Updating, Transition Constraints and Possibilistic Markov Chains

Didier DUBOIS – Florence DUPIN de SAINT CYR – Henri PRADE

Institut de Recherche en Informatique de Toulouse (I.R.I.T.) – C.N.R.S.
Université Paul Sabatier, 118 route de Narbonne
31062 TOULOUSE Cedex – FRANCE

Abstract. Possibility theory is applied to the updating problem in a knowledge base that describes the state of an evolving system. The system evolution is described by a possibilistic Markov chain whose agreement with the axioms of updating is examined. Then it is explained how to recover a possibilistic Markov chain from a set of transition constraints, on the basis of a specificity ordering.

1. Introduction

One of the most challenging problems in databases and knowledge-based systems is that of modifying a knowledge base under the arrival of a new piece of information. Basically, if the new information contradicts the contents of the knowledge base, there exists several ways of restoring consistency, unless strict guidelines are supplied that lead to a unique solution. An important distinction has been drawn by Katsuno and Mendelzon [12] between revising a knowledge base and updating it. In revision, the new information is meant to improve our cognitive state regarding a given situation; what was plausibly thought as being true may actually be false. In updating, the new information is meant to inform the knowledge base that something has changed in the actual world; what was thought to be true may no longer be true because things have changed.

This paper is a preliminary investigation about a new approach to updating a knowledge base describing the behaviour of an evolving system. More specifically an attempt is made to unify three points of view on this problem: the point of view of the system analyst who describes the evolution of a system via a transition graph between states, the point of view of formal philosophers who have laid bare the postulates of rational updating, and the point of view of database research, where the update is achieved at the syntactic level by means of transition constraints [4]. It is suggested that such transition constraints partially determine a transition graph between states of the system. This graph can be viewed as a possibilistic Markov chain. The same observation can be derived from Katsuno and Mendelzon's postulates, except that the obtained transition graph underlies an inertia property. Some hints about how to derive Markovian transition graphs from transition constraints which are either strong or defeasible, are suggested, in the framework of possibility theory [16] [7]. Lastly, it should be pointed out that temporal aspects are not explicitly dealt with and, without loosing generality, actions are not formalized here, only their effects are accounted for (e.g., "shoot" is replaced by "shot" and the proposition "shot" is true when the action "shoot" has been executed).

2. Evolution and Uncertainty

There are at least two kinds of evolutive systems: *dynamical* systems and so-called "*inert* systems" [14]. The difference lies more in the point of view that is chosen for studying these systems than in their nature. One can be interested in modelling the transient behaviour of the system under its own dynamics, with a view to controlling its trajectory towards a stable state. On the contrary, in inert systems, the evolution is only due to some external event, and the system is supposed to stay in the same state unless such an event occurs. One is then only interested in computing the resulting state of the system. The main difficulty is then to find the parts of the system which change (this is called the ramification problem) and those which persist (this is called the frame problem, see, e.g., [3]). The well-known "world of blocks" in robotics, where blocks are moved by some manipulator, is a typical example of inert system. A knowledge base which maintains the available information about the world, and whose contents only changes when new information comes in about the world, is another example of inert system. This paper deals with some aspects of updating knowledge about inert systems.

Uncertainty pervades this kind of problem. First the current state of a system may be ill-known. This means that the knowledge base describing its state may be incomplete. Sometimes, probabilistic knowledge about the state of the system is available. But generally, one may only have some idea about what is/are the most plausible state(s) of the system, among possible ones. This type of incomplete knowledge can be described in terms of possibility distributions. Uncertainty may also affect the transitions between states, i.e., the mapping that computes the following state may be one-to-many. This form of uncertainty expresses a form of non-determinism in the behaviour of the system and can be represented by a Markov chain where the set $\Gamma(\omega)$ of potential successors of the state ω is equipped with a probability distribution, $P(\omega'|\omega)$ being the probability of reaching state $\omega' \in \Gamma(\omega)$ when the system is in state ω. Updating means computing the current state of the system, or the most plausible current state taking into account the knowledge on the previous state and receiving a piece of information about the current state.

Formally, let T be a set of time points, at each time point the real system is in a state. Let S be a set of symbols which can be associated to properties of the system. A fluent is a property of the system considered at a given time point. It is of the form p_t where p is a property of S and t a time point of T, and can be true or false. Let F be the set of fluents. The state ω of the system at time t is described by the set of fluents that are true at t. From a set of fluents of F indexed by the same time point t, sentences can be built using classical connectives. Let K_{to} describes our knowledge of the system at a certain time point $t_o \in T$, K_{to} contains three kinds of information:

- a set of sentences at time t_o, which describes *the state of the system* as it is known, maybe incompletely (K_{to} may have several models);

- a set of sentences of the form $\forall t \in T$, X_t where X_t is a sentence at instant t, which are *static rules* that restrict the possible states of the world to a subset of interpretations of the language. They are true at any instant and called *domain constraints;*

- a set of sentences of the form $\forall t \in T$, $p_t \mapsto q_{t+1}$, where p_t and q_{t+1} are sentences, and $\mapsto$ is a non-classical arrow, which are *dynamic rules* that characterize the way the world can change. A rule $p_t \mapsto q_{t+1}$ means that if presently $K_t \vdash p_t$ then the world evolves in such a way that $K_{t+1} \vdash q_{t+1}$ eventually, where K_{t+1} is what is known of the resulting state. They are called *transition constraints*.

Let α_{to+1} be a sentence representing a new piece of information, and called *input*: it can be viewed as a constraint that the result of a change must satisfy.

Some transition constraints are called hard if they must always be respected. Other transition constraints are not hard in the sense that they can be defeated by harder transition constraints that contradict them. These constraints will be called defeasible. The strength of a transition constraint can be modelled by means of a priority ordering, as suggested by Cordier and Siegel [6]. Such priorities may have different meanings. They may express degrees of persistence through time (e.g., "dead" is more persistent than "asleep"); they may also reflect a specificity ordering among transition constraints.

In order to compute the resulting state of an update α_{to+1}, transition constraints are applied to the initial state and a resulting knowledge base K_{to+1} is obtained such that $K_{to+1} \vdash \alpha_{to+1}$ holds.

3. Possibilistic Markov Chains

The natural approach in systems analysis consists in enumerating all possible states of the system under study, and then all possible direct transitions between states, with a possible quantification of the likelihood of these transitions. Having built the state transition graph, it is possible to simulate the system behaviour. The transition graph is said to be Markovian if the next state only depends on the current state, not on the past earlier states. The most widely known type of such transition graphs are probabilistic Markov chains where transitions $\omega \rightarrow \omega'$ are quantified by a conditional probability $P(\omega'|\omega)$, such that $\sum_{\omega'} P(\omega'|\omega)=1$.

In the following we consider *possibilistic* Markov chains, whose transitions $\omega \rightarrow \omega'$ are quantified by conditional possibility distributions $\prod(\omega'|\omega)$ such that $\max_{\omega'} \prod(\omega'|\omega)=1$. Possibilistic Markov chains describe for each state ω the set of more or less plausible next states ω'. $\Pi(\omega'|\omega)=1$ means that nothing prevents ω' from being the state succeeding to ω. $\Pi(\omega'|\omega)=0$ means that the direct transition from ω to ω' is strictly forbidden. $\Pi(\omega'|\omega)>\Pi(\omega''|\omega)$ means that in the absence of further information, one prefers to consider ω' as more plausible successor state to ω than ω''. Compared to a probabilistic Markov chain, a possibilistic Markov chain is a much more qualitative description of a system behaviour. A possibilistic Markov chain can be viewed

i) either as a set of possibility distributions π_ω, one per state $\omega \in \Omega$, where $\prod(\omega'|\omega)=\pi_\omega(\omega')$, $\forall \omega,\omega' \in \Omega$;

ii) or as a ternary relation expressing "closeness" of the form $\omega' <_\omega \omega''$ on Ω^3 where $\omega' <_\omega \omega'' \Leftrightarrow \prod(\omega'|\omega) > \prod(\omega''|\omega)$. It rank-orders all states $\omega' \in \Omega$ according to their proximity (here understood as: plausibility of being the next state) to ω, for all ω.

The meaning of such possibilistic Markov chains differs from probabilistic ones in the way simulation will take place. In the case of probabilistic Markov chains, the next state is chosen at random following the given probabilities. Simulation of a possibilistic Markov chain is performed differently. The chosen next state is the most plausible one, i.e., ω' such that $\Pi(\omega'|\omega)$ is maximal, given current knowledge about what this next state is. When the most plausible next state is not unique, the simulation becomes non-deterministic. In the absence of knowledge about the next state, all ω' such that $\Pi(\omega'|\omega)=1$ are selected. However we may know from some external source that the next state lies in A. Then the most plausible next state is $\omega' \in A$ such that $\Pi(\omega'|\omega)=\Pi_\omega(A)$, where $\Pi_\omega(A)= \sup\{\Pi(\omega''|\omega),\omega'' \in A\}$. This is how the updating problem is addressed. Note that this mode of simulation could be applied to probabilistic Markov chains (always choosing the most probable next state).

Example: The door-to-door shuttle

Assume a taxicab is running according to customer calls, and a central broadcast station communicates with it, either to receive information about what the cab is doing, or to transmit the call of a new customer. We denote by H the customer house, and D the destination of the customer.

The meanings of the literals are the following. I: idle, C: a call has been received, G_H: taxicab going to H, H: taxicab is at H, G_D: taxicab going to D. G_D, G_H, H, I are mutually exclusive, and in this example it is assumed that no new call is registered by the taxicab before taking the current customer. Moreover if a call has not been received, the taxi cannot go to the client's home nor be there, so the states H $\bar{C}$ and G_H $\bar{C}$ are impossible.

The transition graph modelling the normal behaviour of a taxi is on Fig. 1. G_H C, G_D C, G_D $\bar{C}$, I $\bar{C}$ are persistent states. For instance, when a taxicab is idle, without any customer, it remains so until new information comes in; if a customer calls then the state changes to IC. On the contrary IC and HC are transient states. The difference between transient and persistent states here is chosen so as to distinguish between short activities (letting a customer into the car) and long ones (a drive). A persistent state ω is such that $\Pi(\omega|\omega)=1>\Pi(\omega'|\omega)$ if $\omega\neq\omega'$. A transient state is such that $\Pi(\omega|\omega)<1$. Using this kind of modelling, update takes place upon receiving new information as follows: assume the current state is G_D $\bar{C}$. If a new information comes in saying the cab is now idle, then the possible current states are IC and I $\bar{C}$, and most plausibly I $\bar{C}$.

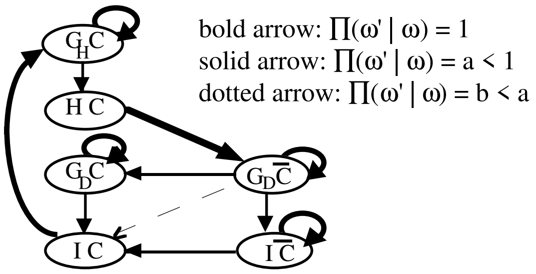

Fig. 1. Possibilistic Markov chain

4. A Possibilistic Analysis of Update Postulates

Katsuno and Mendelzon [12] have suggested 8 postulates that should be satisfied when updating a knowledge base K with a new information α. The arrival of α is supposed to be caused by an evolution in the system whose state is (incompletely) described in K: now things have changed in such a way that α is true. In the following K is a set of propositional sentences whose conjunction is ϕ; α is a propositional sentence. ϕ & α denotes the result of updating K by α. The set of postulates are as follows:

U1 $\phi\&\alpha\vdash\alpha$; U2 $\phi\vdash\alpha\Rightarrow\phi\&\alpha=\phi$
U3 if ϕ and α are satisfiable then so is $\phi\&\alpha$
U4 if $\phi=\phi'$ then $\phi\&\alpha=\phi'\&\alpha$
U5 $(\phi\&\alpha)\wedge\beta\vdash\phi\&(\alpha\wedge\beta)$
U6 if $\phi\&\alpha\vdash\beta$ and $\phi\&\beta\vdash\alpha$ then $\phi\&\alpha=\phi\&\beta$
U7 if ϕ is complete then $(\phi\&\alpha)\wedge(\phi\&\beta)\vdash\phi\&(\alpha\vee\beta)$
U8 $(\phi_1\vee\phi_2)\&\alpha=(\phi_1\&\alpha)\vee(\phi_2\&\alpha)$.

Katsuno and Mendelzon have proved that an update operation satisfies U1-U8 if and only if there exists a partial ordering that satisfies $\omega<_\omega\omega'$, $\forall\omega'\neq\omega$, such that, the set of models of $\phi\&\alpha$ is $\{\omega' \mid \omega\models\alpha, \exists\omega\models\phi, \omega'\leq_\omega\omega'', \forall\omega''\models\alpha\}$. In other words, $\phi\&\alpha$ is obtained by selecting for each model ω of ϕ the models of α closest to ω in the sense of the partial ordering $\leq_\omega$. It is clear that the set of partial orderings $\{\leq_\omega,\omega\in\Omega\}$ that underlie the postulates U1-U8 are very similar to possibilistic Markov chains on Ω (set of interpretations of the language on which K and α are built). However there are some differences: first, the orderings $\{\leq_\omega,\omega\in\Omega\}$ induced by a possibilistic Markov chain are complete, while the ordering relations in Katsuno and Mendelzon are partial. A complete preorder is derived from the postulates U1-U8 and the following additional weak converse of U5:

U9 if ϕ is complete and $(\phi\&\alpha)\wedge\beta$ is satisfiable then $\phi\&(\alpha\wedge\beta)\vdash(\phi\&\alpha)\wedge\beta$.

When ϕ is complete, ϕ corresponds to a single state ω; let $\alpha(\omega)$ be the set of most plausible next states after ω, that satisfy α. Axioms U5 and U9 say that $\alpha(\omega)\cap[\beta]=(\alpha\wedge\beta)(\omega)$, i.e., the set of most plausible next states that satisfy α, which also satisfy β should coincide with the best next states in $[\alpha\wedge\beta]$, where $[\beta]$ is the set of models of β. This is due to the fact that if one maximises a function f over a set A, then if the set $A^*{}_f\cap B$ of maxima of f over A that belong to B is non empty then it is equal to $(A\cap B)^*{}_f$.

There are other more noticeable differences between possibilistic Markov chains and the type of graph obtained by means of the postulates. U2 is an inertia postulate: if $\phi\vdash\alpha$, then the truth of α is already known and the new information causes no change. Hence as a particular case, letting $[\phi]=\{\omega\}$ and $\alpha=\top$, we get $\omega<_\omega\omega'$, $\forall\omega'\neq\omega$, which in possibilistic terminology reads $\prod(\omega|\omega)=1>\prod(\omega'|\omega)$, $\forall\omega'\neq\omega$. In other words, all states are persistent. Note that this assumption may look debatable, e.g., in the previous example. Besides, U3 implies that $\forall\omega'\in\Omega$, $\prod(\omega'|\omega)\neq 0$. Indeed, if it is not the case, the result of updating ϕ such that $[\phi]=\{\omega\}$ by α such that $[\alpha]=\{\omega'\}$ leads to an impossible situation. Hence according to Katsuno and Mendelzon any state can be attained from any other state, so that any constraint on the next state can be satisfied via one transition only. Again this assumption may look too strong. As proved by Dubois and Prade [9], axiom U8, cast in the possibilistic setting, is nothing but Zadeh's extension principle which considers the function f that maps each ω to the next plausible states that satisfy α, and extends this function to the set of models of ϕ. More generally, we can define the models of $\phi\&\alpha$ as follows:

$$[\phi\&\alpha] = f([\phi]) \quad \text{with } f(\omega)=\mathrm{ArgMax}_{\omega'}\, \mu_{[\alpha]\cap R(\omega)}(\omega'),$$

where "ArgMax" retrieves the arguments ω' which here maximize the considered expression, μ denotes membership functions, and $R(\omega)$ is the fuzzy set of states that are successors of ω (whose membership function is π_ω) and $\mu_{[\alpha]\cap R(\omega)}=\min(\mu_{[\alpha]},\mu_{R(\omega)})$. Updating at the semantic level using the possibilistic Markov chain $\prod$ means predicting the states ω' that plausibly follows each model ω of ϕ, deleting those which fail to satisfy α, and keeping only the most plausible ones in α. For instance suppose that the current state of the taxicab is that it is either letting a customer into the car, or going to its destination, or idle, with no new customer call. Hence $[\phi]=\{HC, G_D\bar{C}, I\bar{C}\}$. Assume the new information is that it is idle. The result of the update by α such that $[\alpha]=\{IC, I\bar{C}\}$ is $\{I\bar{C}\}$. But the other state IC is reachable (from $I\bar{C}$ and from $G_D\bar{C}$) with lower plausibility.

Note that if ϕ is complete, i.e., $[\phi]=\{\omega\}$, we have $\phi\&\alpha\vdash\beta$ if and only if $\prod_\omega(\beta\wedge\alpha)>\prod_\omega(\neg\beta\wedge\alpha)$ where $\prod_\omega$ is the possibility measure based on the possibility distribution $\pi_\omega=\prod(\cdot|\omega)$, i.e., $\prod_\omega(\gamma)=\sup_{\omega'\models\gamma}\pi_\omega(\omega')$. Equivalently $\phi\&\alpha\vdash\beta$ if and only if $N_\omega(\beta|\alpha)>0$ where N_ω is the conditional necessity associated with π_ω. See [8] for the semantics of preferential models expressed in terms of conditional necessity. More generally, we conclude that $\phi\&\alpha\vdash\beta$ if and only if $\forall\omega\in[\phi]$, $N_\omega(\beta|\alpha)>0$, i.e., $N(\beta|\alpha)=\inf_{\omega\models\phi}N_\omega(\beta|\alpha)>0$. $N(\beta|\alpha)$ is still a necessity measure which is based on the possibility distribution $\max_{\omega\models\phi}\pi_\omega(\cdot|\alpha)$ where

$$\pi_\omega(\omega'|\alpha)=\begin{cases}1 & \text{if } \pi_\omega(\omega')=\prod_\omega(\alpha),\ \omega'\models\alpha\\ \pi_\omega(\omega') & \text{if } \pi_\omega(\omega')<\pi_\omega(\alpha), \omega'\models\alpha\\ 0 & \text{if } \omega'\not\models\alpha\end{cases}$$

The above results point out that the decision $\phi\&\alpha\vdash\beta$ cannot be computed by a simple aggregation of possibility distributions π_ω for $\omega\models\phi$. Each possibility distribution π_ω

must be revised and only the $\pi_\omega(\cdot|\alpha)$ can be combined. The need for reasoning state by state is typical of the updating problem and is embodied by axiom U8.

5. From Transition Constraints to Possibilistic Markov Chains

Where do possibilistic Markov chains come from? How can the possibilities of the transitions be computed? A way for rank-ordering the transitions between states is to take advantage of the relative specificity of the condition parts of the transition constraints. This can be done either by computing directly a possibilistic ordering between pairs of states or by computing a rank-ordering between transition constraints viewed as default rules [1] and then deducing the ordering between pairs of states.

- the first method consists in considering each transition constraint $p_t \mapsto q_{t+1}$ as a constraint specifying *feasible possibility distributions on* $\Omega_t \times \Omega_{t+1}$ (Ω_t represents the set of the system states at time t). Each transition constraint $p_t \mapsto q_{t+1}$ means implicitly that when p is known at time t, the next state more plausibly satisfies q than ¬q. The chosen ordering on transitions is then encoded by the least specific possibility distribution π on $\Omega_t \times \Omega_{t+1}$ that satisfies the set of inequality constraints $\{\Pi(p_t \wedge q_{t+1}) > \Pi(p_t \wedge \neg q_{t+1})$ for $p_t \mapsto q_{t+1} \in C\}$ and which assigns the greatest possible possibility degrees to pairs of interpretations in $\Omega_t \times \Omega_{t+1}$.

- the second method consists, first, in rank-ordering the transition constraints (considered as classical default rules) in priority levels, using Pearl's algorithm [13]. Namely, the idea is to partition C into an ordered set $\{C_0, C_1, \ldots, C_k\}$ such that transition constraints in C_i are tolerated by all transition constraints in $C_{i+1} \cup \ldots \cup C_k$. A transition constraint $p_t \mapsto q_{t+1}$ is tolerated by a set $\{pi_t \mapsto qi_{t+1}, i=1 \ldots m\}$ if and only if $\{p_t \wedge q_{t+1}, \neg p1_t \vee q1_{t+1}, \ldots, \neg pm_t \vee qm_{t+1}\}$ is consistent. The level C_0 contains the less specific transition constraints and C_k the most specific ones. From this partition, transition between pairs of states can be compared :

$\pi(\omega_1,\omega_2) \geq \pi(\omega'_1,\omega'_2) \Leftrightarrow \{p_t \rightarrow q_{t+1} | \omega_1 \vDash \neg p \text{ or } \omega_2 \vDash q\} \geq_{bo} \{p'_t \rightarrow q'_{t+1} | \omega'_1 \vDash \neg p' \text{ or } \omega_2' \vDash q'\}$

where $\geq_{bo}$ is the "best out" ordering defined in [2]. More precisely if A and B are two sets of default rules : $A \geq_{bo} B$ iff $a(A) \geq a(B)$ where $a(A) = \min_i\{p_t \rightarrow q_{t+1} \notin A \cap C_i\}$. In other words a transition (ω_1,ω_2) is preferred to (ω'_1,ω'_2), if (ω_1,ω_2) violates transition constraints that are less specific (i.e., less important) than those violated by (ω'_1,ω'_2).

The two methods give the same rank-ordering between states. The proof has been made by Benferhat and al. [1] with usual default rules and can easily be generalized to our type of default rule.

Example: The dead battery (Goldszmidt and Pearl [11])

Let $C=\{lo_t \mapsto db_{t+1}, db_t \wedge tk_t \mapsto \neg cs_{t+1}, tk_t \mapsto cs_{t+1}\}$, where lo=lights on all night long, db=dead battery, tk=ignition key had been turned, cs=car starting. We restrict the set of states to the models of lo ∨ tk so that at least one rule applies. There are thus 12 states of interest

	lo tk	lo ¬tk	¬lo tk
cs db	ω_1	ω_5	ω_9
cs ¬db	ω_2	ω_6	ω_{10}
¬cs db	ω_3	ω_7	ω_{11}
¬cs ¬db	ω_4	ω_8	ω_{12}

The 3 transition constraints are modelled as follows: $\Pi(lo_t \wedge db_{t+1}) > \Pi(lo_t \wedge \neg db_{t+1})$, $\Pi(db_t \wedge tk_t \wedge \neg cs_{t+1}) > \Pi(db_t \wedge tk_t \wedge cs_{t+1})$, $\Pi(tk_t \wedge cs_{t+1}) > \Pi(tk_t \wedge \neg cs_{t+1})$.
Maximizing the possibility degrees, one easily finds that in $\Omega_t \times \Omega_{t+1}$

- the models (ω_t, ω_{t+1}) of $(\neg lo_t \vee db_{t+1}) \wedge (\neg db_t \vee \neg tk_t \vee \neg cs_{t+1}) \wedge (\neg tk_t \vee cs_{t+1})$ have maximal possibility
- the models of $db_t \wedge tk_t \wedge cs_{t+1}$ have minimal possibility
- the other models have intermediary possibility.

Taking into account inertia, it yields the following ranking of the transitions:

π	Transitions
high (= 1)	$(\omega_2,\omega_1),(\omega_4,\omega_1),(\omega_5,\omega_5),(\omega_6,\omega_5),$ $(\omega_{10},\omega_{10}),(\omega_{12},\omega_{10}),(\omega_8,\omega_7),(\omega_7,\omega_7)$
medium	$(\omega_1,\omega_3),(\omega_3,\omega_3),(\omega_9,\omega_{11}),(\omega_{11},\omega_{11})$
low	$\{\omega_1,\omega_9,\omega_3,\omega_{11}\} \times \{\omega_1,\omega_2,\omega_5,\omega_6,\omega_9,\omega_{10}\}$

Note that if the transition constraint $tk_t \mapsto ks_{t+1}$ is added to the knowledge base whith ks meaning that the key is in the starter then from an initial state where $tk \wedge db$ is true, this method can conclude that in the next state $\neg cs$ will be true but cannot conclude that ks will also be true. It is the *drowning effect* [2].

6. Prediction and updating

Once the transitions are rank-ordered, updating and prediction can be envisaged according to two different ways : either on the basis of a state by state analysis à la Katsuno and Mendelzon in which all the most plausible transitions are considered from *each* possible initial state ω, or by means of a default analysis where the most plausible transitions are applied to the most normal state corresponding with what is known of the initial state.

6.1. The state by state analysis

Updating in this method consists, as it is suggested by Katsuno and Mendelzon, in considering each possible initial state, then computing each best final state satisfying the input, and finally doing the union of all final states obtained. But in order to compute each best final state, a ternary relation $\{<_\omega, \omega \in \Omega\}$ is needed. Where does it come from? This problem is left open by these authors. Some researchers have tried to give an answer to that problem, and noticeably Winslett [15]. She derives the set of orderings $\{<_\omega, \omega \in \Omega\}$ right away from the language. Namely assume that the language is made of n propositional letters $x_1, \ldots, x_n$; let x_i^* be a literal defined from x_i, i.e., $x_i^* \in \{x_i, \neg x_i\}$. A state ω is equated to an interpretation $x_1^* \wedge x_2^* \wedge \ldots \wedge x_n^*$. Let $D(\omega,\omega')$= {literals whose sign differs in ω and ω'}. Then the ternary relation proposed by Winslett is such that $\omega' <_\omega \omega''$ iff $D(\omega,\omega') \subset D(\omega,\omega'')$. In other words the most plausible transitions are those which tend not to change the signs of literals. This proposal has the drawback of being entirely language-driven. It presupposes that distinct propositional letters correspond to independent properties. In this paper, it has been suggested that the state transitions can also be syntactically specified by means of transition constraints attached to the knowledge base and we use the rank-ordering defined above. Winslett's method can then be used to complete the possibilistic Markov chain obtained by the method described above using persistence assumptions attached to literals.

Let K_{t0} be a knowledge base. A set C of transition constraints is supposed to be available. It contains assertions of the form $p_t \mapsto q_{t+1}$, as already said (those transitions can be hard or defeasible). For each state ω, we can compute the set of successor states as follows. Let $C(\omega)=\{p_t \mapsto q_{t+1}, \omega \models p\}$ be the set of transition constraints that are enabled in state ω. Then the set of successor state of ω is defined as $S(\omega)=\{q \mid p_t \mapsto q_{t+1} \in C_i(\omega)$ $\nexists\, p'_t \mapsto q'_{t+1}$, s.t. $p'_t \mapsto q'_{t+1} \in C_j(\omega), j<i\}$, where $C_i(\omega) = C(\omega) \cap C_i$, and $\{C_0, C_1, \ldots, C_k\}$ is the partition computed in Section 5. Namely we delete rules $p_t \mapsto q_{t+1}$ such that more specific rules $p'_t \mapsto q'_{t+1}$ can be applied to ω. We further assume that $S(\omega)$ will be consistent (otherwise possible successors of ω are undetermined). The set of models of $S(\omega)$, $[S(\omega)]$ is the set of possible states succeeding to ω. The most plausible among these states will be the ones which maximize inertia, i.e., $\{\omega' \in [S(\omega)]$, $D(\omega,\omega')$ is minimal$\}$. If $S(\omega)$ is inconsistent then we can introduce a priority ordering among conflicting rules and select the most prioritary ones. Such an ordering can discriminate among more or less persistent literals as in Cordier and Siegel [6].

Example (continued) We now compute the most plausible transitions from complete states of knowledge obeying C and maximizing inertia. For instance, if we consider state ω_1 where the lights have been on, the battery is dead, the key has been turned and the car is starting, it is a transient state which is necessarily followed by ω_3 where the information cs is turned into ¬cs due to the transition constraints. Indeed, $S(\omega_1)=\{db_{t+1}, \neg cs_{t+1}\}$, which is completed by adding lo_{t+1} and tk_{t+1} by inertia. Using the same principle, ω_3 is clearly a persistent state.
On the whole the reader can check that the following transition graph is obtained:

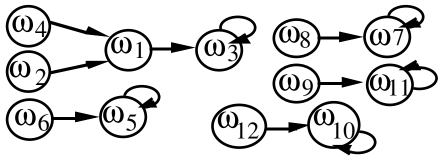

Fig. 2. The most plausible transitions

The arrows express the most plausible transitions (according to inertia and specificity). There are 5 persistent states: $\omega_3=\{lo,db,tk, \neg cs\}$, $\omega_5=\{lo,db,\neg tk,cs\}$, $\omega_7=\{lo,db,\neg tk,\neg cs\}$, $\omega_{10}=\{\neg lo,\neg db,tk,cs\}$, $\omega_{11}=\{\neg lo,db,tk,\neg cs\}$.

Note that ω_5 acknowledges the possibility of starting a car with a dead battery by other means than simply turning the key. This possibility is not ruled out by the set of transition constraints.

6.2 The default method

The above method has a drawback, it gives rather imprecise results (even if only the most plausible transitions are taken into account). For instance, if the initial knowledge base contains $lo \wedge tk$ it is clear that the resulting persistent state is ω_3. However if it contains tk only, then the final state is any of ω_3, ω_{10} or ω_{11}, hence it is not clear if the car starts. But from the transition constraints one should by default conclude that cs follows from tk, hence only ω_{10} should remain.

The default method consists in considering the most plausible transition constraints which can be applied to the initial state as it is known. In other words, the implicit assumption that the system is in *the most normal case* is made, so it gives us a precise but not absolutely sure result (because if the initial state were known more precisely then possibly more specific but opposite transition constraints should have been applied). For instance, if all that is known of the current state at time t_0 is that tk holds then the only possible persistent states are $\{\omega_3,\omega_{10},\omega_{11}\}$, and ω_{10} is the most likely to be reached, whereby we can conclude by default $\neg lo_{to+1}$, $\neg db_{to+1}$, cs_{to+1}, as expected.

In the case of *updating* this method must be applied more cautiously than in the case of prediction, the result must be checked in order to know if it verifies the input. If the result does, then there is no reason to question it, otherwise we must consider that the implicit asumption made about the initial state might be not true and that we should use the state by state method.

As already said, we might want to add persistence transitions à la Winslett, like $db_t \mapsto db_{t+1}$, $lo_t \mapsto lo_{t+1}$, $\neg db_t \mapsto \neg db_{t+1}$, $\neg lo_t \mapsto \neg lo_{t+1}$. This corresponds to new constraints which would restrict the transitions having the highest possibility level. Note that in the case of a complete knowledge of the initial state the two methods presented in this part give the same result.

7. Link with Other Approaches

Cordier and Siegel [5] handle transition constraints in the framework of default logic. Hence they cannot cope with rule specificity in a simple way. Moreover they assume that the initial state is complete. There are some connections between our proposal and the more recent semantic transition model of Cordier and Siegel [6]. In their model the transition constraints are rank-ordered right away in terms of persistence, and are then encoded in a circumscriptive logic with priorities. On the contrary, here we automatically rank the transition constraints so as to respect the specificity ordering, and we can then rank the transitions between states accordingly. The two approaches are complementary since some information about the relative strength of persistence of some litterals may be available (for instance for a regularly maintained car ¬db should persist more than db). Note that this ranking procedure fails if some transition constraints bluntly contradict one another. Our approach also has same connections with Goldszmidt and Pearl's [11] stratified ranking approaches to updating, and especially the idea of going from transition constraints to a ranking of models. However in their approach Goldszmidt and Pearl consider only causal problems, such as the dead battery above, for which the transition constraints form a circuit-free graph. Clearly the above approach is meant to work if there are circuits, since once each transition constraint $p \rightarrow q$ has been written $p_t \mapsto q_{t+1}$ all possible circuits have been broken. On the contrary, examples that can be dealt with using stratified rankings are ones where the system evolves once for all and then "stops".

Our analysis of Katsuno and Mendelzon's postulates has led us to define a new set of update postulates see [10] which are exactly describing the behaviour of every possibilistic Markovian graph (where inertia principles are not taken for granted).

8. Further Research

A direction of study is to find an economic way of expressing the persistent constraints of the form $p_t \mapsto p_{t+1}$ and $\neg p_t \mapsto \neg p_{t+1}$ for each p of the set of symbols S. More work is needed to assess the level of generality and the limitations of the proposed treatment of transition constraints, specially their rank-ordering : dynamic constraints expressing change have clearly a higher priority degree than persistence constraints. Two automatic methods to rank-order dynamic change constraints have been presented in this paper. The rank-ordering of persistent constraints is to be studied, since as in Cordier and Siegel [6], we should take into account differences between degrees of persistence. Moreover, computing the transition graph is the semantic aspect of updating, there is another aspect to develop:

the syntactic one, i.e., how to compute updating without being obliged to develop all the possible states of the system...

Acknowledgements

This study has been partially supported by the French Computer Science Inter-P.R.C. project "Gestion de l'Evolutif et de l'Incertain" of the M.E.S.R. The authors are indebted to Marie-Odile Cordier for many comments and discussions which help them to prepare the final version of this paper.

References

1. Benferhat S., Dubois D., Prade H. (1992) Representing default rules in possibilistic logic. Proc. 3rd Conf. on Principles of Knowledge Representation and Reasoning (KR'92) (B. Nebel et al., eds.), Cambridge, MA, 673-684.
2. Benferhat S., Cayrol C., Dubois D., Lang J., Prade H. (1993) Inconsistency management and prioritized syntax-based entailment. Proc. 13th Int. Joint Conf. on Artificial Intelligence (IJCAI'93), 640-645, Chambéry.
3. Brown F. (Ed.) (1988) The Frame Problem in Artificial Intelligence. Morgan & Kaufmann, San Mateo, CA.
4. Cholvy L. (1994) Database updates and transition constraints. Int. J. Intelligent Systems, 9, 169-180.
5. Cordier M.O., Siegel P. (1992) A temporal revision model for reasoning about change. Proc. 3rd Conf. on Principles of Knowledge Representation and Reasoning (KR'92) (B. Nebel and al., eds.), Cambridge, MA, 732-739.
6. Cordier M.O., Siegel P. (1993) Prioritized transitions for updates. Proc. AI'93 Workshop on "Belief Change: Bridging the Gap Between Theory and Practice", Melbourne and Technical report IRISA, n 884, 1994, Rennes, France.
7. Dubois D., Prade H. (1988) Possibility Theory. Plenum Press, New York.
8. Dubois D., Prade H. (1991) Possibilistic logic, preferential models, non-monotonicity and related issues. Proc. 12th Int. Joint Conf. on Artificial Intelligence (IJCAI'91), Sydney, Australia, Aug. 24-30, 419-424.
9. Dubois D., Prade H.(1993) Belief revision and updates in numerical formalisms. Proc. 13th Int. Joint Conf. on Artificial Intelligence (IJCAI'90), 620-625, Chambéry.
10. Dubois D., Dupin de Saint-Cyr F., Prade H. (1995) Update postulates without inertia. Inter. Rep. IRIT, Toulouse University, France.
11. Goldszmidt M., Pearl J. (1992) Rank-based systems: A simple approach to belief revision belief update and reasoning about evidence and actions. Proc. 3rd Conf. on Principles of Knowledge Representation and Reasoning (KR'92) (B. Nebel and al., eds.), Cambridge, MA, 661-672.
12. Katsuno H., Mendelzon A.O. (1991) On the difference between updating a knowledge base and revising it. Proc. 2nd Conf. on Principles of Knowledge Representation and Reasoning (KR'91) (J. Allen et al., eds.), Cambridge, MA, 387-394.
13. Pearl J. (1990) System Z: a natural ordering of defaults with tractable applications to default reasoning. Proc 3rd Conf. on Theoretical Aspects of Reasoning About Knowledge (M. Vardi, ed.), Morgan Kaufman, San Mateo, CA, 121-135.
14. Sandewall E. (1993) The range of applicability of nonmonotonic logics for the inertia problem. Proc. 13th Int. Joint Conf. on Artificial Intelligence (IJCAI'93), 738-743, Chambéry.
15. Winslett M. (1990) Updating Logical Databases. Cambridge University Press, Cambridge, UK.
16. Zadeh L.A. (1978) Fuzzy sets as a basis for a theory of possibility. Fuzzy Sets and Systems, 1, 3-28.