Fundamenta Informaticae 34 (2000) 1–38
IOS Press

The PMA and Relativizing Minimal Change for Action Update

Patrick Doherty *

Department of Computer and Information Science
Linköping University
Linköping, Sweden

Witold Łukaszewicz and Ewa Madalińska-Bugaj †

Institute of Informatics
Warsaw University
Warsaw, Poland

Abstract. Recently, a great deal of progress has been made using nonmonotonic temporal logics to formalize reasoning about action and change. In particular, much focus has been placed on the proper representation of non-deterministic actions and the indirect effects of actions. For the latter the use of causal or fluent dependency rule approaches has been dominant. Although much recent effort has also been spent applying the belief revision/update (BR/U) approach to the action and change domain, there has been less progress in dealing with nondeterministic update and indirect effects represented as integrity constraints. We demonstrate that much is to be gained by cross-fertilization between the two paradigms and we show this in the following manner. We first propose a generalization of the PMA, called the modified MPMA which uses intuitions from the TL paradigm to permit representation of nondeterministic update and the use of integrity constraints interpreted as causal or fluent dependency rules. We provide several syntactic characterizations of MPMA, one of which

*The author was partially supported by grants from the Wallenberg Foundation
†The authors were partially supported by the Wallenberg Foundation and by KBN grant 8 T 11C 009 19

is in terms of a simple temporal logic and provide a representation theorem showing equivalence between the two. In constructing the MPMA, we discovered a syntactic anomaly which we call the *redundant atom anomaly* that many TL approaches suffer from. We provide a method for avoiding the problem which is equally applicable across paradigms. We also describe a syntactic characterization of MPMA in terms of Dijkstra semantics. We set up a framework for future generalization of the BR/U approach and conclude with a formal comparison of related approaches.

This paper is a revised and extended version of a paper: P. Doherty, W. Łukaszewicz, E. Madalińska-Bugaj. The PMA and Relativizing Minimal Change for Action Update. In: *Proc. of the 6th Int'l Conference on Principles of Knowledge Representation and Reasoning*. Morgan Kaufmann, 1998, 258-269.

1. Introduction

Recently, much effort has been invested in applying the belief revision/update (BR/U) paradigm to the domain of reasoning about action and change. In a similar vein, much effort has been invested in applying temporal logics (TL) to reasoning about action and change. Unfortunately, there have been few attempts at trying to bridge the gap between the two paradigms by analyzing one approach in terms of the evaluation methodology of the other, or by applying solutions generated in one paradigm to similar open problems in the other paradigm (although, see [3, 29, 26]). We believe that there is much to be gained by a cross-fertilization of the two paradigms and in this paper, we will try to do just that.

We begin by noting that much of the recent research in the temporal reasoning paradigm has focused on two issues: 1) proper representation of non-deterministic actions [21, 25, 18], and 2) proper representation of the indirect effects of actions, where the use of causal or fluent dependency rule approaches has received a great deal of attention [20, 22, 16, 28]. On the other hand, the BR/U approach applied to reasoning about action and change, has not reached any satisfactory consensus on the proper approach to non-deterministic update and is continually plagued with an inability to represent casual constraints in terms of integrity constraints.

Beginning with the PMA, we will apply techniques from the TL paradigm resulting in a generalization of the PMA, called the modified PMA (MPMA), which permits the representation of non-deterministic update together with integrity constraints interpreted as causal constraints. In addition, we provide a number of syntactic characterizations of MPMA with integrity constraints and show that a very simple temporal logic, STL, which is a fragment of PMON and PMON(RC), two well-behaved 1st-order temporal logics assessed correct using the Features and Fluents framework, can be used to compute entailments of the MPMA. The representation theorem we provide opens up the possibility of additional generalizations of the BR/U paradigm for reasoning about action and change.

Another interesting result of this comparative evaluation between paradigms is the discovery that many of already existing temporal logic approaches [18, 20, 16, 27], suffer from a syntactic anomaly which we call the *redundant atom anomaly*, where, if one is not careful about the syntactic form of consequents of action or causal rules, certain arguably unintuitive direct and indirect effects of action invocation may result. In the paper, we point out the problem using MPMA, and provide a remedy which can also be applied to the temporal logics which suffer from similar problems.

The generalization to the PMA can succinctly be described as the *relativization* or weakening of the minimal change policy to a subset of atoms in the language rather than to all atoms. The technique is similar to the use of varied predicates in circumscription or the use of an *occlusion*, *release*, or *noninert* predicate in the temporal logic paradigm. The proposed solution to the use of integrity constraints is to view them as causal constraints representing fluent dependency information along the lines of a number of causal approaches recently proposed for solving the ramification problem.

The paper is structured as follows: In Section 2, we introduce the PMA and discuss some well

known criticisms associated with its use as a formalism for reasoning about action and change. In Section 3, we modify the PMA, by relativizing or weakening the minimal change policy built into the definition of distance between interpretations. We discuss the redundant atom anomaly and provide a means of automatically preprocessing updates to remove redundancy and choose the atoms to be excluded from the minimal change policy. An inefficient syntactic characterization of MPMA in terms of *eliminants* is considered along with an analysis of the MPMA in terms of the Katsuno and Mendelzon [17] postulates. In section 6, we show, by translating the update formula into a simple program and treating the initial knowledge-base as the set of initial states for this program, that the MPMA can be naturally characterized in terms of Dijkstra's semantics, originally developed to reason about programs. In Section 4, we introduce the simple temporal logic, STL, a fragment of PMON(RC), a highly expressive logic used to reason about temporal narratives. In Section 5, we reformulate the MPMA in STL and provide a representation theorem. A decision procedure for MPMA in terms of STL is then provided. In Section 7, we extend the MPMA with integrity constraints represented as causal rules or dependency constraints in a manner similar to that used in [16]. In Section 8, we provide a comparison with related work and conclude with a discussion.

2. The Classical PMA

We present the classical PMA semantics, originally developed by Winslett [30, 31, 32].

2.1. The Language $\mathcal{L}_{pma}$

We start with a language $\mathcal{L}_{pma}$ of classical propositional logic based on a finite fixed set $ATM = \{p, q, r, \ldots\}$ of atoms. Formulas are built in the usual way using the connectives $\wedge, \vee, \neg, \rightarrow, \leftrightarrow$, $\top$ (truth) and $\bot$ (falsity). If α and β are formulas and p is an atom, then we write $\alpha[p \leftarrow \beta]$ to denote the formula which is obtained from α by simultaneously replacing all occurrences of p by β. A *literal* is an atom or its negation.

2.2. Semantics

Interpretations are identified with maximal consistent sets of literals. For any formula α, we write $\mid \alpha \mid$ to denote the set of all *models* of α, *i.e.* interpretations satisfying α. A formula α is said to *correspond* to an interpretation u iff $\mid \alpha \mid = u$. To construct such a formula, it suffices to take the conjunction of all literals occurring in u. Similarly, a formula α is said to correspond to a finite set of interpretations $\{u_1, \ldots, u_n\}$ if $\mid \alpha \mid = \{u_1, \ldots, u_n\}$. To obtain such a formula, it suffices to take the disjuncion $\beta_1 \vee \cdots \vee \beta_n$, where, for each $1 \leq i \leq n$, β_i is the formula corresponding to u_i. For instance, the formula corresponding to the set $\{\{p, \neg q\}, \{p, q\}\}$ is $(p \wedge \neg q) \vee (p \wedge q)$ which is equivalent to p.

Definition 2.1. Let w, v be two interpretations. The *distance* between w and v, written $DIST(w, v)$, is a set of atoms that have different truth-values in w and u. ■

For instance, the distance between $w = \{p, q, r, \neg s\}$ and $v = \{q, s, \neg p, \neg r\}$ is $\{p, r, s\}$.

Definition 2.2. The *update of an interpretation w by a set of interpretations V*, written $w \star V$, is the set of those elements of V that are closest to w, *i.e.* whose distance to w is minimal. More formally:

$$w \star V = \{v \in V : \text{there is no } v' \in V \text{ such that } DIST(w, v') \subset DIST(w, v)\}. \blacksquare$$

For instance, if w is $\{\neg p, \neg q\}$ and $V = \{\{p, \neg q\}, \{q, \neg p\}, \{p, q\}\}$, then $w \star V = \{\{p, \neg q\}, \{q, \neg p\}\}$.

Definition 2.3. The *update of a set of interpretations U by a set of interpretations V*, written $U \star V$, is given by

$$U \star V = \bigcup_{w \in U} w \star V. \blacksquare$$

In the classical PMA, the update $KB \star \alpha$ of a knowledge base KB by a formula α is identified with the formula corresponding to the set of interpretations $|KB| \star |\alpha|$.

2.3. Examples

Example 2.1. Let $ATM = \{p, q\}$, $KB = p$ and $\alpha = p \vee q$. Then $|KB| \star |\alpha| = \{p, \neg q\} \star \{\{p, \neg q\}, \{q, \neg p\}, \{p, q\}\} \cup \{p, q\} \star \{\{p, \neg q\}, \{q, \neg p\}, \{p, q\}\} = \{p, \neg q\} \cup \{p, q\}$.

Thus $KB \star \alpha$ is the formula $(p \wedge \neg q) \vee (p \wedge q)$ which is equivalent to p.

Example 2.2. Let $ATM = \{p, q\}$, $KB = \neg p \wedge \neg q$ and $\alpha = p \vee q$. Then $|KB| \star |\alpha| = \{\neg p, \neg q\} \star \{\{p, \neg q\}, \{q, \neg p\}, \{p, q\}\} = \{\{p, \neg q\}, \{q, \neg p\}\}$.

Thus $KB \star \alpha$ is the formula $(p \wedge \neg q) \vee (\neg p \wedge q)$.

3. The Modified PMA

A number of researchers ([1], [14]) have observed that the classical PMA may lead to unintuitive results, particularly when disjunctive update is involved. To see this, reconsider Example 2.1. If all we know about the world is p and the effect of an action performed is $p \vee q$, then the description of the new state of the world should be $p \vee q$ rather than p. If p was accepted as the description of the new state, then this would imply that our knowledge about the current state makes an unpredictable action predictable. Example 2.2 also illustrates an undesirable phenomenon: updating a knowledge base by the inclusive *or* results in the exclusive *or*. What these two examples have in common is the fact that if an action effect follows from the knowledge base, the action will have no effect. The culprit in both cases is the fact that the PMA minimal change policy should be relaxed somewhat, but in a principled manner. This is the insight gained from the TL community and incorporated in current solutions to similar problems.

In order to incorporate a relaxed minimal change policy for the PMA, we will change it in two ways. Firstly, the distance between two interpretations will always be relativized to a chosen

set of atoms. Intuitively, this set represents atoms that are allowed to vary their values during the action execution. Secondly, the update of an interpretation w by a set of interpretations U will be defined as those elements of U whose distance to w is the empty set. We call this new form of the PMA the *modified* PMA and we denote it by MPMA. The details follow.[1]

Definition 3.1. Let w and v be two interpretations and suppose that P is a set of atoms. The *distance* between w and v wrt P is $DIST(w, v) - P$. ■

For instance, the distance between $\{p, q, r, \neg s\}$ and $\{q, s, \neg p, \neg r\}$ wrt $\{p\}$ is $\{r, s\}$.

Definition 3.2. The *update of an interpretation w by a set of interpretations V* wrt a set of atoms P, written $w \star^P V$, is the set of those elements of V whose distance to w wrt P is $\emptyset$. ■

Definition 3.3. The *update of a set of interpretations U by a set of interpretations V* wrt a set of atoms P, written $U \star^P V$, is given by

$$U \star^P V = \bigcup_{w \in U} w \star^P V. \blacksquare$$

In the modified PMA, the update $KB \star^P \alpha$ of a knowledge base KB by a formula α wrt a set of atoms P is identified with the formula corresponding to the set of interpretations $|KB| \star^P |\alpha|$.

Example 3.1. (continuation of 2.1) To properly deal with the example 2.1, the atoms p and q should be allowed to vary. $|KB| \star^{\{p,q\}} |\alpha| =$

$$\begin{aligned}
&\{p, \neg q\} \star^{\{p,q\}} \{\{p, \neg q\}, \{q, \neg p\}, \{p, q\}\} \cup \\
&\quad \{p, q\} \star^{\{p,q\}} \{\{p, \neg q\}, \{q, \neg p\}, \{p, q\}\} = \\
&\{\{p, \neg q\}, \{q, \neg p\}, \{p, q\}\} \cup \{\{p, \neg q\}, \{q, \neg p\}, \{p, q\}\} = \\
&\quad \{\{p, \neg q\}, \{q, \neg p\}, \{p, q\}\}.
\end{aligned}$$

Thus $KB \star^{\{p,q\}} \alpha$ is the formula $(p \wedge \neg q) \vee (q \wedge \neg p) \vee (p \wedge q)$ which is equivalent to $p \vee q$. ■

Example 3.2. (continuation of 2.2) As before, the variable atoms should be p and q. We leave it to the reader to check that $KB \star^{\{p,q\}} \alpha \equiv p \vee q$. ■

It should be noted that the above specified update operator need not preserve consistency. For instance, if $ATM = \{q\}$, $KB = \neg q$, $\alpha = q$ and $P = \emptyset$, then $KB \star^P \alpha \equiv \bot$. However, as we shall see in the next section, the consistency preservation is guaranteed if variable atoms are properly chosen.

[1] The motivation behind MPMA will become clear when we provide its syntactic characterization in Section 3.2. A policy similar to MPMA is mentioned by Winslett [31], but rejected as not representing minimal change. MPMA also reflects similar minimization policies used in the TL paradigm as we will show.

3.1. Determining Variable Atoms

In this section we discuss the fundamental question: how can one provide a mechanism to determine variable atoms which can be applied automatically and still guarantee intuitively reasonable conclusions?

Consider a knowledge base KB and an update formula α. Since α represents effects of a performed action, all atoms occurring in α are potential candidates for variable atoms. However, α may include redundant atoms and these must not be varied. The next example illustrates the point.

Example 3.3. Suppose that $ATM = \{p, q\}$, $KB = \neg p \wedge q$ and $\alpha = p$. Taking p as the only variable atom, we get $KB \star^{\{p\}} \alpha \equiv p \wedge q$.

Replace now α by its equivalent $\alpha' = (p \wedge q) \vee (p \wedge \neg q)$ with a redundant atom q. Taking p and q as variables, we obtain $KB \star^{\{p,q\}} \alpha' \equiv p$. This result is clearly undesirable. ■

This is an example of what we call the *redundant atom anomaly*. It appears to be a flaw that several formalisms [18, 20, 16], based on the use of an *Occlusion* or *Release* predicate, succumb to. *Occlusion* and *Release* have the role of distinguishing variable fluents from non-variable fluents. In fact, it can be shown that other causal formalisms such as Thielscher [28] suffer from the same flaw when syntactic rules are not preprocessed to remove redundant atoms. Semantic approaches toward modeling action effect axioms such as Sandewall [25], where a *full trajectory normal form* (FTNF) is generated from a model-theoretic description, do not suffer from this problem because there are no redundant atoms in the normal form. On the other hand, the claim that the FTNF can be replaced with a logically equivalent formula does not appear to be correct for those logics dependent on the use of an *Occlude* predicate, where the choice of occluded fluents is dependent on syntax.

Definition 3.4. Let α be a formula. An atom p occurring in α is said to be *redundant* for α iff $\alpha[p \leftarrow \top] \equiv \alpha[p \leftarrow \bot]$.[2] ■

As follows from the above definition, an atom is redundant for a formula iff the logical value of the formula does not depend on the logical value of the atom.

Our choice of variable atoms is to identify them with non-redundant atoms for the update formula. This poses the question of how these atoms are to be selected. Definition 3.4 is rather impractical from the computational point of view. Below, we present a generally more efficient method based on the notion of the *Blake canonical form* of a formula. Our discussion follows Brown [2].

We start with preliminary terminology.

A *term* is either $\top$, $\bot$, or a conjunction of literals in which no atom appears more than once. A formula is said to be in *disjunctive normal form* (*DNF*, for short) if it is a disjunction of

[2]Recall that $\alpha[p \leftarrow \top]$ (resp. $\alpha[p \leftarrow \bot]$) is the formula obtained from α by replacing all occurrences of p by $\top$ (resp. $\bot$).

different terms.[3] It is well-known that each formula can be constructively transformed into its equivalent in *DNF*. We say that a term t_1 *absorbs* a term t_2 if either t_1 is $\top$, or t_2 is $\bot$, or t_1 is a subterm of t_2. For instance, the term p absorbs the term $p \wedge q$. Let α be a formula in *DNF*. We write $ABS(\alpha)$ to denote the formula obtained from α by deleting all absorbed terms. Clearly, α and $ABS(\alpha)$ are equivalent.

Two terms are said to have an *opposition* if one of them contains an atom p and the other the atom $\neg p$. For instance, the terms $\neg q \wedge r$ and $q \wedge s$ have a single opposition, in the atom q.

Suppose that two terms, t_1 and t_2, have exactly one opposition. Then the *consensus* of t_1 and t_2, written $c(t_1, t_2)$, is the term obtained from the conjunction $t_1 \wedge t_2$ by deleting the opposed atoms as well as any repeated atoms. For example, $c(\neg q \wedge p, q \wedge r)$ is $p \wedge r$.

Let α be a formula. The *Blake canonical form of* α, written $BCF(\alpha)$, is the formula obtained from α by the following construction.

(1) Replace α by its disjunctive normal form. Denote the resulting formula by β.
(2) Repeat as long as possible:
If β contains a pair t_1 and t_2 of terms whose consensus exists and no term of β is a subformula of $c(t_1, t_2)$, then $\beta := \beta \vee c(t_1, t_2)$.
(3) Take $ABS(\beta)$. This is $BCF(\alpha)$.

The following results can be found in Brown [2].

Theorem 3.1.
(1) Formulas α and $BCF(\alpha)$ are equivalent.
(2) All atoms occurring in $BCF(\alpha)$ are non-redundant for α. ■

Example 3.4. Let α be $(\neg p \wedge q) \vee (p \wedge r \wedge \neg s) \vee (p \wedge \neg s \wedge \neg q)$. Since α is in disjunctive normal form, $\beta = \alpha$. After performing step (2), the result is,

$$(\neg p \wedge q) \vee (p \wedge r \wedge \neg s) \vee (p \wedge \neg s \wedge \neg q) \vee (r \wedge q \wedge \neg s). \tag{1}$$

Since $ABS((1)) = (1)$, the formula (1) is the Blake canonical form of α. ■

In the sequel, we write $atm(\alpha)$ to denote the set of all atoms occurring in a formula α.

By the MPMA we mean the update $KB *^P \alpha$ of a knowledge base KB by a formula α, where $P = atm(BCF(\alpha))$.

The next theorem shows that the MPMA preserves consistency.

Theorem 3.2. For any knowledge base $KB \not\equiv \bot$ and any formula $\alpha \not\equiv \bot$, $P = atm(BCF(\alpha))$ implies $KB *^P \alpha \not\equiv \bot$. **Proof** Let w be any model of KB and let u be any model of α. (Such models exist because KB and α are both consistent.) Let $l_1, \ldots, l_k$ be the set of all literals

[3] In the logical literature *DNF* is often defined as a disjunction of terms where a term is understood as either $\top$, or $\bot$ or any conjunction of literals. Note, however, that we can always restrict ourselves to terms in which no atom appears more than once: each repeated occurence of an atom p can be removed from a term, whereas any term including p and $\neg p$ can be replaced by $\bot$.

from w not including atoms from P. Let $l'_1, \ldots, l'_m$ be the set of all literals from u containing only atoms from P. Consider the interpretation v consisting of $\{l_1, \ldots, l_k, l'_1, \ldots l'_m\}$. Note that the sets $l_1, \ldots, l_k$ and $l'_1, \ldots, l'_m$ are disjoint, so that v is well-defined. Since P contains all non-redundant atoms occurring in α, the truth-value of α does not depend on the literals $l_1, \ldots, l_k$. Finally observe that all atoms from P have the same truth-values in u and v. Thus v is also a model of α. Moreover, the distance between w and v wrt P is $\emptyset$, and so $v \in |KB| \star^P |\alpha|$. We therefore conclude that $|KB| \star^P |\alpha| \neq \emptyset$, and thus $KB *^P \alpha \not\equiv \bot$. ■

3.2. Syntactic Characterization of the MPMA

In this section, we present a syntactic characterization of the MPMA. We start with some terminology.

Let p be an atom and suppose that α is a formula. We write $\exists p.\alpha$ to denote the formula $\alpha[p \leftarrow \top] \vee \alpha[p \leftarrow \bot]$. If $P = \{p_1, \ldots, p_n\}$ is a set of atoms and α is a formula, then $\exists P.\alpha$ stands for $\exists p_1 \cdots \exists p_n.\alpha$.

A formula of the form $\exists P.\alpha$, where $P = \{p_1, \ldots, p_n\}$, is called an *eliminant of* $\{p_1, \ldots, p_n\}$ *in* α. Intuitively, such an eliminant can be viewed as a formula representing the same knowledge as α about all atoms from $ATM - P$ and providing no information about the atoms in P. The reader interested in a detailed theory of eliminants should consult Brown [2].

The following result, which easily follows from the definition of an eliminant, provides its semantical characterization.

Theorem 3.3. Let α be a formula and suppose that P is a set of atoms. Then

$$|\exists P.\alpha| = \{u : \text{ exists } w \in |\alpha| \text{ such that } DIST(w, u) - P = \emptyset\}. \blacksquare$$

We are now ready to provide a syntactic characterization of the MPMA.

Theorem 3.4. Let KB, α and $P = atm(BCF(\alpha))$ be a knowledge base, a formula and a set of atoms, respectively. Then $KB \star^P \alpha \equiv \alpha \wedge \exists P.KB$. **Proof** It suffices to show

$$|KB| \star^P |\alpha| = |\alpha \wedge \exists P.KB| .$$

$|KB| \star^P |\alpha| = \bigcup_{(w \in KB)} w \star^P \alpha = \bigcup_{(w \in KB)} \{u \in |\alpha| : DIST(u, w) - P = \emptyset\} =$ (by Theorem 3.3) $|\alpha| \cap |\exists P.KB| = |\alpha \wedge \exists P.KB|$. ■

Theorem 3.4 clearly shows how the MPMA works. First, we select the atoms that may vary their values when the action corresponding to the update formula α is performed. Next, we weaken the knowledge base KB by eliminating all those variable atoms. Finally, we strengthen the knowledge base $\exists P.KB$ by combining it with the update formula.

3.3. Computing the MPMA

In view of Theorem 3.4, all we need to compute the MPMA is the ability to compute eliminants. The following method can be found in Brown [2].

Let α be a formula and suppose that $P = \{p_1, \ldots, p_k\}$ is a set of atoms. The eliminant $\exists P.\alpha$ is the formula obtained by the following construction.

(1) Replace α by its disjunctive normal form $t_1 \vee \cdots \vee t_n$, where $t_1, \ldots, t_n$ are terms.
(2) From each t_i $(1 \leq i \leq n)$, remove all occurrences (both positive and negative) of $p_1, \ldots, p_k$. If during this process all literals have been removed from some term t_i, stop and return $\top$ as $\exists P.\alpha$. Otherwise, return the resulting formula as $\exists P.\alpha$.

Remark 3.1. Although the above algorithm is the best known, it is not the most efficient one. A better way to compute eliminants is a propositional version of DLS algorithm, originally developed to reduce second-order formulae to their first-order equivalents [8].

Example 3.5. Let $\alpha = (p \rightarrow q) \wedge (\neg q \vee r)$. We compute $\exists q.\alpha$. Converting α into its disjunctive normal form, we get $(\neg p \wedge \neg q) \vee (\neg p \wedge r) \vee (q \wedge r)$. Eliminating the atom q, we finally obtain $\neg p \vee (\neg p \wedge r) \vee r$ which is equivalent to $\neg p \vee r$.

Example 3.6. This is a classical example from Winslett [30]. We have two atoms, b and m, standing for "a book is on the floor" and "a magazine is on the floor", respectively. $KB = (b \wedge \neg m) \vee (\neg b \wedge m)$. The update formula α is m. Since α is already in the Blake canonical form, the only variable atom is m. We first compute $\exists m.KB$. KB is already in disjunctive normal form. Eliminating all occurrences of m in KB, we get $b \vee \neg b$ which is equivalent to $\top$. Hence, $KB *^m \alpha$ is $m \wedge \top$ which is equivalent to m. ■

3.4. Properties of the MPMA

Katsuno & Mendelzon [17] proposed eight postulates which, as they claimed, should be satisfied by each update operator. In this section, we analyse the MPMA in the context of these postulates.

Let KB be a knowledge base and suppose that α is an update formula. In what follows, we shall write $KB \star \alpha$ as an abbreviation for $KB \star^P \alpha$, where P is the set of all atoms occurring in the Blake canonical form of α.

Katsuno and Mendelzon set up the following postulates:

(1) $KB \star \alpha$ implies α.
(2) If KB implies α, then $KB \star \alpha$ is equivalent to KB.
(3) If both KB and α are satisfiable, then $KB \star \alpha$ is also satisfiable.
(4) If $KB_1 \equiv KB_2$ and $\alpha_1 \equiv \alpha_2$, then $KB_1 \star \alpha_1 \equiv KB_2 \star \alpha_2$.
(5) $(KB \star \alpha_1) \wedge \alpha_2$ implies $KB \star (\alpha_1 \wedge \alpha_2)$.
(6) If $KB \star \alpha_1$ implies α_2 and $KB \star \alpha_2$ implies α_1, then $KB \star \alpha_1 \equiv KB \star \alpha_2$.

(7) If KB is complete, *i.e.* has at most one model, then $(KB \star \alpha_1) \wedge (KB \star \alpha_2)$ implies $KB \star (\alpha_1 \vee \alpha_2)$.
(8) $(KB_1 \vee KB_2) \star \alpha \equiv (KB_1 \star \alpha) \vee (KB_2 \star \alpha)$.

We have already seen (Example 2.1 (continued)) that postulate (2) does not generally hold in MPMA. This is also the case as regards postulates (5), (6) and (7).

To see that (5) does not generally hold, take $ATM = \{m, b\}$, $KB = \neg m \wedge b$, $\alpha_1 = \neg b \vee m$ and $\alpha_2 = b \vee m$. $(KB \star \alpha_1) \wedge \alpha_2 = (\neg b \vee m) \wedge (b \vee m) \equiv m$. On the other hand, $KB \star (\alpha_1 \wedge \alpha_2) \equiv KB \star m \equiv b \wedge m$. Clearly, m does not imply $b \wedge m$. We claim that our result is intuitively plausible. Let us interpret b and m as "a book is on the table" and "a magazine is on the table", respectively. KB states that initially the magazine is not on the table and the book is on it. α_1 corresponds to the action "remove the book from the table or put the magazine on it (or perform both these subactions together). It is obvious that the updated knowledge base should be $\neg b \vee m$. If we take the conjunction of the new knowledge base and α_2, we will obtain m. On the other hand, $\alpha_1 \wedge \alpha_2$ is equivalent to m, which means that the action it corresponds to is "put the magazine on the table". Now, if we apply this action to our original knowledge base, the updated knowledge base should be $m \wedge b$.

It turns out that the following weaker form of postulate (5) holds for MPMA.

Theorem 3.5. If $atm(BCF(\alpha_1)) \subseteq atm(BCF(\alpha_1 \wedge \alpha_2))$, then $(KB \star \alpha_1) \wedge \alpha_2$ implies $KB \star (\alpha_1 \wedge \alpha_2)$. **Proof** Denote $atm(BCF(\alpha_1))$ and $atm(BCF(\alpha_1 \wedge \alpha_2))$ by P and Q, respectively. $(KB \star \alpha_1) \wedge \alpha_2 =$ (by Theorem 3.4) $\alpha_1 \wedge \exists P.KB \wedge \alpha_2$ which implies (because $P \subseteq Q$) $\alpha_1 \wedge \exists Q.KB \wedge \alpha_2$. Again using Theorem 3.4, we conclude that the last formula is just $KB \star (\alpha_1 \wedge \alpha_2)$. ■

Intuitively, Theorem 3.5 states that postulate (5) holds, provided that the action corresponding to $\alpha_1 \wedge \alpha_2$ may vary every atom that can be varied by the action corresponding to α_1.

To see that (6) need not hold, suppose that $ATM = \{b, m\}$, $KB = \neg b \wedge m$, $\alpha_1 = (b \vee m)$ and $\alpha_2 = \top$. $KB \star \alpha_1 \equiv b \vee m$ implies α_2. Similarly, $KB \star \alpha_2 \equiv \neg b \wedge m$ implies α_1. On the other hand, $KB \star \alpha_1$ and $KB \star \alpha_2$ are not equivalent. We leave it to the reader to check that the interpretation of the atoms b and m as "a book is on the table" and "a magazine is on the table", respectively, makes our result plausible.

The following weaker form of postulate (6) holds for the MPMA.

Theorem 3.6. Let $atm(BCF(\alpha_1)) = atm(BCF(\alpha_2))$. If $KB \star \alpha_1$ implies α_2 and $KB \star \alpha_2$ implies α_1, then $KB \star \alpha_1 \equiv KB \star \alpha_2$. **Proof** Denote $atm(BCF(\alpha_1))$ and $atm(BCF(\alpha_2))$ by P. Since $KB \star \alpha_1$ implies α_2 and $KB \star \alpha_2$ implies α_1, we infer (using Theorem 3.4) that the formulas $\alpha_1 \wedge \exists P.KB \rightarrow \alpha_2$ and $\alpha_2 \wedge \exists P.KB \rightarrow \alpha_1$ are tautologies. Hence $\alpha_1 \wedge \exists P.KB \leftrightarrow \alpha_2 \wedge \exists P.KB$ is also a tautology, what (again in view of Theorem 3.4) proves the theorem. ■

Intuitively, Theorem 3.6 states that postulate (6) holds, provided that the actions corresponding to α_1 and α_2 may vary exactly the same atoms.

To see that (7) need not to hold, suppose that $ATM = \{b, m, n\}$, $KB = m \wedge b \wedge n$, $\alpha_1 = (b \wedge m) \vee n$ and $\alpha_2 = (b \wedge \neg m) \vee n$. It is easily verified that $(KB \star \alpha_1) \wedge (KB \star \alpha_2) = \alpha_1 \wedge \alpha_2$ which is equivalent to n. On the other hand, $\alpha_1 \vee \alpha_2$ reduces to $b \vee n$, when the redundant atom m is eliminated. So, $KB \star (\alpha_1 \vee \alpha_2)$ is equivalent to $(b \vee n) \wedge m$ which is not a consequence of n. We leave it to the reader to check that our result is plausible if the atoms b, m and n are interpreted as "a book is on the table", "a magazine is on the table" and "a newspaper is on the table", respectively.

The following version of postulate (7) holds for the MPMA.

Theorem 3.7. If $atm(BCF(\alpha_1) \subseteq atm(BCF(\alpha_1 \vee \alpha_2))$ and $atm(BCF(\alpha_2) \subseteq atm(BCF(\alpha_1 \vee \alpha_2))$, then $(KB \star \alpha_1) \wedge (KB \star \alpha_2)$ implies $KB \star (\alpha_1 \vee \alpha_2)$.[4] **Proof** Denote $atm(BCF(\alpha_1))$, $atm(BCF(\alpha_2))$ and $atm(BCF(\alpha_1 \vee \alpha_2))$ by P, Q and R, respectively. By Theorem 3.4, $(KB \star \alpha_1) \equiv \alpha_1 \wedge \exists P.KB$, $(KB \star \alpha_2) \equiv \alpha_2 \wedge \exists Q.KB$ and $KB \star (\alpha_1 \vee \alpha_2) \equiv (\alpha_1 \vee \alpha_2) \wedge \exists R.KB$. Since $P \subseteq R$ and α_1 implies $\alpha_1 \vee \alpha_2$, $\alpha_1 \wedge \exists P.KB$ implies $(\alpha_1 \vee \alpha_2) \wedge \exists R.KB$ and so, $KB \star \alpha_1$ implies $KB \star (\alpha_1 \vee \alpha_2)$. Similarly, we show that $KB \star \alpha_2$ implies $KB \star (\alpha_1 \vee \alpha_2)$. Hence $(KB \star \alpha_1) \wedge (KB \star \alpha_2)$ implies $KB \star (\alpha_1 \vee \alpha_2)$. ■

Intuitively, Theorem 3.7 says that postulate (7) holds, provided that the set of variable atoms of the action corresponding to α_1 and the set of variable atoms of the action corresponding to α_2 are both subsets of the set of variable atoms of the action corresponding to $\alpha_1 \vee \alpha_2$.

The remaining postulates of Katsuno & Mendelzon hold for the MPMA.

Theorem 3.8. The MPMA satisfies postulates (1), (3)-(4) and (8). **Proof** (1) Follows immediately from Theorem 3.4. (3) See Theorem 3.2. (4) Follows immediately from Theorem 3.4. (8) Let P be the set of all non-redundant atoms occurring in α. By Theorem 3.4, $(KB_1 \vee KB_2) \star \alpha = \alpha \wedge \exists P.(KB_1 \vee KB_2) \equiv \alpha \wedge [(\exists P.KB_1) \vee (\exists P.KB_2)] \equiv (\alpha \wedge \exists P.KB_1) \vee (\alpha \wedge \exists P.KB_2) \equiv (KB_1 \star \alpha) \vee (KB_2 \star \alpha)$. ■

4. A Simple Temporal Logic

Since our primary concern in this section is to compare the knowledge base update and temporal logic approaches to reasoning about action and change, we will only be concerned with action scenarios specified in terms of an initial state description, a formula describing an effect of an action executed and a resulting state description. This will correlate with an initial KB, an update formula added to the KB, and the resulting KB. The above restrictions allow us to work with a very simple pseudo-temporal logic, referred to as STL. Actually, STL is the classical propositional logic which simulates a fragment of the temporal logic in Doherty [6, 16], restricted to two time-points.

[4] The assumption that KB is complete is not necessary here.

4.1. The Language $\mathcal{L}_{stl}$

Given the language $\mathcal{L}_{pma}$, based on a set of atoms ATM, we define the language $\mathcal{L}_{stl}$ as that of classical propositional logic over ATM_{stl}, where

$$ATM_{stl} = \{p^I : p \in ATM\} \cup \{p^R : p \in ATM\} \cup \{O^p : p \in ATM\}.$$

In other words, in the language $\mathcal{L}_{stl}$, each atom $p \in ATM$ is replaced by two copies p^I and p^R. In addition, for each atom $p \in ATM$, we introduce an auxiliary atom O^p. Intuitively, p^I and p^R represent the values of p in the initial state and in the resulting state, respectively. The atom O^p informally states that p is *occluded*, where occlusion plays the role of releasing an atom from keeping its value persistent from the initial state to the final state.

By an *I-formula* (resp. *R-formula*) we mean any formula constructable using only the atoms of the form p^I (resp. p^R). For any formula $\alpha \in \mathcal{L}_{pma}$, we write α^I (resp. α^R) to denote the result of replacing each atom symbol p in α by p^I (resp. p^R).

Let α be an R-formula and suppose that $p_1^R, \ldots, p_n^R$ are all non-redundant atoms occurring in α. We write $Occlude(\alpha)$ as an abbreviation for $O^{p_1} \wedge \cdots \wedge O^{p_n}$.

4.2. Action Scenarios

Action scenarios, or narratives, are often used in the TL paradigm to represent sequences of action occurrences and observations together with a set of timing constraints which provide a partial or total order on the action occurrences. One then proposes an entailment policy, views the narrative as a theory and deduces facts about the narrative. Below, we provide a lightweight version of scenarios appropriate for our two state sequences.

Definition 4.1. (Initial State Description) An *initial state description* is any I-formula. ■

Definition 4.2. (Action Description) An *action description* is the conjunction $\alpha \wedge Occlude(\alpha)$, where α is any R-formula. ■

Definition 4.3. (Action Scenario) An *action scenario* is a conjunction $\alpha \wedge \beta$, where α is an initial state description and β is an action description. ■

4.3. The Minimization Policy

The minimization policy for STL is based on the PMON policy introduced by Sandewall [25] and investigated and extended in Doherty *et al.* [6, 7, 16].

A *nochange axiom*, written NCA, is the conjunction

$$\bigwedge_{p \in ATM} \neg O^p \rightarrow (p^I \leftrightarrow p^R).$$

Intuitively, the nochange axiom states that the value of each non-occluded atom persists from the initial state to the resulting state.

Given the language $\mathcal{L}_{stl}$, we write OCC to denote the set of all atoms of the form O^p.

We are now ready to state the minimization policy. Given an action scenario $\mathcal{A}$ we will use the following filtered preferential entailment policy:

$$NCA \wedge CIRC[\mathcal{A}; OCC],$$

where $CIRC[\Gamma; P]$ represents the standard second-order circumscription axiom which minimizes the predicates P relative to the finite formula Γ. In other words, we minimize the atoms of the form O^p relative to the action description conjoined with the initial state description, and then filter with the nochange axiom NCA. Since the predicates we minimize are actually 0-ary ones, $CIRC[\mathcal{A}; OCC]$ is reducible to propositional logic. Consequently, any efficient decision procedure for classical propositional logic will suffice for computing preferential entailment.

Definition 4.4. (Preferential Entailment) Let $\mathcal{A}$ be an action scenario and suppose that $\alpha \in \mathcal{L}_{stl}$. We say that α is *preferentially entailed* from $\mathcal{A}$ in STL, written $\mathcal{A}|\approx_{stl} \alpha$, iff $NCA \wedge CIRC[\mathcal{A}; OCC] \models \alpha$, where $\models$ denotes the entailment relation of classical propositional logic. ■

5. Reformulating the MPMA in STL

In this section, we provide a reformulation of the MPMA in terms of STL.

Definition 5.1. Let $KB \in \mathcal{L}_{pma}$ and $\alpha \in \mathcal{L}_{pma}$ be a knowledge base and an update formula, respectively. An action scenario $\mathcal{A}$ *corresponding* to KB and α is given by

$$\mathcal{A} = KB^I \wedge \alpha^R \wedge Occlude(\alpha).$$

Example 5.1. Suppose that $KB = p$ and $\alpha = p \vee q$. The action scenario corresponding to KB and α is $p^I \wedge (p^R \vee q^R) \wedge O^p \wedge O^q$. ■

5.1. Proving Equivalence between MPMA and STL

Theorem 5.1. Assume that the language $\mathcal{L}_{pma}$ is based on a set of atoms ATM. Let $KB \in \mathcal{L}_{pma}$ and $\alpha \in \mathcal{L}_{pma}$ be a knowledge base and an update formula, respectively. Suppose that α is in Blake canonical form and let P be the set of all atoms from ATM occurring in α. If $\mathcal{A}$ is an action scenario corresponding to KB and α, then for any formula $\beta \in \mathcal{L}_{pma}$

$$KB \star^P \alpha \models \beta \quad \text{iff} \quad \mathcal{A}|\approx_{stl} \beta^R.$$

Proof By Definition 5.1, $\mathcal{A} = KB^I \wedge \alpha^R \wedge Occlude(\alpha)$ and so, in view of Definition 4.4,

$$\mathcal{A}|\approx_{stl} \beta^R \qquad \text{iff}$$

$$NCA \wedge CIRC[KB^I \wedge \alpha^R \wedge Occlude(\alpha); OCC] \models \beta^R.$$

It is immediately seen that $CIRC[KB^I \wedge \alpha^R \wedge Occlude(\alpha); OCC]$ is equivalent to

$$KB^I \wedge \alpha^R \wedge \bigwedge_{p \in P} O^p \wedge \bigwedge_{q \in ATM-P} \neg O^q$$

and so, it suffices to show that

$$KB \star^P \alpha \models \beta \text{ iff } \gamma \models \beta^R \tag{2}$$

where γ is the the formula

$$NCA \wedge KB^I \wedge \alpha^R \wedge \bigwedge_{p \in P} O^p \wedge \bigwedge_{q \in ATM-P} \neg O^q.$$

The equivalence (2) is an immediate consequence of the following two lemmas.

Lemma 5.1. If w is a model of $KB *^P \alpha$, then there exists a model v of γ such that, for each $p \in ATM$, $p \in w$ iff $p^R \in v$ (and hence $\neg p \in w$ iff $\neg p^R \in v$).

Proof Let w be a model of $KB *^P \alpha$. Thus, in view of Definition 3.3, $w \in |\alpha|$ and there exists an interpretation $u \in |KB|$ such that the distance between w and u wrt P is $\emptyset$. We define an interpretation v of the language $\mathcal{L}_{stl}$ as follows.

(1) For each $p \in ATM$, $p^R \in v$ iff $p \in w$.
(2) For each $p \in ATM$, $p^I \in v$ iff $p \in u$.
(3) For each $p \in P$, $O^p \in v$.
(4) For each $q \in ATM - P$, $\neg O^q \in v$.

It is immediately seen that v is a model of

$$KB^I \wedge \alpha^R \wedge \bigwedge_{p \in P} O^p \wedge \bigwedge_{q \in ATM-P} \neg O^q.$$

Moreover, since the distance between w and u is $\emptyset$, it is clear that v is also a model of NCA. Hence, v is a model of γ, what proves the lemma.

Lemma 5.2. If v is a model of γ, then there exists a model w of $KB *^P \alpha$ such that, for each $p \in ATM$, $p^R \in v$ iff $p \in w$.

Proof Let v be a model of γ. We define two interpretations of the language $\mathcal{L}_{pma}$, w and u, by

(1) For each $p \in ATM$, $p \in w$ iff $p^R \in v$.
(2) For each $p \in ATM$, $p \in u$ iff $p^I \in v$.

Since v is a model of KB^I and α^R, it is obvious that $w \in |\alpha|$ and $u \in |KB|$. In addition, since v is also a model of $NCA \wedge \bigwedge_{p \in P} O^p \wedge \bigwedge_{q \in ATM-P} \neg O^q$, it is clear that the distance between w and u is $\emptyset$. Thus, by Definition 3.3, we conclude that w is a model of $KB *^P \alpha$, what proves the lemma.

This completes the proof of the theorem. ■

5.2. The Decision Procedure for MPMA

Theorem 5.1 provides the basis for a simple decision procedure for computing the MPMA using STL.[5]

Let $\mathcal{L}_{pma}$ and $\mathcal{L}_{stl}$ be the languages used to describe knowledge base update and action scenario queries, respectively. Given an MPMA update query

$$KB \star^P \alpha \models \beta,$$

where KB and α are formulas in $\mathcal{L}_{pma}$ and α is in Blake canonical form, we first construct an action scenario $\mathcal{A}$ corresponding to KB and α, and then we translate the query β in $\mathcal{L}_{pma}$ into its correlate β^R in $\mathcal{L}_{stl}$. We then solve the equivalent problem

$$NCA \wedge CIRC[\mathcal{A}; OCC] \models \beta^R$$

using a decision procedure for classical propositional logic.

Example 5.2. Let $ATM = \{p, q\}$, $KB = p \wedge q$ and $\alpha = \neg q$. We use Theorem 5.1 to show that p is entailed by $KB \star^{\{q\}} \alpha$. An action scenario corresponding to KB and α is $\mathcal{A} = p^I \wedge q^I \wedge \neg q^R \wedge O^q$. It is easily checked that $CIRC[\mathcal{A}, OCC] \equiv p^I \wedge q^I \wedge \neg q^R \wedge O^q \wedge \neg O^p$ and so, $NCA \wedge CIRC[\mathcal{A}, OCC] \equiv p^I \wedge q^I \wedge \neg q^R \wedge O^q \wedge \neg O^p \wedge (p^I \leftrightarrow p^R)$. Since $NCA \wedge CIRC[\mathcal{A}, OCC] \models p^R$, we conclude that $KB \star^{\{q\}} \alpha \models p$. ■

6. Reformulating the MPMA in Dijkstra's semantics

In this section, we provide an equivalent formulation of MPMA in terms of Dijkstra's semantics, originally developed to reason about programs [4, 5].

In [5] we are provided with a very simple programming language whose semantics is specified in terms of formula transformers. More specifically, with each command S there are associated two formula transformers, called the *weakest precondition* and the *strongest postcondition*, denoted by wp and sp, respectively. Before providing the meaning of these transformers, two remarks are in order.[6]

Firstly, we assume here that the programming language under consideration contains one type of variables only, namely Boolean variables. This assumption may seem overly restrictive, but as a matter of fact no other variables will be needed for our purpose. To be in accord with our earlier terminology, Boolean variables will be referred to as *atoms*. Secondly, in Programming Languages terminology, interpretations are usually referred to as *states*. We shall use this latter term here because it seems to be more appropriate in dynamic settings. However, the reader should be aware that states are just interpretations.

[5] For an on-line implementation of a restricted first-order version of STL, called TAL (Temporal Action Logic), see http://www.anton.ida.liu.se/vital/vital.html.

[6] We ignore the *weakest liberal precondition* transformer, considered in [5], because it will not be used in the sequel.

The formula transformers mentioned above are to be understood as follows. For each command S and each formula α:

- $wp(S, \alpha)$ is the formula whose models are precisely all states such that execution of S begun in any one of them is guaranteed to terminate in a state satisfying α.
- $sp(S, \alpha)$ is the formula whose models are precisely all states such that each of them can be reached by starting execution of S in some state satisfying α.

6.1. SDPL: A Simple Dijkstra Programming Language

The considered programming language consists of *skip* command, *abort* command, *assignment* command, *sequential composition* of commands and *alternative* command. Semantics of these commands is specified in terms of formula transformers as follows.

1. **The *skip* command.** This is the "empty" command in that its execution does not change the computation state. The semantics of *skip* is thus given by (here y is a fresh variable)

$$wp(skip, \alpha) = sp(skip, \alpha) = \alpha.$$

2. **The *abort* command.** This command aborts the computation. Its semantics is given by

$$wp(abort, \alpha) = \begin{cases} \bot & \text{if } \alpha \not\equiv \bot \\ \top & \text{otherwise} \end{cases}$$

$$sp(abort, \alpha) = \bot.$$

3. **The *assignment* command.** This command is of the form $x := e$, where x is a (Boolean) variable and e is a (Boolean) expression. The effect of the command is to replace the value of x by the value of e. The weakest precondition of this command is given by

$$wp(x := e, \ \ \alpha) = \alpha[x \leftarrow e].$$

The strongest postcondition for assignment command is a bit more complex. In general, it is given by

$$sp(x := e, \alpha) = \exists y.((x \equiv e[x \leftarrow y]) \wedge \alpha[x \leftarrow y]). \tag{3}$$

If the variable x does not occur in the expression e, the equation (3) can be simplified. In this case

$$sp(x := e, \alpha) = (x \equiv e) \wedge \exists x.\alpha$$

or equivalently

$$sp(x := e, \alpha) = (x \equiv e) \wedge (\alpha[x \leftarrow \top] \vee \alpha[x \leftarrow \bot]). \tag{4}$$

In the sequel we shall only deal with assignment commands, $x := e$, where e is $\top$ or $\bot$. In this case the equation (4) can be replaced by

$$sp(x := e, \alpha) = \begin{cases} x \wedge (\alpha[x \leftarrow \top] \vee \alpha[x \leftarrow \bot]) & \text{if } e \text{ is } \top \\ \neg x \wedge (\alpha[x \leftarrow \top] \vee \alpha[x \leftarrow \bot]) & \text{if } e \text{ is } \bot \end{cases} \tag{5}$$

4. **The *sequential composition* command.** This command is of the form $S_1; S_2$, where S_1 and S_2 are any commands. It is executed by first executing S_1 and then executing S_2. Its semantics is given by

$$\begin{aligned} wp(S_1; S_2, \alpha) &= wp(S_1, wp(S_2, \alpha)). \\ sp(S_1; S_2, \alpha) &= sp(S_2, sp(S_1, \alpha)). \end{aligned}$$

5. **The *alternative* command.** This command is of the form

$$\textbf{if} \quad B_1 \Rightarrow S_1 \quad [\!] \quad B_2 \Rightarrow S_2 \quad [\!] \quad \cdots \quad [\!] \quad B_n \Rightarrow S_n \quad \textbf{fi} \tag{6}$$

where $B_1, \ldots, B_n$ are Boolean expressions and $S_1, \ldots, S_n$ are any commands. $B_1, \ldots, B_n$ are called *guards* and expressions of the form $B_i \Rightarrow S_i$ are called *guarded commands*. In the sequel, we refer to the general command (6) as IF. The command is executed as follows. If none of the guards is true, then the execution aborts. Otherwise, one guarded command $B_i \Rightarrow S_i$ with true B_i is *randomly* selected and S_i is executed.[7] The semantics of IF is given by

$$wp(\text{IF}, \alpha) = \textstyle\bigwedge_{i=1}^{n}(B_i \rightarrow wp(S_i, \alpha)) \wedge \bigvee_{i=1}^{n} B_i$$

$$sp(\text{IF}, \alpha) = \textstyle\bigvee_{i=1}^{n}(sp(S_i, B_i \wedge \alpha)).$$

6.2. A Decision Procedure for MPMA

We now proceed to the reformulation of the MPMA in the context of Dijkstra's semantics.

Suppose we are given a knowledge base KB and an update formula α. Without loss of generality we may assume that α is in Blake canonical form. Suppose that P is the set of all atoms occurring in α. The obvious idea is to construct a command S such that $sp(S, KB) = KB \star^P \alpha$. In other words, we want to replace the update formula by a command which it is the effect of.

Let t be a term. We write $C(t)$ to denote the command *corresponding to* t. This is defined as follows. If t is $\top$, then $C(t)$ is *skip*. If t is $\bot$, then $C(t)$ is *abort*. Finally, let t be of the form $l_1 \wedge \cdots \wedge l_k$. Denote by $[l_i]$ $(1 \leq i \leq k)$ the atom occurring in l_i. Then $C(t)$ is the sequential composition $[l_1] := V_1; \ldots; [l_k] := V_k$, where, for each $1 \leq i \leq k$, V_i is $\top$ if l_i is positive and $\bot$ otherwise. For instance, if t is $p \wedge \neg q$, then $C(t)$ is $p := \top;\ q := \bot$.

Let p be an atom. We write $release(p)$ as an abbreviation for the command **if** $\top \Rightarrow p := \top \ [\!] \ \top \Rightarrow p := \bot$ **fi**.

Let $\alpha = t_1 \vee \cdots \vee t_k$ be a formula in Blake canonical form and suppose that $atm(\alpha) = \{p_1, \ldots, p_n\}$. The command corresponding to α, written $C(\alpha)$, is the sequential composition

$$release(p_1); \ldots; release(p_n); \tag{7}$$

$$\textbf{if} \quad \top \Rightarrow C(t_1) \ [\!] \ \cdots \ [\!] \ \top \Rightarrow C(t_k) \quad \textbf{fi}. \tag{8}$$

[7] Note that when more than one guard is true, the selection of a guarded command is nondeterministic.

Note that if $k = 1$, then (8) is **if** $\top \Rightarrow C(t_1)$ **fi** which is clearly equivalent to $C(t_1)$.[8]

6.3. Proving Correctness of the Decision Procedure

Theorem 6.1. Let KB be a knowledge base and let $\alpha = t_1 \vee \cdots \vee t_k$ be an update formula in the Blake canonical form. Suppose that $P = \{p_1, \ldots, p_n\}$ is the set of all atoms occurring in α. Then

$$KB *^P \alpha \equiv sp(C(\alpha), KB).$$

Proof Note first that for any formula β, $sp(release(p_i), \beta)$ is equivalent to $\beta[p_i \leftarrow \top] \vee \beta[p_i \leftarrow \bot]$ and hence equivalent to $\exists p_i.\beta$. Accordingly, $sp((7), KB)$ is equivalent to $\exists P.KB$. So, $sp(C(\alpha), KB) = sp((8), \exists P.KB) = \bigvee_{i=1}^{k}[sp(C(t_i), \exists P.KB)] =$ (since none of the atoms in P occurs in $\exists P.KB$) $\bigvee_{i=1}^{k}(t_i \wedge \exists P.KB) = (\bigvee_{i=1}^{k} t_i) \wedge \exists P.KB = \alpha \wedge \exists P.KB = KB \star^P \alpha$. This completes the proof of the theorem. ■

Corollary 6.1. Let KB be a knowledge base and let $\alpha = t_1 \vee \cdots \vee t_k$ be an update formula in Blake canonical form. Suppose that P is the set of all atoms occurring in α. Then for each formula β, $KB \star^P \alpha \models \beta$ iff $KB \rightarrow wp(C(\alpha), \beta)$ is a tautology.

Proof Follows immediately from Theorem 6.1 and the definition of wp. ■

Corollary 6.1 is of great practical importance. It shows that we may decide whether a formula is derivable from the updated knowledge base without constructing this knowledge base. All we need is the command representing the update formula.

Example 6.0. (continued) We now use Corollary 6.1 to determine whether p follows from KB updated by α in the MPMA. Since α is already in Blake canonical form, $C(\alpha)$ is

$$release(p); release(q);$$

$$\textbf{if } \top \Rightarrow p := \top \; [\!] \; \top \Rightarrow q := \top \textbf{ fi}.$$

It is easily checked that $wp(C(\alpha), p)$ is equivalent to $\bot$. By Corollary 6.1 we therefore conclude that $KB \star \alpha \not\models p$. ■

7. The MPMA and Integrity Constraints

In this section, we provide a means of extending the MPMA operator for integrity constraints. To deal with integrity constraints in the framework of the classical PMA, it has been proposed in Katsuno & Mendelzon [17] that one define such an extension by $KB \star (A \wedge IC)$, where $\star$ denotes the classical PMA operator and IC is the conjunction of the set of integrity constraints under consideration. Unfortunately, as has been observed by many researchers (see for instance Herzig [14]) this solution is very problematic. To show this, we use a classical example from Herzig [14], originally due to Lifschitz [19].

[8] Two commands S_1 and S_2 are said to be equivalent iff, for each formula α, $sp(S_1, \alpha)$ is equivalent to $sp(S_2, \alpha)$ and $wp(S_1, \alpha)$ is equivalent to $wp(S_2, \alpha)$.

Example 7.1. There are three atoms $sw1$, $sw2$ and l standing for "switch 1 is up", "switch 2 is up" and "light is on", respectively. The integrity constraint $IC = l \leftrightarrow (sw1 \leftrightarrow sw2)$. We would like to update the database $KB = l \wedge sw1 \wedge sw2$ with the update formula $\alpha = \neg sw1$. It is well known that describing the problem with the classical PMA as

$$KB \star (\alpha \wedge IC),$$

will not provide intuitive results. The new knowledge base has two models, w_1 and w_2, given by $\{\neg sw1, \neg l, sw2\}$ and $\{\neg sw1, l, \neg sw2\}$, respectively. The model w_2 is obviously an unintended one.

It is interesting to note that using this solution, but replacing the PMA with the MPMA, gives the same two models. ■

The problem which is well understood by now is that integrity constraints must include fluent dependency information in one form or another. First, we shall employ *causal rules* as described in Gustafsson & Doherty [16]. These are expressions of the form

$$\alpha \gg \beta \tag{9}$$

where α and β are formulas, referred to as an *antedecent* and a *consequent* of the rule, respectively. A rule of the form (9) has the following intuitive interpretation:

(1) The formula $\alpha \rightarrow \beta$ holds in both the initial and the updated knowledge base.
(2) If $\neg\alpha$ held in the initial knowledge base and α holds in the updated knowledge base, then there is a cause for β to hold in the updated knowledge base.[9]

To assure intuitive results we should assume that $IC \equiv \bigwedge_{i=1}^{n}(\alpha_i \rightarrow \beta_i)$.

Recall that in the MPMA we minimize change, excluding from the minimization the atoms occurring in the update formula. In the MPMA with integrity constraints we, in addition, exclude from the minimization the atoms occurring in the consequents of *active* causal rules, where a rule is said to be active iff its antedecent changed its truth-value from *False* to *True* during the update.

Example 7.1. (continued) To encode causal information contained in IC, we introduce two rules: $(sw1 \leftrightarrow sw2) \gg l$ and $\neg(sw1 \leftrightarrow sw2) \gg \neg l$. Suppose we want to check whether $w_1 = \{\neg sw1, \neg l, sw2\}$ should be considered as a model of KB updated by $\neg sw1$ under the integrity constraints IC represented by the above causal rules. Denote by u the only model of $KB \wedge [(sw1 \leftrightarrow sw2) \rightarrow l] \wedge [\neg(sw1 \leftrightarrow sw2) \rightarrow \neg l]$, *i.e.* $u = \{l, sw1, sw2\}$.[10] Note that the antecedent of the rule $\neg(sw1 \leftrightarrow sw2) \gg \neg l$ has changed its value from $False$ to $True$ during the update. Therefore, we are allowed to ignore the atom l while minimizing change. The atom

[9] That is, if $\neg\alpha$ held in the initial knowledge base and α holds in the updated knowledge base, we are allowed to change the values of atoms occurring in β to guarantee that this formula holds in the updated knowledge base.

[10] If KB is a knowledge base and the set of causal rules is $\cup_{i=1}^{n}\{\alpha_i \gg \beta_i\}$, then the original knowledge base should be considered not as KB, but rather as $KB \wedge \bigwedge_{i=1}^{n}(\alpha_i \rightarrow \beta_i)$.

$sw1$ is also ignored since it occurs in the update formula. On the other hand, the atom $sw2$ has the same value in both u and w_1. Accordingly, w_1 is to be viewed as an intended model of the considered update. Consider now the model $w_2 = \{\neg sw1, l, \neg sw2\}$. It is readily checked that we are allowed to ignore $sw1$ and l while minimizing change. However, the models u and w_1 still differ in the atom $sw2$. Acordingly, w_2 should not be viewed as the intended model of the considered update. ■

We now formalize the above idea.

Definition 7.1. A causal rule $\alpha \gg \beta$ is said to be *active* wrt a pair of interpretations $\langle w, v\rangle$ iff the truth-values of α in w and u are $False$ and $True$, respectively. ■

Definition 7.2. Let w and u be two interpretations and suppose that α is a formula in Blake canonical form. Assume further that $CR = \{\alpha_i \gg \beta_i\}$, $i = 1, \ldots, n$ is a set of causal rules such that all β_i's are in Blake canonical form. Let P be a set of atoms given by

$$P = atm(\alpha) \cup \bigcup \{atm(\beta_i) : \alpha_i \gg \beta_i \text{ is active wrt } \langle w, u\rangle\}.$$

The *distance between w and u wrt α and CR* is $DIST(w, u) - P$. ■

Definition 7.3. Let α and CR be as in Definition 7.2. The *update of an interpretation w by a set of interpretations V* wrt α and CR, written $w \star_{\alpha,CR} V$, is the set of those elements from V whose distance wrt α and CR is $\emptyset$. ■

Definition 7.4. Let α and CR be as in Definition 7.2. Let KB be a knowledge base. Let $T(CR)$ denotes the formula $\bigwedge_{i=1}^{n}(\alpha_i \rightarrow \beta_i)$ and let V be the set of all models of $\alpha \wedge T(CR)$. The *update* of KB by α wrt CR, written $KB *_{CR} \alpha$, is given by

$$KB *_{CR} \alpha = \bigcup_{w \in |KB \wedge T(CR)|} w \star_{\alpha,CR} V. \quad ■$$

7.1. STL and Integrity Constraints

To represent the MPMA with integrity constraints in STL, we first slightly generalize the notion of an action scenario.

Let $CR = \{\alpha_i \gg \beta_i : i = 1, \ldots, n\}$ be a set of causal rules. We write $Trans(CR)$ to denote the formula

$$\bigwedge_{i=1}^{n}(\alpha_i^I \rightarrow \beta_i^I) \wedge \bigwedge_{i=1}^{n}(\alpha_i^R \rightarrow \beta_i^R) \wedge \bigwedge_{i=1}^{n}(\neg\alpha_i^I \wedge \alpha_i^R \rightarrow Occlude(\beta_i)).$$

Definition 7.5. An *action scenario under a set of causal rules CR* is the conjunction $Trans(CR) \wedge \alpha \wedge \beta$, where α is an initial state description and β is an action description. ■

Definition 7.6. Let KB, α and CR be a knowledge base, an update formula and a set of causal rules, respectively. An action scenario $\mathcal{A}$ *corresponding* to KB, α and CR is given by

$$\mathcal{A} = KB^I \wedge \alpha^R \wedge Occlude(\alpha) \wedge Trans(CR). \blacksquare$$

Theorem 7.1. Assume that the language $\mathcal{L}_{pma}$ is based on a set of atoms ATM. Let KB, α and CR be a knowledge base, an update formula and a set of causal rules, respectively. Suppose that α and all consequents of causal rules are in Blake canonical form. If $\mathcal{A}$ is an action scenario corresponding to KB, α and CR, then for any formula $\beta \in \mathcal{L}_{pma}$

$$KB \star_{CR} \alpha \models \beta \quad \text{iff} \quad \mathcal{A} | \approx_{stl} \beta^R.$$

Proof The proof immediately follows from the following two lemmas.

Lemma 7.1. If u is a model of $KB \star_{CR} \alpha$, then there exists a model v of $NCA \wedge CIRC[\mathcal{A}, OCC]$ such that, for each $p \in ATM$, $p \in u$ iff $p^R \in v$.

Proof Suppose that u is a model of $KB \star_{CR} \alpha$. Thus, by Definition 7.4, $u \in | \alpha \wedge T(CR) |$ and there exists an interpretation $w \in | KB \wedge T(CR) |$ such that the distance between u and w wrt α and CR is $\emptyset$. Assume that $CR = \{\alpha_i \gg \beta_i : i = 1, \ldots, n\}$ and denote by P the set of atoms given by

$$P = atm(\alpha) \cup \bigcup \{atm(\beta_i) : \alpha_i \gg \beta_i \text{ is active wrt } \langle w, u \rangle\}.$$

We define an interpretation v of the language $\mathcal{L}_{stl}$ as follows.

(1) For each $p \in ATM$, $p^R \in v$ iff $p \in u$.
(2) For each $p \in ATM$, $p^I \in v$ iff $p \in w$.
(3) For each $q \in ATM - P$, $\neg O^q \in v$.
(4) For each $p \in P$, $O^p \in v$.

To prove the lemma, it suffices to show that v is a model of $NCA \wedge CIRC[\mathcal{A}, OCC]$. Recall that $\mathcal{A} = KB^I \wedge \alpha^R \wedge Occlude(\alpha) \wedge Trans(CR)$. Note that literals containing fluents of the form O^p occur only in $Occlude(\alpha)$ and in the part $\bigwedge_{i=1}^n (\neg \alpha_i^I \wedge \alpha_i^R \rightarrow Occlude(\beta_i))$ of $Trans(CR)$. Accordingly, $NCA \wedge CIRC[\mathcal{A}, OCC]$ is equivalent to

$$NCA \wedge KB^I \wedge \alpha^R \wedge \bigwedge_{i=1}^n (\alpha^I \rightarrow \beta^I) \wedge \bigwedge_{i=1}^n (\alpha^R \rightarrow \beta^R) \wedge$$
$$CIRC[Occlude(\alpha) \wedge \bigwedge_{i=1}^n (\neg \alpha_i^I \wedge \alpha_i^R \rightarrow Occlude(\beta_i)), OCC].$$

Since $u \in | \alpha \wedge T(CR) |$ and $w \in | KB \wedge T(CR) |$, it is clear that v is a model of

$$KB^I \wedge \alpha^R \wedge \bigwedge_{i=1}^n (\alpha^I \rightarrow \beta^I) \wedge \bigwedge_{i=1}^n (\alpha^R \rightarrow \beta^R).$$

Also, since the distance between u and w is $\emptyset$, v is a model of NCA. So, it remains to show that v is a model of

$$CIRC[Occlude(\alpha) \wedge \bigwedge_{i=1}^{n}(\neg\alpha_i^I \wedge \alpha_i^R \rightarrow Occlude(\beta_i)), OCC]. \quad (10)$$

Let w_1 and w_2 be any models of (10). We write $w_1 < w_2$ iff the set $\{O^p : O^p \in w_1\}$ is a proper subset of the set $\{O^p : O^p \in w_2\}$ and all literals not containing fluents of the form O^p have the same truth-value in both w_1 and w_2. By the semantics of circumscription, the models of (10) are those models of $Occlude(\alpha) \wedge \bigwedge_{i=1}^{n}(\neg\alpha_i^I \wedge \alpha_i^R \rightarrow Occlude(\beta_i))$ that are minimal wrt the above relation. It is easily seen that v is a model of $Occlude(\alpha) \wedge \bigwedge_{i=1}^{n}(\neg\alpha_i^I \wedge \alpha_i^R \rightarrow Occlude(\beta_i))$. So, we have still to prove that v is minimal wrt $<$. Assume to the contrary that there exists a model v_1 such that $v_1 < v$. The key observation is that the formula $\neg\alpha_i^I \wedge \alpha_i^R$ is true in v for every causal rule $\alpha_i \gg \beta_i$ which is active wrt $\langle w, u\rangle$. Therefore, since this formula contains no fluents of the form O^p, it is also true in v_1. But this means, since v_1 is a model of $Occlude(\alpha) \wedge \bigwedge_{i=1}^{n}(\neg\alpha_i^I \wedge \alpha_i^R \rightarrow Occlude(\beta_i))$, that, for each $p \in P$, $O^p \in v_1$. Consequently, we conclude that $v_1 = v$. A contradiction.

Lemma 7.2. If v is a model of $NCA \wedge CIRC[\mathcal{A}, OCC]$, then there exists a model u of $KB \star_{CR} \alpha$ such that, for each $p \in ATM$, $p \in u$ iff $p^R \in v$.

Proof Let v be a model of $NCA \wedge CIRC[\mathcal{A}, OCC]$. We define two interpretations of the language $\mathcal{L}_{pma}$, w and u, by

(1) For each $p \in ATM$, $p \in w$ iff $p^I \in v$.
(2) For each $p \in ATM$, $p \in u$ iff $p^R \in v$.

Clearly, it suffices to show that u is a model of $KB \star_{CR} \alpha$. Recall that $NCA \wedge CIRC[\mathcal{A}, OCC]$ is equivalent to

$$NCA \wedge KB^I \wedge \alpha^R \wedge \bigwedge_{i=1}^{n}(\alpha^I \rightarrow \beta^I) \wedge \bigwedge_{i=1}^{n}(\alpha^R \rightarrow \beta^R) \wedge$$
$$CIRC[Occlude(\alpha) \wedge \bigwedge_{i=1}^{n}(\neg\alpha_i^I \wedge \alpha_i^R \rightarrow Occlude(\beta_i)), OCC].$$

Hence, it is obvious that $u \in |\, \alpha \wedge T(CR)\,|$ and $w \in |\, KB \wedge T(CR)\,|$. So, in view Definitions 7.2, 7.3 and 7.4, it suffices to show that the distance between w and u wrt α and CR is $\emptyset$. To this end, denote by P the set of all fluents from ATM such that, for each $p \in P$, $O^p \in v$ and consider the part

$$CIRC[Occlude(\alpha) \wedge \bigwedge_{i=1}^{n}(\neg\alpha_i^I \wedge \alpha_i^R \rightarrow Occlude(\beta_i)), OCC]$$

of $NCA \wedge CIRC[\mathcal{A}, OCC]$. It is obvious that P is precisely the set of fluents occurring in the update formula and in the consequents of these causal rules, $\alpha_i \gg \beta_i$, for which the formula $\neg\alpha_i^I \wedge \alpha_i^R$ is true in v. On the other hand, the truth of $\neg\alpha_i^I \wedge \alpha_i^R$ in v implies that the rule $\alpha_i \gg \beta_i$ is active wrt $\langle w, u\rangle$. So, we finally conclude that the distance between w and u wrt α and CR is $\emptyset$. This proves the lemma and hence completes the proof of the theorem. ■

Practically, the theorem shows that we have provided an alternative syntactic characterization and implementation of MPMA with integrity constraints in terms of STL. Since STL is a very restricted version of a richer temporal logic TAL [6, 16], it would be straightforward to provide additional generalizations to MPMA, based on intuitions from TAL.

7.2. SDPL and Integrity Constraints

In this section we provide the reformulation of the MPMA under integrity constraints IC following the same semantic idea as in STL but now we represent the fluent dependency information by *influence relation.*

Let KB be a knowledge base satifying integrity constraints formula IC, α be an update formula and $P = atm(\alpha)$. The task is to construct an additional modification of $KB' = KB *^P \alpha$ such that the resulting KB'' satisfies IC. The idea used below is based on [23]. We start by introducing some terminology.

By *clause* we mean a formula of the form $l_1 \vee \ldots \vee l_n,\ n \geq 1$, where l_i is a literal.

We say that a clause α *absorbs* a clause β if α is a subformula of β. For instance, the clause a absorbs the clause $a \vee l$. Let α be a formula in conjunctive normal form *CNF*. We write $ABS(\alpha)$ to denote the formula obtained from α by deleting all absorbed clauses. Clearly, α and $ABS(\alpha)$ are equivalent.

Two clauses are said to have an *opposition* if one of them contains a literal l and the other the literal $\neg l$.

Suppose that two clauses, α and β, have exactly one opposition. Then the *resolvent* of α and β, written $res(\alpha, \beta)$, is the clause obtained from the disjunction $\alpha \vee \beta$ by deleting the opposed literals as well as any repeated literals. For example, $res(\neg a \vee l, a \vee d)$ is $l \vee d$. Let α be a formula. The *canonical form of* α, written $CF(\alpha)$, is the formula obtained from α by the following construction.

1. Let β be the conjunctive normal form of α.
2. Repeat as long as possible:
   if β contains a pair δ and γ of clauses whose resolvent exists and no clause of β is a subformula of $res(\delta, \gamma)$, then $\beta := \beta \wedge res(\delta, \gamma)$.
3. Take $ABS(\beta)$. This is $CF(\alpha)$.

Theorem 7.2. Let α be a formula. $CF(\alpha)$ is a conjunction of all minimal clauses implied by α[11]. ■

Now, we use additional information on atoms, represented by an *influence relation*,which is a transitive relation $\mathcal{I} = \{(f_i, g_i) | f_i, g_i \in ATM \text{ for } i = 1, \ldots, n\}$. Its meaning is that any change of the atom g_i (called a *successor*) may influence the atom f_i (called a *predecessor*).

Let $\mathcal{I}$ be an influence relation, α be an update formula and β be a formula. The *canonical form of* β *wrt* α *and* $\mathcal{I}$, written $CFI(\beta, \alpha, \mathcal{I})$ is the formula obtained from β by the following construction.

[11] A clause C is a minimal clause implied by α iff no proper subclause of C is implied by α.

1. Let γ be $CF(\beta)$ where $\gamma \equiv \gamma_1 \wedge \ldots \wedge \gamma_n$.
2. For all i, $1 \leq i \leq n$, transform γ_i into its equivalent implication $\beta_i \rightarrow \alpha_i$, where α_i contains all atoms from γ_i which are predecessors wrt $atm(\alpha)$ in relation $\mathcal{I}$, and β_i the remaining atoms[12].
3. $CFI(\beta, \alpha, \mathcal{I}) = (\beta_1 \rightarrow \alpha_1) \wedge \ldots \wedge (\beta_n \rightarrow \alpha_n)$[13].

7.2.1. Solving Integrity Constraints in SDPL

Now we proceed to construct Dijkstra's command realizing integrity constraints.

Let KB be a knowledge base, α an update formula and $CFI(IC, \alpha, \mathcal{I}) = C_1 \wedge \ldots \wedge C_m$. Note that if a conjunct C_i contains no atoms from $atm(\alpha)$, it holds after updating KB by α. And if $C_i = (\beta_i \rightarrow F)$, then it cannot be realized, because there are no atoms, whose values may be modified[14]. Taking into account the above observations we should consider only those conjuncts of $CFI(IC, \alpha, \mathcal{I})$, which contain atoms from $atm(\alpha)$ and can be realized. We define

$$IC(\alpha) = \{C_i \mid C_i = \beta_i \rightarrow \alpha_i \text{ and } atm(\alpha) \cap atm(\beta_i) \neq \emptyset\}.$$
$$IC_Q(\alpha) = \{C_i \mid C_i = \beta_i \rightarrow F\}.$$
$$IC_{ram}(\alpha) = IC(\alpha) - IC_Q(\alpha).$$

Thus we should consider the modification of $KB' = KB *^P \alpha$ wrt $IC_{ram}(\alpha)$. Assume that $IC_{ram}(\alpha) = \{\beta_i \rightarrow \alpha_i, 1 \leq i \leq n\}$.

Now we formulate Dijkstra's command $C^{MPMA}_{IC}(\alpha)$ realizing update α of KB wrt IC. This is the composition of $C^{MPMA}(\alpha); C_{ram}(\alpha)$, where $C^{MPMA}(\alpha)$ is a command realizing update and C_{ram} is a command realizing integrity constraints given below (we use auxilliary variables $AUX = \{x_1, x_2, \ldots\}$ to denote conjuncts of $IC_{ram}(\alpha)$ which do not hold after the update of KB).

$x_1 := \bot; \ldots; x_n := \bot;$
if $\beta_1 \wedge \neg\alpha_1 \rangle x_1 := \top [\!] \neg\beta_1 \vee \alpha_1 \rangle skip$ **fi**; $\{x_i = \top$ denotes that$\}$
$\vdots$ {i-th element of IC_{ram}}
if $\beta_n \wedge \neg\alpha_n \rangle x_n := \top [\!] \neg\beta_n \vee \alpha_n \rangle skip$ **fi**; { should be realized}
if $x_1 \rangle release(atm(\alpha_1)) [\!] \neg x_1 \rangle skip$ **fi** ;
$\vdots$
if $x_n \rangle release(atm(\alpha_n)) [\!] \neg x_n \rangle skip$ **fi**

$KB' = \exists AUX.(sp(C^{MPMA}(\alpha); C_{ram}(\alpha), KB \wedge IC) \wedge IC).$

[12] If a conjunct does not contain predecessors we put $\beta_i \rightarrow F$ and if a conjunct contains only predecessors we put $T \rightarrow \alpha_i$.

[13] Note that β_i is a clause and α_i is a term.

[14] In reasoning about action paradigm this case is usually referred to as qualification (see [23] for details).

7.2.2. Examples

Example 7.2. Let $KB = \neg a \wedge \neg b \wedge c$, an update formula $\alpha = a$, $IC = (a \rightarrow b) \wedge (b \rightarrow c)$ and $\mathcal{I} = \{(c, b), (b, a), (c, a)\}$.

Hence, $CFI(IC, \alpha, \mathcal{I}) = (a \rightarrow b) \wedge (a \rightarrow c) \wedge (\top \rightarrow (\neg b \vee c))$, $IC(\alpha) = \{(a \rightarrow b), (a \rightarrow c)\}$, $IC_Q(\alpha) = \emptyset$ and $IC_{ram}(\alpha) = IC(\alpha)$.
$IC_{ram}(\alpha)$ has two elements, thus we need x_1, x_2. $C_{IC}^{MPMA}(\alpha)$ is

$$
\begin{array}{l}
\{C^{MPMA}(\alpha)\} \quad a := \top; \\
\{C_{ram}(\alpha)\} \qquad x_1 := \bot; \;\; x_2 := \bot; \\
\textbf{if } a \wedge \neg b \rangle x_1 := \top [\![\neg a \vee b \rangle skip \textbf{ fi}; \textbf{ if } a \wedge \neg c \rangle x_2 := \top [\![\neg a \vee c \rangle skip \textbf{ fi}; \\
\textbf{if } x_1 \rangle release(b) [\![\neg x_1 \rangle skip \textbf{ fi}; \textbf{ if } x_2 \rangle release(c) [\![\neg x_2 \rangle skip \textbf{ fi}
\end{array}
$$

$KB' = \exists x_1 \exists x_2 . (sp(a := \top; C_{ram}(\alpha), KB \wedge IC) \wedge IC) \equiv a \wedge b \wedge c$. ■

Example 7.3. We have three atoms: $sw1$, $sw2$, $light$, standing for positions of switch1, switch2 and the state of the light bulb, respectively. The update formula $\alpha = \neg sw1$, and $KB \equiv light \wedge sw1 \wedge sw2$. $IC = (light \leftrightarrow (sw1 \leftrightarrow sw2))$. $\mathcal{I} = \{(light, sw1), (light, sw2)\}$.

$$
\begin{array}{rcl}
CF(IC) & \equiv & (\neg sw1 \vee sw2 \vee \neg light) \wedge (\neg sw2 \vee sw1 \vee \neg light) \\
& \wedge & (sw1 \vee sw2 \vee light) \wedge (\neg sw1 \vee \neg sw2 \vee light) \\
CFI(IC, \alpha, \mathcal{I}) & \equiv & (sw1 \wedge \neg sw2 \rightarrow \neg light) \wedge (sw2 \wedge \neg sw1 \rightarrow \neg light) \\
& \wedge & (\neg sw1 \wedge \neg sw2 \rightarrow light) \wedge (sw1 \wedge sw2 \rightarrow light)
\end{array}
$$

We need here only one auxilliary variable x_1, because the right hand side of implication of all conjuncts consists of the same atom $light$. $C_{IC}^{MPMA}(\alpha)$ is

$$
\begin{array}{l}
\{C^{MPMA}(\alpha)\} \quad sw1 := \bot; \\
\{C_{ram}(\alpha)\} \qquad x_1 := \bot; \\
\textbf{if} sw1 \wedge \neg sw2 \wedge light \rangle x_1 := \top [\![\neg sw1 \vee sw2 \vee \neg light \rangle skip \textbf{ fi}; \\
\textbf{if } sw2 \wedge \neg sw1 \wedge light \rangle x_1 := \top [\![\neg sw2 \vee sw1 \vee \neg light \rangle skip \textbf{ fi}; \\
\textbf{if } sw1 \wedge \neg sw2 \wedge \neg light \rangle x_1 := \top [\![\neg sw1 \vee sw2 \vee light \rangle skip \textbf{ fi}; \\
\textbf{if } sw2 \wedge \neg sw1 \wedge \neg light \rangle x_1 := \top [\![\neg sw1 \vee sw1 \vee light \rangle skip \textbf{ fi}; \\
\textbf{if } x_1 \rangle release(light) [\![\neg x_1 \rangle skip \textbf{ fi}
\end{array}
$$

$KB' = \exists x_1 . (sp(C_{IC}^{MPMA}(\alpha), KB \wedge IC) \wedge IC) \equiv \neg sw1 \wedge sw2 \wedge \neg light$. ■

8. Comparisons with Existing Work

In this section, we compare our approach with that of Herzig [14], whose work is most closely related to ours, del Val and Shoham [29] Sandewall [26], and Thielscher [28]. Before doing that, we show and discuss some narratives from the Gustafsson and Doherty [6, 16] defined in a temporal logic which subsumes STL.

8.1. Doherty and Gustafsson

We stated earlier that STL is a propositional fragment of PMON, introduced by Sandewall [25]. A syntactic characterization for PMON in terms of circumscription was proposed by Doherty [6] where it was shown that any action scenario in a particular class of theories was always reducible to a first-order theory. STL uses the same minimization policy and because it is propostional, the reduction results apply. PMON was extended to PMON(RC) [16] for ramification using causal rules which in effect are simply abbreviations for a fluent categorization technique. STL with causal rules is a propositional fragment of PMON(RC) with the same minimization policy and reduction results.

In this section, we will consider two examples from [16], the latter slightly modified from the original. These examples should help in acquiring both a conceptual and technical understanding of PMON(RC) and show how easily STL could be generalized to iterated updates.

Example 8.1. The walking turkey problem is a well-known ramification scenario and relatively straightforward to encode and solve using causal constraints. We will require one causal constraint stating that dead turkeys do not walk. One ramification of shooting a turkey is that it no longer walks and our theory should entail this indirect effect. The following action scenario description in $\mathcal{L}(SD)$ describes the walking turkey problem. (The predicate symbol "H", used in the sequel, is an abbreviation for "Holds".)

$$\begin{array}{ll} obs1 & [0]walking \\ ac1 & [2,4]Shoot \\ acs1 & [s,t]Shoot \rightsquigarrow [s,t]alive := F \\ csc1 & \forall t[t]\neg alive \gg \neg walking \end{array}$$

The corresponding translation into $\mathcal{L}(FL)$ is

$$\begin{array}{ll} obs1 & H(0, walking) \\ scd1 & \neg H(4, alive) \wedge \forall t(2 < t \leq 4 \rightarrow Occlude(t, alive)) \\ csc1 & \forall t[(\neg H(t, alive) \rightarrow \neg H(t, walking)) \wedge \\ & \quad (H(t-1, alive) \wedge \neg H(t, alive) \rightarrow Occlude(t, walking))] \end{array}$$

The scenario is partitioned as:

$\Gamma_{FIL} = \Gamma_{NCG} \cup \Gamma_{OBS} \cup \Gamma_{ACC} \cup \Gamma_{UNA} \cup \Gamma_T$, where

$$\Gamma_{OBS} = \{obs1\}, \Gamma_{UNA} = \{alive \neq walking\}, \Gamma_{ACC} = \{\}.$$

$\Gamma_{CHG} = \Gamma_{SCD} \wedge \Gamma_{CSC}$, where

$$\Gamma_{SCD} = \{scd1\}, \Gamma_{CSC} = \{csc1\}.$$

Circumscribing *Occlude* in Γ_{CHG} results in the following definition which can be generated using either our algorithm [8], or standard predicate completion:

$$\begin{aligned}\forall t, f(Occlude(t,f) \leftrightarrow \quad & ((f = alive \wedge 2 < t \leq 4) \vee \\ & (f = walking \wedge \\ & \quad H(t-1, alive) \wedge \\ & \quad \neg H(t, alive)))).\end{aligned}$$

For readability, we list the complete translation of the scenario in $\mathcal{L}(FL)$, with the definition of *Occlude* derived via circumscription:

$$\begin{aligned}&\forall f, t(H(t,f) \oplus H(t+1,f) \rightarrow Occlude(t+1,f)) \wedge \\ &H(0, walking) \wedge alive \neq walking \wedge \neg H(4, alive) \wedge \\ &\forall t[(\neg H(t, alive) \rightarrow \neg H(t, walking)) \wedge \\ &\forall t, f(Occlude(t,f) \leftrightarrow ((f = alive \wedge 2 < t \leq 4) \vee \\ &\qquad (f = walking \wedge H(t-1, alive) \wedge \neg H(t, alive)))).\end{aligned}$$

There are two classes of preferred models for this scenario due to the fact that the shoot action has duration and its effects may occur at time point 3 or 4:

$$\begin{aligned}&[0,2] alive \wedge walking \\ &[3,\infty] \neg alive \wedge \neg walking\end{aligned}$$

and

$$\begin{aligned}&[0,3] alive \wedge walking \\ &[4,\infty] \neg alive \wedge \neg walking \ \blacksquare\end{aligned}$$

Example 8.2. The extended stuffy room problem is based on the original problem due to Winslett [30]. We extend it by introducing a box which requires the use of chained causal rules, and by encoding a delayed effect of an action by using one of the causal rules.

In this scenario, there are two causal constraints. The first asserts that *duct*2 is blocked if something is put on top of it (*loc*2). The second asserts that the room gets stuffy two time-points after both ducts are blocked. Between time-points 2 and 5 a box is moved to location *loc*2. The action scenario is represented in $\mathcal{L}(SD)$ as:

$$\begin{aligned}obs \quad & [0](blocked(duct1) \wedge \neg blocked(duct2) \wedge \\ & \qquad\qquad at(box, loc1) \wedge \neg stuffy) \\ ac \quad & [2,5] move(box, loc1, loc2) \\ acs \quad & \forall x, l1, l2([s,t] move(x, l1, l2) \rightsquigarrow\end{aligned}$$

$$[s,t]at(x,l1) := F \wedge [s,t]at(x,l2) := T)$$

$csc1$ $\quad \forall t, x[t](at(x,loc2) \gg blocked(duct2))$

$csc2$ $\quad \forall t([t](blocked(duct1) \wedge blocked(duct2)) \gg [t+2]stuffy)$

Translating into $\mathcal{L}(FL)$ and circumscribing, results in the following theory:

$$H(0,blocked(duct1)) \wedge \neg H(0,blocked(duct2)) \wedge H(0,at(box,loc1)) \wedge$$
$$\neg H(0,stuffy) \wedge \neg H(5,at(box,loc1)) \wedge H(5,at(box,loc2)) \wedge$$
$$\forall t,x(H(t,at(x,loc2)) \rightarrow H(t,blocked(duct2))) \wedge$$
$$\forall t(H(t,blocked(duct1)) \wedge H(t,blocked(duct2)) \rightarrow H(t+2,stuffy)) \wedge$$
$$\forall f,t(H(t,f) \oplus H(t+1,f) \rightarrow Occlude(t+1,f)) \wedge$$
$$\forall t,f(Occlude(t,f) \leftrightarrow$$
$$(2 < t \leq 5 \wedge f = at(box,loc1)) \vee (2 < t \leq 5 \wedge f = at(box,loc2)) \vee$$
$$\forall x(\neg H(t-1,at(x,loc2)) \wedge H(t,at(x,loc2)) \wedge f = blocked(duct2)) \vee$$
$$(\neg[H(t-3,blocked(duct1)) \wedge H(t-3,blocked(duct2))] \wedge$$
$$H(t-2,blocked(duct1)) \wedge H(t-2,blocked(duct2)) \wedge f = stuffy)).$$

In addition, the unique name axioms are included, but not listed here. The theory correctly entails that if something is placed on top of $duct2$ while $duct1$ is blocked, then the room will become stuffy two time-points later. In the current axiomatization, the room would remain stuffy forever even after one of the ducts became free. This is because the nochange axiom prevents the room from becoming unstuffy without reason. In the current theory, there is no axiom which states otherwise. A rule stating that the room is stuffy only when both ducts are blocked can be added without difficulty:

$csc3$ $\quad \forall t([t]\neg(blocked(duct1) \wedge blocked(duct2)) \gg$
$[t+2]\neg stuffy).$

This example demonstrates both the use of casual chaining and how causal constraints can be used to specify delayed effects of actions. ■

8.2. Herzig

Herzig [14] provides a sound and complete decision procedure for the PMA by constructing an equivalent presentation of the PMA in terms of conditional logic.[15] A syntactic characterization of an update $w \star U$ of an interpretation w by a set of interpretations U is provided in terms of a conditional operator $>$ where $A > C$ is read as a hypothetical update: "if the current database is updated with A then C follows". Herzig proves the following proposition:

[15] A relationship between the PMA and conditionals has also been studied in Fariñas del Cerro *et al.* [10, 11, 12] and Grahne [15].

Proposition 8.1. (Herzig [14]) *Let KB, A and C be classical. Then $KB \star A \models C$ iff $KB \models A > C$.* ■

Herzig then shows how the conditional $A > C$ can be rewritten to a classical formula using normal forming which together with the proposition above provides the decision procedure.

In Section 8 of his paper, Herzig sketches how one might deal with integrity constraints by introducing a new update operator which after careful analysis can be shown to be quite similar in spirit to the MPMA operator presented in this paper. Although similar in concept, Herzig's decision procedure is still based on translating update queries into conditionals and then translating via normal forming to a classical formula. In addition, instead of computing one PMA update, he is forced to compute 2^n classical PMA updates per query, where n is the number of atoms dependent on any of the atoms in the update formula.

To show the relation between the MPMA update operator $\star^P$ and Herzig's new update operator, written $\star_{IC,DEP}$, we will use Example 7.1 which is also used in Herzig [14], but with the atoms renamed.

In addition to the integrity constraint IC, one must also include fluent dependency information in one form or another. Herzig does this in the following manner: He first specifies a dependency function, DEP, from atoms to sets of atoms such that $p \in DEP(p)$, for each atom p. In addition, $DEP(p)$ may contain atoms other than p. Intuitively, $q \in DEP(p)$ means that updates concerning p may change the truth-value of q. For a formula A, $DEP(A)$ stands for $\bigcup_{[p \in atm(A)]} DEP(p)$.

In the example, $DEP(l) = \{l, sw1, sw2\}$, $DEP(sw1) = \{sw1, l\}$, and $DEP(sw2) = \{sw2, l\}$. The dependence function DEP encodes the change dependency inherent in a causal reading of IC and represents a fluent dependency graph.

An update operator $\star_{IC,DEP}$ under a set of integrity constraints IC and a dependence function DEP is then defined by

$$KB \star_{IC,DEP} A = \left(\bigvee_{B \in CTX(A)} (KB \star B) \right) \wedge IC \wedge A$$

where $\star$ is the classical PMA operator and $CTX(A)$ is $\{l_1 \wedge \cdots \wedge l_n : l_i = p_i \text{ or } l_i = \neg p_i\}$ if $DEP(A) = \{p_1, \ldots, p_n\}$. So, for example, if $A = \neg sw1$ and the function DEP is as specified above where $DEP(sw1) = \{sw1, l\}$, then

$$CTX(sw1) = \{sw1 \wedge l, sw1 \wedge \neg l, \neg sw1 \wedge l, \neg sw1 \wedge \neg l\}.$$

This means that if the cardinality of $DEP(\alpha)$ is n then the cardinality of $CTX(\alpha)$ is 2^n.

The role of $CTX(A)$ is central to the approach and essentially does the job that the set P of atoms in the MPMA operator $\star^P$ does together with the eliminant $\exists P.KB$, or that the OCC atoms together with the NCA axiom plays in the translation to STL. Taking the disjunction of PMA updates of KB with each of the formulas in $CTX(sw1)$, has the effect of generating

the same models as one would with the eliminant of KB where $P = \{sw1, l\}$. It can easily be shown that

$$KB \star_{IC,DEP} \neg sw1 \equiv KB \star_{CR} \neg sw1,$$

where,

$$KB = l \wedge sw1 \wedge sw2,$$

$$IC = (sw1 \leftrightarrow sw2) \leftrightarrow l,$$

and

$$CR = \{[(sw1 \leftrightarrow sw2) \gg l], \\ [\neg(sw1 \leftrightarrow sw2) \gg \neg l]\},$$

or the translation into STL where,

$$\mathcal{A} = KB^I \wedge \neg sw1^R \wedge O^{sw1} \wedge Trans(CR).$$

The advantage of using the latter is that the decision procedure is much more efficient and that the dependency information included in DEP is implicit in the expansion of the abbreviation $\gg$.

Herzig's new update operator can be also used for the empty set of integrity constants by putting $IC = \top$. It turns out, that the operator $\star_{\top,DEP}$ is just the MPMA operator, with the set of variable atoms identified with the set of all the atoms occurring in the update formula.[16] More precisely:

Theorem 8.1. For each KB and A

$$KB \star_{\top,DEP} A \equiv KB \star^P (A)$$

where $P = atm(A)$. **Proof** The proof immediately follows from the fact that $\bigvee_{B \in CTX(A)}(KB \star B)$ is equivalent to $\exists P.KB$. ■

8.3. del Val and Shoham

The approach we use to compare belief update and temporal logic approaches to reasoning about action and change has some similarity to the work of del Val and Shoham [29], but the focus and goals are quite different. del Val and Shoham:

- begin with an independently motivated theory of action based on the situation calculus (VSL).
- provide an encoding of the update problem in this theory of action.

[16] This means that Herzig's new update operator does not distinguish between redundant and non-redundant atoms occurring in the update formula. In consequence, it is syntax dependent and may lead to counterintuitive results (see Example 3.3).

- show that the encoding allows them to analytically derive the Katsuno-Mendelzon (KM) postulates, where all the postulates hold for VSL.
- begin with an independently motivated generalization of VSL based on allowing the facts that persist by default(those in the *Frame* [19]) to depend on the current state of the world.
- By doing this, they identify a class of action theories which stands in one-to-one correspondence with the family of KM operators.
- They then show that the correspondence is achieved by imposing certain arbitrary restrictions on the action theories where each restriction corresponds to exactly one KM postulate. The restrictions, SDP, PDS, and TI, are based on the way the set of persistent fluents is defined at each state.
- They then suggest that the KM proposal can be viewed as embodying these same restrictive assumptions and by relaxing these assumptions, one can define reasonable update operators outside the KM family.

On the other hand, we

- begin with the PMA which satisfies the KM postulates and modify it for disjunctive update, basing the modification on intuitions derived from work in the nonmonotonic temporal logic area and in particular with the PMON (TAL) family of logics.
- We analyze the derived update operator relative to the KM postulates and notice that a number of the postulates do not apply, or apply in a weaker form.
- We also notice that the update policy proposed is sensitive to the syntactic form of the update formula in a subtle manner associated with the use of redundant atoms. This is rectified and the resulting update operator MPMA with restrctions is no longer syntactically sensitive.
- We then introduce a number of syntactic characterizations for the new operator, the most efficient, in terms of encoding the MPMA into a fragment of an independently motivated action theory, PMON (TAL), which is a state-based temporal logic.

As a result of the similarity between the analyses, a number of interesting questions arise:

- What is the relation between the two logics STL and VSL used by del Val and Shoham?
- How can one make sense out of the KM postulates that do not hold for MPMA in terms of the restrictions SDP, PDS, and IT, claimed to explain some of the less obvious KM postulates?

Complete answers to these questions would involve a paper in itself, but some informal comparisons can be made. Let's begin with the relation between STL and VSL.

If one peels away all the complications associated with the intricacies of the situation calculus and the situation/state relation,VSL works as follows: Given two states s_I and s_R, where s_R is one of the states resulting from executing an action A, fix $Holds$ in s_I, vary it in S_R, and minimize change from s_I to s_R for all fluents asserted to persist in s_I ($Persistent(f, s_I)$). It is assumed that there is a persistence axiom encoding the policy of what fluents persist where.

STL works as follows: Given two states s_I and s_R, where s_R is one of the states resulting from executing an action A, fix $Holds$ in s_I and s_R, minimize $Occlude$ in s_R, and remove any state s_R where a fluent f which is not occluded changes value from s_I to s_R. All fluents in the postcondition of action A are occluded, conditioned on the precondition of the action. The generalization to causal rules is similar. For the case with integrity constraints, the policy is essentially the same.

It is clear that $Persistent$ plays a role similar to $\neg Occlude$. By minimizing $Occlude$, we know what *must* persist. By minimizing $Persistent$, we know what is not occluded by default. Since STL uses filtering to remove unoccluded change, it's persistence policy will not be as pliant as that of VSL. In addition, the persistence policy is arbitrarily set up in VSL relative to a persistence axiom which is stated to be a parameter to the action theory. In STL, it is implicitly defined in a principled manner via the definition of action and causal rules. There are certainly strong similarities between the two logics, but formal representation results would surely be dependent on what constraints are placed on the use of persistence in VSL and on the differences in the linear time, state-based ontology of STL and the branching time, event-based ontology of VSL. When STL is extended with causal rules, the minimization policy is essentially the same, but the triggering conditions for when a rules applies are dependent on both s_I and s_R.

One of the main points being made by del Val and Shoham [29] is the relation between the KM postulates and the update operator implicit in VSL, where the constraints placed on the persistence policy in VSL determine which of the KM postulates hold or not. Relaxing each of three constraints, TI, PDS, and SPD, individually or in combination, generate different update operators which can still be claimed to be well-behaved or at least independently motivated from the temporal reasoning point of view. We have proposed an update operator, MPMA, encoded it in a fragment STL of an independently motivated temporal logic TAL and analyzed the KM postulates which hold. It would be interesting to compare this analysis with that of del Val and Shoham to see if one can acquire any insight as to why certain KM postulates hold for MPMA and why some do not. If del Val and Shoham's claim holds concerning the role the persistance policy plays in this regard and if the idea is transferable to action theories other than that of VSL, then there should be some correlations. We begin the comparison by setting up the following table[17]:

[17] The symbol No* stands for "No, but a weaker version exists."

KM Postulate	VSL	MPMA/STL
U1	Yes	Yes
U2	PDS	No
U3	Yes	Yes
U4	TI	Yes
U5	Yes	No^*
U6	Yes	No^*
U7	SDP	No^*
U8	Yes	Yes

The three persistence constraints discussed in [29] are:

PDS – Persistent fluents determine state.
TI – Persistent fluents are identical for identical states at different times.
SDP – State determines persistent fluents.

As regards (U2), it is clearly the case that it shouldn't hold because the main point in defining MPMA is to deal with disjunctive update. As regards (U4), recall that if the *redundant atom anomaly* had not been dealt with in MPMA, then (U4) would not hold. As regards (U5)-(U7), only weaker versions of these postulates hold. What the theorems tell us is essentially that the occlusion policy is "upwards compositional" in the sense that literals occluded due to an update formula will still be occluded if the update formula is used as a subformula in another update formula. What relation this may have to SDP and TI is unclear to us. On the other hand, the application dependent counterexamples we use to show that (U5)-(U7) do not hold for MPMA, are prefectly reasonable and not artificial constructions.

It would be interesting to analyze the full PMON [6] and PMON(RC) [16] in the context of the above discussion. In this case, there could be even stronger correlation with VSL in the sense that narratives are not restricted to 2 states and one can take arbitrary temporal evolution of a database into account. For example, following the argument in [29], postulate (U4) would not hold, because of the non-markovian behavior of causal rules, whose antecedent can often refer to more than the current state of action execution. Because the postcondition contains *Occlusion* ($\neg Persistence$) assertions, the set of persistent fluents for a single state might vary at different timepoints.

8.4. Sandewall

Sandewall [26] analyzes and compares methods for ramification, or for update, which are based on minimization of change while respecting domain constraints. He provides comparative assessments of a number of previously proposed entailment methods based on the partitioning of fluents into categorizations which relax minimal change policies. In particular, he introduces *measure systems* which correspond to structures representing different types of update policies in the spirit of PMA and it's generalizations. When doing comparative assessments, he introduces the method MSC which he states is the method proposed by del Val and Shoham and which we

discussed in the previous section. Of all the methods discussed, MSC is the closest one would get to the MPMA approach. Sandewall's claim of a correspondence between MSC and VSL only holds under the assumption that VSL has TL and SDP and lacks PDS. This is due to the fact that VSL is *faithfull* if TL, SDP, and PDS hold, while MSC is not. Based on these relations, it is safe to say that neither MPMA, STL with integrity constraints, PMON, or PMON(RC) are equivalent with any of the methods considered by Sandewall.

8.5. Thielscher

We stated earlier, that the *redundant atom anomaly* was not only a problem for MPMA type update operators, but also for other formalisms in the temporal logic paradigm. As an example, we will show that the approach proposed by Thielscher [27, 28] for reasoning about actions which deals with ramification in terms of casual rules, would suffer from the redundant atom problem without some additional preprocessing of domain constraints. We assume knowledge of [28] in the following.

Suppose that the domain constraints D are given by

$$D = \{(p \vee q) \wedge (p \vee \neg q)\},$$

and influence information by

$$I = \{(q, p)\}.$$

According to definition 9 in [28], D and I lead to the following causal relationship R, where

$$R = \{\neg q \text{ causes } p \text{ if } \top\} \cup \{q \text{ causes } \neg p \text{ if } \top\}.$$

But D is equivalent to p. If we take $D = \{p\}$ with I as before then,

$$R = \emptyset.$$

9. Discussion

We have demonstrated the benefits of applying intuitions derived from research on action and change in the temporal reasoning community to the BR/U paradigm. The result is a generalization of the PMA to MPMA which handles disjunctive updates and integrity constraints interpreted as causal constraints. We have provided a number of syntactic characterizations of the MPMA and shown how an MPMA query can be mapped into a query of a temporal narrative represented in STL, a simple temporal logic for reasoning about action and change. Since STL is the propositional fragment of a highly expressive 1st-order temporal logic, it should be straightforward to generalize the results described here to the 1st-order case, where it has already been shown that the circumscription formula associated with the minimization policy in the full first-order version of STL is reducible to a first-order formula. An additional generalization would include the modeling of iterated belief update in MPMA in terms of narratives with

more than two timepoints. Because the minimization policy associated with STL is similar to many other current approaches using temporal logics, we also believe that additional generalizations to the BR/U paradigm such as distinguishing between observations and action effects, or concurrent update can be added to MPMA or at least fully understood in this context.

Acknowledgements

Patrick Doherty and Witold Łukaszewicz have been partially supported by grants from the Wallenberg Foundation and the Swedish Research Council for Engineering Sciences (TFR). Ewa Madalińska-Bugaj is supported by grant KBN 8 T11C 035 111. Thanks to Jonas Kvarnström and Patrik Haslum for comments.

References

[1] G. Brewka, J. Hertzberg. How to Do Things with Worlds: on Formalizing Actions and Plans. *Journal of Logic and Computation*. Vol. 3, No. 5, 1993, 517-532.

[2] F. M. Brown. *Boolean Reasoning*. Kluwer Academic Publishers, 1990.

[3] M-O. Cordier, P. Siegel. A Temporal Revision Model for Reasoning about World Change. In *Int'l. Conference on Principles of Knowledge Representation and Reasoning*, 732-739, Morgan Kaufmann, 1992.

[4] E. W. Dijkstra. *A Discipline of Programming*. Prentice Hall, 1976.

[5] E. W. Dijkstra, C. S. Scholten. *Predicate Calculus and Program Semantics*. Springer-Verlag, 1990.

[6] P. Doherty. Reasoning about Action and Change Using Occlusion. In: *Proc. of the 11th European Conference on Artificial Intelligence*, 1994, 401-405.

[7] P. Doherty, W. Łukaszewicz. Circumscribing Features and Fluents. In: *Proc. of the 1st Int'l. Conference on Temporal Logic*, Lecture Notes on Artificial Intelligence, vol. **827**, Springer-Verlag, 1994, 82-100.

[8] P. Doherty, W. Łukaszewicz, A. Szałas. Computing Circumscription Revisited: A Reduction Algorithm. *Journal of Automated Reasoning*, 18, 297-336, 1997.

[9] P. Doherty, W. Łukaszewicz, E. Madalińska-Bugaj. The PMA and Relativizing Minimal Change for Action Update. In: *Proc. of the 6th Int'l Conference on Principles of Knowledge Representation and Reasoning*. Morgan Kaufmann, 1998, 258-269.

[10] L. Fariñas del Cerro, A. Herzig. An Automated Modal Logic for Elementary Changes. In: P. Smets, A. Mamdami, D. Dubois and H. Prade (eds.), *Non-Standard Logics for Automated Reasoning*, Academic Press, 1988, 63-79.

[11] L. Fariñas del Cerro, A. Herzig. A Conditional Logic for Updating in the Possible Model Approach. In: *Proc. of the 18th German Conference on Artificial Intelligence*, Springer-Verlag, LNAI, **861**, 1994.

[12] L. Fariñas del Cerro, A. Herzig. All Other Things Being Equal: On a Notion of Inertia in Conditional Logic. In: E. Ejerhed and S. Lindström (eds.), *Logic, Action and Cognition*, Kluwer Academic Press, to appear.

[13] N. Y. Foo, A. S. Rao. Minimal Change and Maximal Coherence: A Basis for Belief Revision and Reasoning about Actions. *In Proc. of the 11th Int'l. Joint Conference on Artificial Intelligence*, 1989.

[14] A. Herzig. The PMA Revisited. In: *Proc. of the 5th Int'l. Conference on Principles of Knowledge Representation and Reasoning*, Morgan Kaufmann, 1996, 40-50.

[15] G. Grahne. Updates and Counterfactuals. In *Proc. of the 2nd Int'l. Conference on Principles of Knowledge Representation and Reasoning*, Morgan Kaufmann, 1991, 269-276.

[16] J. Gustafsson, P. Doherty. Embracing Occlusion in Specifying the Indirect Effects of Actions. In: *Proc. of the 5th Int'l. Conference on Principles of Knowledge Representation and Reasoning*, Morgan Kaufmann, 1996, 87-98.

[17] H. Katsuno, A. O. Mendelzon. On the Difference between Updating a knowledge base and revising it. In: *Proc. of the 2nd Int'l. Conference on Principles of Knowledge Representation and Reasoning*, KR'91, Morgan Kaufmann, 1991, 387-395.

[18] G.N. Kartha, V. Lifschitz. Actions with Indirect Effects (Preliminary Report). In *Proc. of the 4th Int'l. Conference on Principles of Knowledge Representation and Reasoning*, Morgan Kaufmann, 1994.

[19] V. Lifschitz. Frames in the Space of Situations. *Artificial Intelligence J.*, **46**, 1990, 365-376.

[20] F. Lin. Embracing Causality in Specifying the Indirect Effects of Actions. *In Proc. of the 14th Int'l. Joint Conference on Artificial Intelligence*, 1995.

[21] F. Lin. Embracing Causality in Specifying the Indeterminate Effects of Actions. *In Proc. of the National Conference on Artificial Intelligence*, 1996.

[22] N. McCain, H. Turner. A Causal Theory of Ramifications and Qualifications. In: *Proc. of the 14th Int'l. Joint Conference on Artificial Intelligence*, Morgan Kaufmann, 1995, 1978-1984.

[23] E. Madalińska-Bugaj. How to Solve Qualification and Ramification Using Dijkstra's Semantics for Programming Languages, *AI*IA-97: Advances in Artificial Intelligence*, Springer-Verlag, Lecture Notes in Artificial Intelligence, **1321**, 381-392, 1997.

[24] P. Peppas, W. Wobcke. On the Use of Epistemic Entrenchment in Reasoning about Action. In: *Proc. of the 10th European Conference on Artificial Intelligence*, 1994, 403-407.

[25] E. Sandewall. *Features and Fluents. A Systematic Approach to the Representation of Knowledge about Dynamical Systems.* Volume 1. Oxford University Press, 1994.

[26] E. Sandewall. Assessments of Ramification Methods that use Static Domain Constraints. In *Proc. of the 5th Int'l. Conference on Principles of Knowledge Representation and Reasoning*, Morgan Kaufmann, 1996.

[27] M. Thielscher. Computing Ramifications by Postprocessing. In C. S. Mellish, editor, *Proc. of the Int'l. Joint Conference on Artificial Intelligence (IJCAI)*, pages 1994–2000, Montreal, Canada, August 1995. Morgan Kaufmann.

[28] M. Thielscher. Ramification and causality. *Artificial Intelligence J.*, 1997.

[29] A. del Val, Y. Shoham. Deriving Properties of Belief Update from Theories of Actions (II). In *Proc. of the 13th Int'l. Joint Conference on Artificial Intelligence*, pages 732–737, 1993.

[30] M. Winslett. Reasoning about Action Using a Possible Model Approach. In: *Proc. AAAI-88*, St. Paul, MN, 1988, 89-93.

[31] M. Winslett. *Updating Logical Databases.* Cambridge Tracks in Theoretical Computer Science. Cambridge University Press, 1990.

[32] M. Winslett. Updating Logical Databases. In: D. Gabbay, A. Galton, J. A. Robinson (eds.). *Handbook of Logic in Artificial Intelligence and Logic Programming.* Vol. 4 (Epistemic and Temporal Reasoning), 1995, Oxford University Press, 133-174.