}\Big)$;

iii) $\varepsilon \subseteq \cup\Big\{[A \to B]'(s) \,|\, s \in \varepsilon\Big\}$;

iv) $\varepsilon_A^+ \subseteq \Big\{s \sqcup s' \,\Big|\, s \in \varepsilon_A^+,\, s' \in m(Tset\,(B))\Big\}$;

v) $m(Tset\,(B)) \leq \varepsilon_A$.

Proof. $i) \Leftrightarrow ii)$ follows immediately from the equality

$$[A \to B](\varepsilon) = m\Big(\cup\Big\{[A \to B]'(s) \,|\, s \in \varepsilon\Big\}\Big),$$

which is proved in [KM 93].

$ii) \Leftrightarrow iii)$. It is obviously enough to prove the implication $iii) \Rightarrow ii)$. So, assume

$$\varepsilon \subseteq \cup\Big\{[A \to B]'(s) \,|\, s \in \varepsilon\Big\}. \tag{3}$$

Let $s_0 \in \varepsilon$. Then $s_0 \in \cup\Big\{[A \to B]'(s)\,|s \in \varepsilon\Big\}$. Because of Descending Chain Condition, there is a state $s_1 \in m\Big(\cup\Big\{[A \to B]'(s)\,|s \in \varepsilon\Big\}\Big)$ such that $s_1 \leq s$. Now, we notice that

$$\varepsilon \leq \cup\Big\{[A \to B]'(s)\,|s \in \varepsilon\Big\}. \tag{4}$$

holds, in fact, for any epistemic state ε (cf. (1)). In virtue of (4), there is $s_2 \in \varepsilon$ such that $s_2 \leq s_1$. Thus, we have both $s_2 \leq s_1 \leq s$ and $s_2, s \in \varepsilon$. However, ε is an antichain. Hence, $s_2 = s_1 = s$. The latter follows that $s \in m\Big(\cup\Big\{[A \to B]'(s)\,|s \in \varepsilon\Big\}\Big)$.

Now suppose that

$$s_0 \in m\Big(\cup\Big\{[A \to B]'(s)\,|s \in \varepsilon\Big\}\Big)$$

According to (3), there is $s_1 \in \varepsilon$ such that $s_1 \leq s_0$. By means (4), we conclude that

$$s_1 \in \cup\Big\{[A \to B]'(s)\,|s \in \varepsilon\Big\}.$$

Again, because of Descending Chain Condition, there is

$$s_2 \in m\Big(\cup\Big\{[A \to B]'(s)\,|s \in \varepsilon\Big\}\Big)$$

such that $s_2 \leq s_1$. However, $m\Big(\cup\Big\{[A \to B]'(s)\,|s \in \varepsilon\Big\}\Big)$ is a minimal state (cf. [KM 93]). Therefore, $s_2 = s_1 = s_0$ and, hence, $s_0 \in \varepsilon$.

iii) $\Leftrightarrow$ *iv*). Notice that

$$\cup\Big\{[A \to B]'(s)\,|s \in \varepsilon\Big\} = (\varepsilon \setminus \varepsilon_A^+) \cup \cup\Big\{s \sqcup s' \,\Big|\, s \in \varepsilon_A^+,\ s' \in m(Tset(B))\Big\}$$

(cf. (1)).

iv) $\Leftrightarrow$ *v*). Assume that

$$\varepsilon_A^+ \subseteq \Big\{s \sqcup s' \,\Big|\, s \in \varepsilon_A^+,\ s' \in m(Tset(B))\Big\}$$

Let $s_0 \in \varepsilon_A$. Then there exist $s_1 \in \varepsilon_A^+$ and $s_2 \in m(Tset(B))$ such that $s_0 = s_1 \sqcup s_2$. Therefore, $s_1 \leq s_0$. However, ε being a minimal state is an antichain. Hence, $s_0 = s_1$. Therefore, $s_2 \leq s_0$.

Now assume that $m(Tset(B)) \leq \varepsilon_A$, and let $s_0 \in \varepsilon_A^+$. From our premise follows that there is $s_1 \in m(Tset(B))$ such that $s_1 \leq s_0$, that is, $s_0 \sqcup s_1 = s_0$. Hence

$$s_0 \in \cup\Big\{s \sqcup s' \,\Big|\, s \in \varepsilon_A^+,\ s' \in m(Tset(B))\Big\}.$$

The proof of the Theorem 2 is completed.

We will bind with every setup s a partial function π^s (here: $\pi \in \mathsf{Var}$) as follows:

$$\pi^s \stackrel{\text{def}}{=} \begin{cases} \pi & \text{if } s(\pi) = \boldsymbol{t} \\ \neg\pi & \text{if } s(\pi) = \boldsymbol{f} \\ \pi \wedge \neg\pi & \text{if } s(\pi) = \top. \end{cases}$$

For the next Theorem 3, we need the following lemma.

Lemma 2.1 *Let s and s_1 be setups. If $s \leq s_1$ then $\vdash_{E_{fde}} \pi^{s_1} \to \pi^s$ for every $\pi \in V(s)$.*

Proof. We have to consider five possible cases:

$$\pi^{s_1} \to \pi^s = \begin{cases} \pi \to \pi \\ \pi \wedge \neg\pi \to \pi \\ \neg\pi \to \neg\pi \\ \pi \wedge \neg\pi \to \neg\pi \\ \pi \wedge \neg\pi \to \pi \wedge \neg\pi. \end{cases}$$

All right entailments are derived in E_{fde} (cf. [AB 75]).

Theorem 3 *Let ε and ε' be finite states. Then, $\varepsilon' \leq \varepsilon$ if and only if*

$$\vdash_{E_{fde}} \vee\{\wedge\{\pi^s \,|\pi \in V(s)\} \,|s \in \varepsilon\} \to \vee\{\wedge\{\pi^s \,|\pi \in V(s)\} \,|s \in \varepsilon'\}.$$

Proof. Suppose $\varepsilon' \leq \varepsilon$. According to the Theorem 1, it is enough to prove that

$$s(\vee\{\wedge\{\pi^s \,|\pi \in V(s)\} \,|s \in \varepsilon\}) \leq s(\vee\{\wedge\{\pi^s \,|\pi \in V(s)\} \,|s \in \varepsilon'\})$$

for any setup s. Let s_0 be any fixed setup. Let us consider any function $f : \varepsilon \to \varepsilon'$ such that $f(s) \leq s$ for any $s \in \varepsilon$. Notice that in virtue of the previous Lemma 2.1, $\vdash_{E_{fde}} \pi^s \to \pi^{f(s)}$ for any $\pi \in V(f(s))$, and $V(f(s)) \subseteq V(s)$ for any $s \in \varepsilon$. Thus by means the Theorem 1, we conclude that

$$s_0(\wedge\{\pi^s \,|\pi \in V(f(s))\}) \leq s_0(\wedge\Big\{\pi^{f(s)} \,|\pi \in V(f(s))\Big\})$$

and

$$s_0(\wedge\{\pi^s \,|\pi \in V(s)\}) \leq s_0(\wedge\left\{\pi^{f(s)} \,|\pi \in V(f(s))\right\})$$

for any $s \in \varepsilon$. Hence,

$$s_0(\vee\{\wedge\{\pi^s \,|\pi \in V(s)\} \,|s \in \varepsilon\}) \leq s_0(\vee\{\wedge\{\pi^s \,|\pi \in V(f(s))\} \,|s \in f(\varepsilon)\}).$$

However,

$$s_0(\vee\{\wedge\{\pi^s \,|\pi \in V(s)\} \,|s \in f(\varepsilon)\}) \leq s_0(\vee\{\wedge\{\pi^s \,|\pi \in V(s)\} \,|s \in \varepsilon'\}).$$

Conversely, assume that

$$\vdash_{E_{fde}} \vee\{\wedge\{\pi^s \,|\pi \in V(s)\} \,|s \in \varepsilon\} \to \vee\{\wedge\{\pi^s \,|\pi \in V(s)\} \,|s \in \varepsilon'\}.$$

Let $s_0 \in \varepsilon$. First of all, we notice that $\boldsymbol{t} \sqsubseteq s_0(\pi^{s_0})$ for every $\pi \in V(s_0)$. Hence,

$$\boldsymbol{t} \sqsubseteq s_0(\vee\{\wedge\{\pi^s \,|\pi \in V(s)\} \,|s \in \varepsilon\}).$$

Then in virtue of the Theorem 1,

$$\top \leq s_0(\vee\{\wedge\{\pi^s \,|\pi \in V(s)\} \,|s \in \varepsilon'\})$$

There is a setup $s_1 \in \varepsilon'$ such that

$$\top \leq s_0(\pi^{s_1}) \tag{5}$$

for every $\pi \in V(s_1)$. Firstly, we will prove that

$$V(s_1) \subseteq V(s_0). \tag{6}$$

Suppose the contrary: there exists $\pi \in V(s_1) \setminus V(s_0)$. It is obvious that $s_0 = \bot$ and $s_1 \neq \bot$. The latter follows that π^{s_1} is defined, and the former follows that $s_0(\pi^{s_1}) = \bot$. That is contrary to (5). So, $V(s_1) \subseteq V(s_0)$.

Now we want to prove that $s_1(\pi) \sqsubseteq s_0(\pi)$ for every $\pi \in V(s_1)$. If it is not the case then there is $\pi_0 \in V(s_1)$ (and, hence, $\pi_0 \in V(s_0)$) such that

$s_1(\pi_0)$	$s_0(\pi_0)$	$\pi_0^{s_1}$	$s_0(\pi_0^{s_1})$
$\boldsymbol{f}$	$\boldsymbol{t}$	$\neg\pi_0$	$\boldsymbol{f}$
$\boldsymbol{t}$	$\boldsymbol{f}$	π_0	$\boldsymbol{f}$
$\top$	$\boldsymbol{f}$	$\pi_0 \wedge \neg\pi_0$	$\boldsymbol{f}$
$\top$	$\boldsymbol{t}$	$\pi_0 \wedge \neg\pi_0$	$\boldsymbol{f}$

Again, we receive a contradiction to (5). Thus (5) and (7) give together that $s_1 \leq s_0$. Therefore, we conclude that $\varepsilon' \leq \varepsilon$.

The desirable criterion for local rules is established in the following

Theorem 4 *Let ε be a minimal state, A and B be formulas. Then, $[A \rightarrow B](\varepsilon) = \varepsilon$ if and only if*

$$\vdash_{E_{fde}} \vee\Big\{\wedge\{\pi^s \,|\pi \in V(s)\}\, \Big|s \in \varepsilon_A\Big\} \rightarrow \vee\{\wedge\{\pi^s \,|\pi \in V(s)\}\, |s \in Tset(B)\},$$

provided that $\varepsilon^+_A \neq \emptyset$. If $\varepsilon^+_A = \emptyset$ then the equation $[A \rightarrow B](\varepsilon) = \varepsilon$ is true.

Proof follows immediately from the Theorems 2 and 3.

5 An Epistemic Logic

Now we want to ask what a computer knows about its own state of knowledge and how this state can evolve. We suppose that an intelligent system is formed to include facilities for knowledge revision in the form of continuous operations in the space of AGE, which are coordinated with AFE, and some epistemic logic as the computer's knowledge of its epistemological capacity. This knowledge being embodied in such an intelligent system must be effectively accessible. In our case, it can be expressed in an epistemic language by means of *epistemic formulas* (e-formulas, for short) that are built up from atomic e-formulas of the form $(A : \tau)$, where $\tau \in \Im$, using ordinary propositional connectives $\wedge$, $\vee$, $\neg$ and modality $\diamond$.

Thus, we define $\mathcal{A}$ as being an e-formula whenever $\mathcal{A}$ is an atomic e-formula or of the form $(\mathcal{B} \wedge \mathcal{C})$, $(\mathcal{B} \vee \mathcal{C})$, $\neg\mathcal{B}$ or $\diamond\mathcal{B}$, where $\mathcal{B}$ and $\mathcal{C}$ are e-formulas.

According to [KM 93], some minimal epistemic state is accessible to a current epistemic state if the computer can move from the latter to the former using some rules or operations of the form[3] $[A]$ for some formula A. We limit ourselves here with the consideration of an intelligent system with no rules, though the notion of rule is retained for our knowledge of the improper rules [4].

[3] Recall: we write $[A]$ for A^+ in [KM 93].

[4] As I have recently established in [Mur 94b], that limitation is only apparent: considering all the $\boldsymbol{CAC}$-operations as basic actions leads to the same epistemic logic.

We say that the minimal state ε_1 is accessible from e_0 (symbolically, $\Re\varepsilon_0\varepsilon_1$), if there exists a formula A such that $[A](\varepsilon_0) = \varepsilon_1$, that is, $\varepsilon \sqcup Tset\,(A)$ (cf. [KM 93]). It seems that we need to introduce the transitive closure of the relation of accessibility. However, we prove elsewhere that $\Re$ is transitive and, hence, equivalent to its transitive closure (cf. [Mur 94a]).

Now, we define the notion of *validity* of e-formula in minimal epistemic state (symbolically, $\varepsilon \models \mathcal{A}$) as follows:

$\varepsilon \models (A : \tau)$	iff	$\varepsilon\,(A) = \tau$, where $\tau \in \Im$;
$\varepsilon \models (\mathcal{B} \wedge \mathcal{C})$	iff	$\varepsilon \models \mathcal{B}$ and $\varepsilon \models \mathcal{C}$;
$\varepsilon \models (\mathcal{B} \vee \mathcal{C})$	iff	$\varepsilon \models \mathcal{B}$ or $\varepsilon \models \mathcal{C}$;
$\varepsilon \models \neg\mathcal{B}$	iff	not $\varepsilon \models \mathcal{B}$;
$\varepsilon \models \Diamond\mathcal{B}$	iff	there is a state ε_0 such that $\Re\varepsilon\varepsilon_0$ and $\varepsilon_0 \models \mathcal{B}$.

We regard $\varepsilon \models (A : \tau)$ as meaning that the computer knows that an assignment of the formula A takes the value τ at the state ε.

Let S be the set of all the e-formulas valid in every minimal epistemic state. We may call S a logic, because it is not empty and closed under *modus ponens*. Our purpose is to show that S is decidable.

Theorem 5 *The logic* S *is decidable.*

Let V everywhere below be a finite fixed set of propositional variables. Denote

$$\mathcal{E} \stackrel{\text{def}}{=} \left\{\varepsilon \,\middle|\, \varepsilon \in \mathsf{AFE} \text{ and } \varepsilon^{V\perp} = \varepsilon\right\}$$

and $f : \varepsilon \mapsto m\left(\varepsilon^{V\perp}\right)$ as a mapping from AFE onto $\mathcal{E}$.

Lemma 5.1 $< \mathcal{E}, \sqcup >$ *is an up-semilattice and* f *is a homomorphism from* $< \mathsf{AFE}, \sqcup >$ *onto* $< \mathcal{E}, \sqcup >$*.*

Proof. What we need is to prove the equation:

$$f(\varepsilon) \sqcup f(\varepsilon_1) = f(\varepsilon \sqcup \varepsilon_1)$$

for any minimal states ε and ε_1.

Let δ^{-1} be the reverse mapping to the mapping $\varepsilon \mapsto \overline{\varepsilon}$ introduced in [KM 93]. Using in order the Theorem 4.3, the Theorem 3.1, and the Lemma 6.1, all in [KM 93], we receive:

$$f(\varepsilon) \sqcup f(\varepsilon_1) = m\left(\varepsilon^{V\perp}\right) \sqcup m\left(\varepsilon_1^{V\perp}\right) = \delta^{-1}(\overline{m\left(\varepsilon^{V\perp}\right) \sqcup m\left(\varepsilon_1^{V\perp}\right)}) =$$

$$\delta^{-1}(\overline{m\left(\varepsilon^{V\perp}\right)} \sqcup \overline{m\left(\varepsilon_1^{V\perp}\right)}) = \delta^{-1}(\overline{\varepsilon^{V\perp}} \sqcup \overline{\varepsilon_1^{V\perp}}) = \delta^{-1}(\overline{\varepsilon}^{V\perp} \sqcup \overline{\varepsilon_1}^{V\perp}) =$$

$$\delta^{-1}((\overline{\varepsilon} \sqcup \overline{\varepsilon_1})^{V\perp}) = \delta^{-1}(\overline{\varepsilon \sqcup \varepsilon_1}^{V\perp}) = \delta^{-1}(\overline{\varepsilon \sqcup \varepsilon_1^{V\perp}}) =$$

$$\delta^{-1}(\overline{m\left(\varepsilon \sqcup \varepsilon_1^{V\perp}\right)}) = m\left((\varepsilon \sqcup \varepsilon_1)^{V\perp}\right) = f(\varepsilon \sqcup \varepsilon_1)$$

Now, let us consider $< \mathsf{AFE}, \Re, \models>$ and $< \mathcal{E}, \Re', \models>$ as *(Kripke) model structures*, where $\Re'$ is a binary relation (of accessibility) on $\mathcal{E}$, defined as follows:

$$\Re' \varepsilon \varepsilon_1 \text{ means } \varepsilon_1 = \varepsilon \sqcup m\Big((m(Tset\,(A)))^{V\perp}\Big) \text{ for some formula } A.$$

We recall that a mapping h is called *a pseudo-morphism* (p-morphism, for short) from $< W, R, \models>$ onto $< W', R', \models'>$, if the following holds:

i) h is onto;

ii) Rxy implies $R'h(x)h(y)$;

iii) if $R'h(x)h(y)$ then there is $z \in W$ such that $h(z) = h(y)$ and Rxz;

iv) $x \models \alpha$ iff $h(x) \models \alpha$ for every atomic formula α.

In fact, we will consider a p-morphism, where the item *iv)* is replaced by the following:

iv') $x \models \alpha$ iff $h(x) \models \alpha$ for every α in some fixed set of atomic formulas.

The next proposition gives the basic property of p-morphism.

Proposition 3 ([Seg 68]) *Let h be a* p*-morphism from $< W, R, \models>$ onto $< W', R', \models'>$ with respect to a set Σ of atomic formulas. Then, for any formula φ built up of Σ, the following holds:*

$$x \models \varphi \textit{ if and only if } h(x) \models' \varphi.$$

Proof see, for example, in [HC 84].

Our concern to the notion of p-morphism is in connection to the

Lemma 5.2 *The mapping f is a p-morphism from $< \mathsf{AFE}, \Re, \models>$ onto $< \mathcal{E}, \Re', \models>$.*

Proof. We have to check items $i) - iii)$ and $iv')$ above. The item $i)$ is obvious.

Let $\Re \varepsilon \varepsilon_1$, that is, $\varepsilon_1 = \varepsilon \sqcup m(Tset\,(A))$ for some formula A. By means of the Lemma 5.1, we have the equation $f(\varepsilon_1) = f(\varepsilon) \sqcup f(m(Tset\,(A)))$, that is, $\Re' f(\varepsilon) f(\varepsilon_1)$.

Conversely, beginning with $\Re' f(\varepsilon) f(\varepsilon_1)$, we receive the equation $f(\varepsilon_1) = f(\varepsilon \sqcup m(Tset(A)))$. Let $\varepsilon_2 \overset{\text{def}}{=} \varepsilon \sqcup m(Tset(A))$. It is clear that $\Re \varepsilon \varepsilon_2$.

Finally, in virtue of the Theorem 3.1 and the Proposition 4, both in [KM 93], $f(\varepsilon)(A) = \varepsilon(A)$ for any formula A built up of V.

Let for definiteness $V = \{\pi_1, \ldots, \pi_n\}$, and denote

$$\Lambda = \{\pi_1{}^{\boldsymbol{t}}, \ldots, \pi_n{}^{\boldsymbol{t}}\} \cup \{\pi_1{}^{\boldsymbol{f}}, \ldots, \pi_n{}^{\boldsymbol{f}}\}.$$

Furthermore, we associate with every $\lambda \subseteq \Lambda$ a setup, which will denote by λ^+, defined as follows:

$$\lambda^+(\pi) \overset{\text{def}}{=} \begin{cases} \boldsymbol{t} & \text{if } \pi^{\boldsymbol{t}} \in s \text{ and } \pi^{\boldsymbol{f}} \notin s \\ \boldsymbol{f} & \text{if } \pi^{\boldsymbol{t}} \notin s \text{ and } \pi^{\boldsymbol{f}} \in s \\ \top & \text{if } \pi^{\boldsymbol{t}} \in s \text{ and } \pi^{\boldsymbol{f}} \in s \\ \bot & \text{if } \pi^{\boldsymbol{t}} \notin s \text{ and } \pi^{\boldsymbol{f}} \notin s. \end{cases}$$

For any setup s such that $V(s) \subseteq V$, in turn, we define a subset of Λ as follows:

$$\pi^\tau \in s^- \text{ iff} \begin{cases} \tau = \boldsymbol{t} & \text{and} \quad \boldsymbol{t} \sqsubseteq s(\pi) \\ & \text{or} \\ \tau = \boldsymbol{f} & \text{and} \quad \boldsymbol{f} \sqsubseteq s(\pi). \end{cases}$$

It is easy to check that mappings "+" and "–" establish mutually-reverse one-one-correspondences between $\left\{s \,\middle|\, s^{V\perp} = s\right\}$ and $\{\lambda \,|\, \lambda \subseteq \Lambda\}$. Thereby, $s \leq s_1$ if and only if $s^- \subseteq s_1^-$ for setups in $\mathcal{E}$, and $\lambda \subseteq \lambda_1$ if and only if $\lambda^+ \leq \lambda_1^+$ for the subsets of Λ.

For every $\pi^\tau \in \Lambda$, we define:

$$(\pi^\tau)^* \overset{\text{def}}{=} \begin{cases} \pi & \text{for } \tau = \boldsymbol{t} \\ \neg\pi & \text{for } \tau = \boldsymbol{f}. \end{cases}$$

And for every $\varepsilon \in \mathcal{E}$, define:

$$A_{\mathcal{E}} \overset{\text{def}}{=} \vee\{\wedge\{(\pi^\tau)^* \,|\pi^\tau \in s^-\} \,|s \in \varepsilon\}.$$

Lemma 5.3 *For every $\varepsilon \in \mathcal{E}$, the equation $\varepsilon = m(Tset(A_{\mathcal{E}}))$ holds.*

Proof. It is easy to notice that for every $\pi^\tau \in s^-$, $\boldsymbol{t} \sqsubseteq s((\pi^\tau)^*)$. Therefore,

$$\boldsymbol{t} \sqsubseteq s(\wedge\{(\pi^\tau)^* \,|\pi^\tau \in s^-\}).$$

It implies that

$$\text{if } s \in \varepsilon \text{ then } \boldsymbol{t} \sqsubseteq s(A_{\mathcal{E}}). \tag{7}$$

Notice, thereby, that $V(s) \subseteq V(A_{\mathcal{E}})$. It means that $\varepsilon \subseteq Tset\,(A_{\mathcal{E}})$.

Let $s \in \varepsilon$ and assume $s_1 \leq s$. Then, first of all, $s_1^- \subset s^-$. Notice that for any $\lambda \subseteq \Lambda$, $\lambda = (\lambda \setminus s_1^-) \cup (\lambda \cap s_1^-)$. We have $s^- \setminus s_1^- \neq \emptyset$. For any $\lambda \subseteq \Lambda$ and $\pi^\tau \in \lambda \setminus s_1^-$, consider the following cases:

Case 1: $\tau = \boldsymbol{t}$. Then $(\pi^\tau)^* = \pi$. If $\pi^{\boldsymbol{f}} \in s_1^-$ then $s_1((\pi^\tau)^*) = \boldsymbol{f}$. Otherwise, $s_1((\pi^\tau)^*) = \bot$.

Case 2: $\tau = \boldsymbol{f}$. Then $(\pi^\tau)^* = \neg\pi$. If $\pi^{\boldsymbol{t}} \in s_1^-$ then $s_1((\pi^\tau)^*) = \boldsymbol{f}$, again. Otherwise, $s_1((\pi^\tau)^*) = \neg\pi$.

Thus, for every $\lambda \subseteq \Lambda$,

$$s_1(\wedge\{(\pi^\tau)^* \,|\pi^\tau \in \lambda\}) \in \{\bot, \boldsymbol{f}\},$$

that is, $s_1(A_{\mathcal{E}}) \in \{\bot, \boldsymbol{f}\}$. This completes the proof of that $\varepsilon \subseteq m(Tset\,(A_{\mathcal{E}}))$.

Conversely, let $s \in m(Tset\,(A_{\mathcal{E}}))$. That is, $\boldsymbol{t} \sqsubseteq s(A_{\mathcal{E}})$ and $\boldsymbol{t} \not\sqsubseteq s'(A_{\mathcal{E}})$ for $s' < s$, because of $V(s') \subseteq V(s) \subseteq V(A_{\mathcal{E}})$. It follows that there is a setup $s_1 \in \varepsilon$ such that $\boldsymbol{t} \sqsubseteq s((\pi^\tau)^*)$ for every $\pi^\tau \in s_1^-$. For any $\pi^\tau \in s_1^-$, consider three cases:

Case 1: $s_1(\pi) = \boldsymbol{t}$. Then $(\pi^{\boldsymbol{t}})^* = \pi$ and, hence, $\boldsymbol{t} \sqsubseteq s(\pi)$.

Case 2: $s_1(\pi) = \boldsymbol{f}$. Then $(\pi^{\boldsymbol{f}})^* = \neg\pi$ and, hence, $\boldsymbol{f} \sqsubseteq s(\pi)$.

Case 3: $s_1(\pi) = \top$. Then both $\pi^{\boldsymbol{f}} \in s_1^-$ and $\pi^{\boldsymbol{t}} \in s_1^-$, and, hence, both $\boldsymbol{f} \sqsubseteq s(\pi)$ and $\boldsymbol{t} \sqsubseteq s(\pi)$ hold. Therefore, $s(\pi) = \top$.

As a consequence, we receive $s_1 < s$. In virtue of (7), $\boldsymbol{t} \sqsubseteq s_1(A_{\mathcal{E}})$. Consequently, we have simultaneously $s \in m(Tset\,(A_{\mathcal{E}}))$, $s_1 \leq s$ and $s_1 \in Tset\,(A_{\mathcal{E}})$. It implies the equation $s_1 = s$.

Proof of the Theorem 5. First of all, the set $\mathcal{E}$ can be effectively listed. Then, in virtue of the Lemma 5.3, for any $\varepsilon, \varepsilon_1 \in \mathcal{E}$, if $\varepsilon \leq \varepsilon_1$ then

$$\varepsilon_1 = \varepsilon \sqcup m\Big((m(Tset\,(A_{\varepsilon_1})))^{V\bot}\Big),$$

that is, the restriction of $\leq$ on $\mathcal{E}$ and $\Re'$ coincide. However, the relation $\leq$ on $\mathcal{E}$ is recursive. Hence, in virtue of the Lemma 5.2, for any e-formula $\mathcal{A}$ built up of V, $\mathcal{A}$ is valid in S if and only if $\mathcal{A}$ is true in the model structure $< \mathcal{E}, \leq, \models >$.

Theorem 6 *For any minimal state ε and e-formula $\mathcal{A}$, the relation $\varepsilon \models \mathcal{A}$ is recursively decidable.*

Proof. For fixed ε and $\mathcal{A}$ we can take V so that $V(\varepsilon) \subseteq V$ and all propositional variables of $\mathcal{A}$ be in V. Then, notice that for such V, the mapping f gives the equivalence

$$< \mathsf{AFE}, \Re, \varepsilon > \models \mathcal{A} \text{ if and only if } < \mathcal{E}_V, \Re_V, f(\varepsilon) > \models \mathcal{A}$$

(cf. Lemma 5.2); thereby, $\mathcal{E}_V$ is recursively listed and $\Re_V$ is a recursive relation on $\mathcal{E}_V$.

References

[AB 75] A.R.Anderson and N.D.Belnap, Jr., Entailment: the Logic of Relevance and Necessity, Vol. 1, Princeton, Princeton University Press, 1975.

[ABD 92] A.R.Anderson, N.D.Belnap, Jr., and J.M.Dunn, Entailment: the Logic of Relevance and Necessity, Vol. 2, Princeton, Princeton University Press, 1992.

[Bel 75] N.D.Belnap, A Useful Four-Valued Logic, in: J.M.Dunn and G.Epstein (eds.), Modern Uses of Multiple-Valued Logic, *Proceedings of International Symposium on Multiple-Valued Logic, 5th, Indiana University*, D. Reidel Publ. Co., 1975, pp. 9-37; Also see §81 in [ABD 92].

[Bel 76] N.D.Belnap, How a Computer Should Think, in: G.Ryle (ed.), Contemporary Aspects of Philosophy, *Proceedings of the Oxford International Symposium*, 1975, Oriel Press, 1976, pp. 30-56; Also see §81 in [ABD 92].

[HC 84] G.E.Hughes and M.J.Cresswell, A Companion to Modal Logic, Methuen-London-New York, Methuen & Co., 1984.

[GHKLMS 80] G.Gierz, K.H.Hofmann, K.Keimel, J.D.Lawson, M.Mislove and D.S.Scott, A Compendium of Continuous Lattices, Springer-Verlag, 1980.

[Gin 87] M.L.Ginsberg, Introduction, in: M.I.Ginsberg (ed.), Readings in Nonmonotonic Reasoning, 1987, pp. 1-23.

[Gin 88] M.I.Ginsberg, Multivalued Logics: A Uniform Approach to Inference in Artificial Intelligence, *Computational Intelligence*, Vol. 4, no. 3, 1988, pp. 265–316.

[Isr 93] D.J.Israel, The Role(s) of Logic in Artificial Intelligence, in: D.M.Gabbay, C.J.Hogger and J.A.Robinson (eds.), *Handbook of Logic in Computer Science*, Vol. 1, 1993, Oxford University Press, pp. 1–29.

[KL 89] M.Kifer and E.L.Lozinskii, RI: A Logic for Reasoning with Inconsistency, *Proceedings of 4th IEEE Symposium on Logic in Computer Science*, Asilomar, California, 1989, pp. 253–262.

[KM 90] Y.M.Kaluzhny and A.Yu.Muravitsky, Modification of Epistemic States, *10th Soviet Conference on Mathematical Logic*, Abstracts, 1990, Alma-Ata, p.73 (Russian).

[KM 93] Y.M.Kaluzhny and A.Yu.Muravitsky, A Knowledge Representation Based on the Belnap's Four-Valued Logic, *Journal of Applied Non-Classical Logics*, Vol. 3, no. 2, 1993, pp. 189–203.

[Mur 94a] A.Yu.Muravitsky, A Framework for Knowledge-Based Systems, 1994, submitted to *Journal of Applied Non-Classical Logics*.

[Mur 94b] A.Yu.Muravitsky, Knowledge Representation as Domains, unpublished manuscript, 1994.

[Pra 65] D.Prawitz, Natural Deduction: a Proof-Theoretical Study, Almqvist & Wiksell, Stockholm, 1965.

[PR 89] G.Priest and R.Routley, First Historical Introduction. A Preliminary History of Paraconsistent and Dialethic Approaches, in: G.Priest, R.Routley and J.Norman (eds.), *Paraconsistent Logic. Essays on the Inconsistent*, Philosophia Verlag, München-Hamden-Wien, 1989, pp. 3–75.

[RB 79] N.Rescher and R.Brandom, The Logic of Inconsistency, Rowman & Littlefield, Totowa, 1979.

[Sco 71] D.S.Scott, Outline of a Mathematical Theory of Computation, in: *Proceedings of Princeton Conference on Information Science*, 1971, pp. 169–176.

[Sco 72] D.S.Scott, Continuous Lattices, in: *Lecture Notes in Mathematics*, Vol. 274, Springer-Verlag, 1972, pp. 97–136.

[Seg 68] K.Segerberg, Results in Non-classical Logic, Lund, Berlingska Boktryckeriet, 1968.