Concurrent Actions and Changes in the Situation Calculus*

José Júlio Alferes **Renwei Li** **Luís Moniz Pereira**
AI Centre, Uninova and DCS, U. Nova de Lisboa
2825 Monte da Caparica, Portugal

Abstract In this paper we investigate concurrent actions and changes. We extend the standard situation calculus with concurrent actions and show that the extended situation calculus has the same expressive power as the original one. In the extended situation calculus we identify and focus on the composition problem which relates effects of concurrent actions to those of their component actions. A defeasible solution to the composition problem is proposed. The believability of some choices of default rules for the composition problem and atomicity of actions are discussed. We also provide some simple examples to illustrate the usefulness of our defeat rules. The result of this paper has been used in an abductive planner based on extended logic programs with explicit negation.

1 Introduction

The situation calculus [11] is a very general framework for reasoning about actions and changes. Recent investigations have shown that the situation calculus is very useful in many AI applications such as prediction, explanation, planning, and natural language understanding.

In this paper we will particularly investigate concurrent actions and their effects. In reality, many actions overlap in time, which complicate the temporal prediction and explanation problem in AI. For some definite goal some actions may be planned to be carried out at the same time in order to save time, or decrease production cost, or for many other context-dependent purposes. Admitting concurrent actions in formalisms for reasoning about change is so important that Shoham took it as a requirement [20]. In the survey talk [21] temporal reasoning methods in AI were classified into two approaches: the change-based approach (such as the situation calculus) and the time-based approach, and seven problems (limitations) were identified with temporal reasoning systems: instantaneous actions, instantaneous and immediate effects, concurrent actions, continuous processes, qualification problem, ramification problem and the frame problem. It was also pointed out that the first four of the seven problems arise due to the choice of the particular formalism such as situation calculus, and the other three have a global nature and do not result from particular choices of formalism. In this paper we will show that it is actually possible to allow for concurrent actions in the situation calculus. Admitting concurrent actions in the situation calculus does not increase its expressive power. In addition, a new problem (we term it "composition problem" and define it in Section 3) arises in the extended situation calculus with concurrent actions. It will be indicated that the composition problem is global in the sense that it also exists in many other formalisms for reasoning about concurrent actions. After we had finished the work of this paper, we found three other similar work, namely, Gelfond, Lifschitz and Rabinov [4], Lin and Shoham [9] and Baral and Gelfond [1]. The technical differences still exist among our work and theirs, as discussed in Section 8.

The rest of the paper is organized as follows. In Section 2 we extend the situation calculus of [11] with concurrent actions, and show that the extended situation calculus has the same

*Published in: *Proc. of IBERAMIA 94*, Hector Geffner (ed.), McGraw-Hill, 1994, pp. 93 - 104

expressive power as the original one. In Section 3 we identify a problem, termed composition problem, concerning describing effects of concurrent actions. The composition problem is simply as follows: given effects of atomic actions how do we avoid *unnecessary* descriptions of effects of concurrent actions? It will be indicated that the composition problem also exists in other formalisms for reasoning about concurrent actions. In Section 4 we make use of defeasible rules of inference to solve the composition problem. In Section 5 we give some examples to show the usefulness of the defeasible rules of inference in the extended situation calculus. In Section 6 we discuss atomicity of actions and modify the defeasible rules proposed in Section 4. In Section 7 we discuss the believability of defaults for reasoning about concurrent actions. In Section 8 we compare our work with others and conclude with a brief discussion. All the proofs are omitted for brevity, but can be found in the extended version of this paper.

2 Situation calculus with concurrent actions

In the situation calculus there are three basic elements: situations, fluents and actions [11]. A situation s is the complete state of the universe at an instant (or in a period) of time. In order to give partial information about situations, [11] introduced the concept of *fluent*. A fluent is a function whose domain is the set of all situations. If the range of the function is $\{true, false\}$, then it is called a *propositional fluent*. If its range is the set of all situations, then it is called a *situational fluent*. Actions are things which can happen or can be performed in the world. Performing actions can change the truth values of propositional fluents. Following [11] we use $Result(a, s)$ to represent situational change and $Holds(p, s)$ to assert something is true of situations, where a is an action, s a situation, and p a property. Note that actions, situations, and fluents are all logical terms. For example, the action term $Puton(A, B)$ could be used to denote the action in which block A is placed on top of block B. Similarly, the fluent term $On(A, B)$ could represent that A is on top of B. The predicates in situation calculus are used primarily to make statements about the values of fluents in particular situations. For example, $Holds(On(A, B), Result(Puton(A, B), s))$ could be used to denote that the fluent $On(A, B)$ has value *true* in the situation resulting from putting block A on top of B. In the remainder of this paper we will simply write H to stand for $Holds$ and R for $Result$. The formulas such as $H(On(A, B), R(Puton(A, B), s))$ are regarded as effect descriptions of actions. In general, effects of actions can be described by formulas of the form: $H(\phi, s) \rightarrow H(\psi, R(a, s))$, which means that if fluent ϕ is *true* in situation s, then fluent ψ is *true* in the situation resulting from the performance of action a in situation s. When the situation calculus is applied to a practical universe of discourse, some formulas of the above forms may be given as non-logical axioms (called domain-specific axioms), from which one can reason about change by use of first-order logic. In addition, we admit that the following formulas are valid[1]: $H(\neg p, s) \leftrightarrow \neg H(p, s)$ and $H(p \wedge q, s) \leftrightarrow (H(p, s) \wedge H(q, s))$. In [11] no basic domain-independent axioms were given. A study of the situation calculus may start with some simple acceptable and intuitive assumptions as domain-independent axioms. For this purpose Reiter gave some basic domain-independent axioms and developed an inductive technique to prove properties of states in the situation calculus [18]. In this paper we will generally consider a situation calculus as a first-order theory C which can be defined by giving its domain-independent axioms. Applying first-order proof rules (together with some inference mechanisms for frame problem, qualification problem and ramification problem) to the domain-independent axioms of C, we can derive all the theorems of C. Note that we may have different situation calculi if we are given different sets of domain-independent axioms.

Now we extend the situation calculus of [11] with concurrent actions as follows. Let a and

[1] We should point out that there is a logical difference between appearances of symbols $\neg$ and $\wedge$ at the two different levels: $\neg$ and $\wedge$ appearing at the term level are functions, while they appearing at the formula level are logical connectives.

b be any two action (terms), then $a||b$ is an action (term). Thus, by recursion the expression $a_1||a_2||\cdots||a_n$ is an action term if a_1, a_2, $\cdots$, a_n are all action terms. We use $R(a||b, s)$ to denote a new situation which is the result of performing actions a and b at the same time in situation s. In general, the situation $R(a||b, s)$ is different from both $R(a, s)$ and $R(b, s)$. For example, $R(Switchon||Shoot, s)$ could be used to denote a new situation from s by switching on the light and shooting at the same time. Thus, we can assert

$$H(Lighton, R(Switchon||Shoot, s))$$
$$H(Loaded, s) \rightarrow H(Dead, R(Switchon||Shoot, s))$$

These formulas can be regarded as the effect of $(Switchon||Shoot)$. It should be clear that $R(Switchon||Shoot, s)$ is not the same as $R(Switchon, s)$, nor as $R(Shoot, s)$.

We admit something like $(a||b = b||a)$ as a domain-independent axiom in the extended situation calculus. $||$ is considered as a function, and is used in the relaxed infix form. An action term is called atomic action term if there is no appearance of the symbol $||$ in it, otherwise it is called concurrent action term. For example, *Switchon* is an atomic action term while $Switchon||Shoot||Opendoor$ is a concurrent action term. In this paper by *an extended situation calculus (ESC)* we mean a first-order theory defined by its domain-specific axioms which may contain expressions of the form $a_1||a_2||\cdots||a_n$ discussed above. When no confusion may arise, by extended situation calculus we mean any extended situation calculus.

From the above discussion it can be seen that any situation calculus (SC) of [11] is also an extended situation calculus (ESC), i.e. ESC is as expressive as SC. But is SC as expressive as ESC? Our conclusion is that both SC and ESC have the same expressive power. Note that if we ignore the internal structure of the expression $Switchon||Shoot$ and consider it as an atomic one, say $\overline{Switchon||Shoot}$, then the extended situation calculus becomes the same as the non-extended one. This observation can be used to prove the equivalence between ESC and SC. Before we prove the equivalence between ESC and SC we need some preparation. In the following two definitions, SC_1 and SC_2 can be SC or ESC.

Definition 2.1 (Action Mapping) A mapping τ is said to be an action mapping from situation calculus SC_1 to SC_2 iff (i) SC_1 and SC_2 have the same alphabet except for actions; (ii) For any action term a in SC_1, $\tau(a)$ is an action term of SC_2.

Definition 2.2 (Equivalence of two situation calculi) Two situation calculi SC_1 and SC_2 are said to be equivalent iff there is a one-one action mapping τ between them such that for any formula ϕ and ψ in SC_1, $\phi \vdash \psi$ in SC_1 iff $\tau(\phi) \vdash \tau(\psi)$ in SC_2.

Proposition 2.3 The extended situation calculus with concurrent actions (ESC) has the same expressive power as the situation calculus (SC), i.e. for any ESC there is an equivalent SC.

The above proposition means that extending the situation calculus with concurrent actions does not increase its expressive power. In order to reason about change caused by concurrent actions, we need to describe effects of all the possible concurrent actions. The combinations of all the possible concurrent actions are obviously infinite, if there is at least one atomic action. For example, let a be an atomic action in a situation calculus SC, then $a||a$, $a||a||a$, $\cdots$, $a||a||\cdots||a$, $\cdots$ are all concurrent actions[2].

[2]Strictly speaking, expressions of the form $a||a||\cdots||a$ are syntactical *action terms*. Their semantics can be given by mapping them into actions in reality. Some syntactical actions may be meaningless. One can give some formulas as effect descriptions of actions for the intended actions in reality.

3 The composition problem

When the extended situation calculus is applied to a practical universe of discourse, each concurrent action should be considered as being uninterrupted and look like an atomic action, and its effect might be *explicitly* given in the form of $H(\phi, s) \rightarrow H(\psi, R(a_1 || \cdots || a_n, s))$. It is, however, often impossible to do so, since there are infinitely many *well-defined* or *syntactically recognized* concurrent actions. On the other hand, given effect descriptions of atomic actions, we may be able to derive the effects of concurrent actions.

Example 3.1 Suppose there is a banking account CGD No. 000 1000 780 for donations. The atomic action set may be $\{\ donate(n) | n \in \aleph\ \}$. A domain-specific axiom is $H(balance(0), s_0)$. The effects of atomic actions may be described as follows:

$$H(balance(x), s) \rightarrow H(balance(x+d), R(donate(d), s))$$

Note that many people may donate their money at the same time. Assume n donators donate their money $d_1 \cdots d_n$ at the same time, respectively. Then, the simultaneous donations can be represented by $donate(d_1) || \cdots || donate(d_n)$. Let D stand for $donate(d_1) || \cdots || donate(d_n)$, then the effect of $donate(d_1) || \cdots || donate(d_n)$ may be described by the following formula:

$$H(balance(x), s) \rightarrow H(balance(x + d_1 + \cdots + d_n), R(D, s))$$

In the above example it seems that the effect description of $donate(d_1) || \cdots || donate(d_n)$ is *unnecessary* and *redundant*, and thus can be omitted, since there is *some* relationship between its effects and the effects of its component actions $donate(d_1), \cdots, donate(d_n)$.

We can find many other examples in which effects of concurrent actions *may* be derived from effects of their component actions. It is certainly favoured to avoid unnecessary descriptions of effects of concurrent actions, if they can be derived from the effects of their component actions. Now the problem is: how do we avoid these unnecessary effect descriptions, given effect descriptions of atomic actions? If an effect description of concurrent actions has been omitted, how do we derive it later? This problem looks simple, but solutions to it may not be so easy as one imagines. In this paper we call it the composition problem. The composition problem concerns the effect relationship between concurrent actions and their component actions. In reality, when some atomic action is performed, it has certain effects; when it is performed concurrently with other actions, either it may have the same effects as if it were performed atomically, or it may have additional side-effects, or it may have different effects from those it has when it is performed atomically. The following example is about side-effects of concurrent actions.

Example 3.2 In order to encourage its clients to consume, a shop has a special policy: If a client buys shampoo, toothpaste, and toothbrush, then the shop gives a soap to the client free of charge. Let the atomic actions be $\{buy(x) \mid x \in \{shampoo, paste, brush, soap\}\}$. Then, the effects of atomic actions may be described as follows: $H(has(x), R(buy(x), s))$. The effects of the concurrent action $buy(shampoo) || buy(paste) || buy(brush)$ are as follows:

$$H(has(shampoo), R(buy(shampoo) || buy(paste) || buy(brush), s))$$
$$H(has(paste), R(buy(shampoo) || buy(paste) || buy(brush), s))$$
$$H(has(brush), R(buy(shampoo) || buy(paste) || buy(brush), s))$$
$$H(has(soap), R(buy(shampoo) || buy(paste) || buy(brush), s))$$

The last formula above may be regarded as a side-effect of the concurrent action, and the others may be derived from effects of atomic actions.

As a matter of fact, the composition problem also exists in many other formalisms for reasoning about change caused by concurrent actions. If a formalism does not allow for concurrent actions, we can follow the idea in the previous section and extend it so that it can allow for concurrent actions. In the longer version of this paper we considered the composition problem in other two formalisms in other fields: Modal action logic of [10] for software requirements specification and a tableaux system for dynamic entities of [7] based on OBLOG of [19] for conceptual modelling.

4 Defeasible reasoning

In the previous section we identified the composition problem and indicated that this problem exists in many other formalisms for reasoning about concurrent actions. A trivial, but often impractical, solution to the composition problem is just to write down all the axioms describing effects of concurrent actions. In what follows we provide a defeasible solution to the composition problem.

Defeasible reasoning is a type of non-monotonic reasoning which permits "jumping to conclusions" or reaching conclusions which rely in part on the "absence of evidence to the contrary". The typical and early systems for non-monotonic reasoning include circumscription [12], modal non-monotonic logics [13, 14], and default logic [17]. In this paper we make use of the consistency-based default rules of [17] to solve the composition problem. Default rules are of the form $\frac{\phi:M\varphi}{\psi}$, where ϕ, φ, and ψ, called *prerequisite, justification*, and *consequent*, respectively, are first-order sentences and M means "consistent". A default theory is a pair (D, W), where D is a set of default rules of the above form and W is a set of first-order sentences. Given a default theory (D, W), an extension E for (D, W) is defined to be a set of sentences satisfying the following properties: (i) $W \subseteq E$; (ii) $E = Th(E)$, where $Th(E)$ is the deductive closure of E; (iii) For any default $\frac{\phi:M\varphi}{\psi} \in D$, if $\phi \in E$ and $\neg\varphi \notin E$, then $\psi \in E$. A default theory (D, W) defines zero or more extensions, any of which is a representation of a *possible world*.

In what follows, we write $R(a_1; a_2; \cdots; a_n, s)$ to mean $R(a_n, R(\cdots, R(a_2, R(a_1, s))))$. As a tentative solution to the composition problem we propose $(IT2)$[3]:

$$\frac{H(p, R(a;b,s)) \wedge H(p, R(b;a,s)) : MH(p, R(a||b,s))}{H(p, R(a||b,s))}$$

In fact, the above default is an open default. In practice, it should be Skolemized (cf. [17]). The usefulness of the above default rule is best displayed with some examples.

Example 4.1 Consider two simultaneous actions: switching on the light and putting the block A on the block B. Suppose we are given $H(Lightoff, s_0)$ and the usual effect descriptions of actions *Switch* and *Puton*. Then, by using frame axioms[4] and first-order rules of infernece:

$$H(Lighton, R(Switch, s_0)) \qquad H(Lighton, R(Switch; Puton(A,B), s_0))$$
$$H(Lightoff, R(Puton(A,B), s_0)) \qquad H(Lighton, R(Puton(A,B); Switch, s_0))$$

Thus, by the default rule $(IT2)$ we have $H(Lighton, R(Switch||Puton(A,B), s_0))$. On the other hand, we have (where Sw stands for $Switch$)

$$H(On(A,B), R(Puton(A,B), s_0)) \qquad H(On(A,B), R(Puton(A,B); Sw, s_0))$$
$$H(On(A,B), R(Puton(A,B), R(Sw, s_0))) \qquad H(On(A,B), R(Sw; Puton(A,B), s_0))$$

Thus by $(IT2)$ we have $H(On(A,B), R(Switch||Puton(A,B), s_0))$.

[3] IT comes from *Interleaving Traces*. Although interleaving traces are used, we do not imply that concurrent actions are *physically* performed in interleaving orders; they are physically performed *at the same time*.

[4] The frame problem, qualification problem and ramification problem are not addressed in this paper. To deal with frame problem, an extreme approach is to use *frame axioms* [11].

Notice that the inference should be carried out by computing the extension of the default theory. The above inference is relaxed. Later we will still make inference in a relaxed form without formally computing extensions of default theories.

We should emphasize that by using $(IT2)$ we can only omit *unnecessary* effect descriptions of concurrent actions. Sometimes we still need to give the *necessary* descriptions of effects. In the shampoo-brush-paste-soap scenario we still need to give the following formula to describe the *side-effect* of the concurrent action $buy(shampoo)||buy(paste)||buy(brush)$:

$$H(has(soap), R(buy(shampoo)||buy(paste)||buy(brush), s))$$

The other three formulas for $buy(shampoo)||buy(paste)||buy(brush)$ can be omitted.

In general, we need to consider the preconditions. In this case, the above default rule $(IT2)$ can be generalized as follows:

$$\frac{H(q,s) \rightarrow H(p, R(a;b,s)) \wedge H(q,s) \rightarrow H(p, R(b;a,s)) : M(H(q,s) \rightarrow H(p, R(a||b,s)))}{H(q,s) \rightarrow H(p, R(a||b,s))}$$

For concurrent actions composed of more than two atomic actions, the above default rule $(IT2)$ can be easily extended. Let A be a bag[5], we write $Per(A)$ to stand for all the ordered permutations of the elements of the bag A. For example, let $A = \{a,b,c\}$, then $Per(A) = \{(a;b;c),(a;c;b),(c;a;b),(c;b;a),(b;a;c),(b;c;a)\}$. For concurrent action $a_1||\cdots||a_n$, we propose to use the following general default rule, denoted by (ITn):

$$\frac{\Pi(p,A,s) \ : \ MH(p, R(a_1||\cdots||a_n, s))}{H(p, R(a_1||\cdots||a_n, s))}$$

where $A = \{a_1, \cdots, a_n\}$ and $\Pi(p,A,s)$ denotes $\bigwedge_{e \in Per(A)} H(p, R(e,s))$. Like $(IT2)$, the above (ITn) is an open default. In practice, it should be Skolemized. For conditional effects the above default rule (ITn) can further be generalized as follows:

$$\frac{H(q,s) \rightarrow \Pi(p,A,s) : M(H(q,s) \rightarrow H(p, R(a_1||\cdots||a_n, s)))}{H(q,s) \rightarrow H(p, R(a_1||\cdots||a_n, s))}$$

We should point out that rules (ITn) cannot be derived from $(IT2)$ as might be expected. Now it should be clear that reasoning about change caused by concurrent actions amounts to computing extensions of a default theory. A default theory (D, W) defines zero or more extensions, any of which is a representation of a possible world. Now some questions about our extended situation calculus armed with (ITn) are: What about its extensions? May it have no extensions? etc. These questions are concerned with properties of default theories. Recall [17]. A default is said to be normal iff it is of the form $\frac{\phi : M\psi}{\psi}$. Thus, our default rule (ITn) is actually a normal default. By [17] every normal default theory has an extension. Thus we can immediately have the following proposition:

Proposition 4.2 An ESC armed with the default rule (ITn) always has an extension.

5 Examples

In this section we give three representative examples to show the usefulness of the above default rule. The three examples represent the following three cases: (i) The default rule always gives the correct and expected result, and conclusions from the default rule cannot be defeated; (ii) Conclusions from the default rule may be defeated by new knowledge; (iii) The default rule cannot be used to produce new beliefs. In the following we use very relaxed default proofs to avoid formal computation of extensions for default theories.

[5]A bag may contain multiple appearances of the same object, whereas a set always contain one and only one appearance of an object. A bag is sometimes called multi-set. In this paper we sometimes abuse *set* and its associated notations for *bag*.

Example 5.1 Consider a chocolate slot machine which has two coin entries. Suppose there are 100 units of money inside the machine, that is, $H(In(100), s_0)$. Now insert coins into the two different entries at the same time. Suppose one unit of money is inserted into the first entry and two units into the second entry. Then how much money is there inside the machine? Now we have the following inference:

$$H(In(101), R(insert(1), s_0)) \qquad H(In(103), R(insert(1); insert(2), s_0))$$
$$H(In(102), R(insert(2), s_0)) \qquad H(In(103), R(insert(2); insert(1), s_0))$$

Thus, by the default rule $(IT2)$ we have $H(In(103), R(insert(2)||insert(1), s_0))$. In reality, it is really the case. The conclusion from $(IT2)$ cannot be defeated[6]. In this example the following composition axiom is unnecessary and can thus be omitted:

$$H(In(x), s) \rightarrow H(In(x + n_1 + n_2), R(insert(n_1)||insert(n_2), s))$$

Example 5.2 Consider two simultaneous actions: raining and using an umbrella. Assume that if it rains and you do not use an umbrella, you will be wet. If you are wet, you will still be wet after using an umbrella. If you are dry, you will still be dry after using an umbrella. Suppose you are dry initially, that is, $H(Dry, s_0)$. We have the following three formulas as effect descriptions of atomic actions *rains* and *umbrella*:

$$H(Wet, R(rains, s))$$
$$H(Wet, s) \rightarrow H(Wet, R(umbrella, s)) \qquad H(Dry, s) \rightarrow H(Dry, R(umbrella, s))$$

Now we have the following inference:

$$H(Wet, R(rains, s_0)) \qquad H(Wet, R(rains; umbrella, s_0))$$
$$H(Dry, R(umbrella, s_0)) \qquad H(Wet, R(umbrella; rains, s_0))$$

Thus, by the default rule $(IT2)$ we have $H(Wet, R(rains||umbrella, s_0))$. If, however, we have the additional knowledge "if you use an umbrella while it rains, you will be as dry as before", that is, we know the effects of the concurrent actions "raining and using an umbrella":

$$H(Dry, s) \rightarrow H(Dry, R(rains||umbrella, s))$$
$$H(Wet, s) \rightarrow H(Wet, R(rains||umbrella, s))$$

Then we can never have the above conclusion $H(Wet, R(rains||umbrella, s_0))$, which is defeated by new additional knowledge.

Example 5.3 Consider two simultaneous actions: Taking out all the money in the wallet and putting 10 dollars into the wallet. Suppose there are 50 dollars in the wallet in the initial situation, i.e., $H(In(50), s_0)$. Then how much money is left in the wallet?

In this example, we have the following formulas as descriptions of effects of atomic actions $Putin(n)$ and $TakeAll$:

$$H(0, R(TakeAll, s))$$
$$H(In(x), s) \rightarrow H(In(x + n), R(Putin(n), s))$$

Then we have the following inference:

$$H(In(0), R(TakeAll, s_0)) \qquad H(In(10), R(TakeAll; Putin(10), s_0))$$
$$H(In(60), R(Putin(10), s_0)) \qquad H(In(0), R(Putin(10); TakeAll, s_0))$$

Since we don't have $H(In(0), R(TakeAll; Putin(10), s_0))$, we cannot apply $(IT2)$ to achieve $H(In(0), R(TakeAll||Putin(10), s_0))$. Analogously, neither can we apply $(IT2)$ to achieve $H(In(10), R(TakeAll||Putin(10), s_0))$.

[6] One may find an exceptional slot machine where the above conclusion can also be defeated, which is concerned with another problem in reasoning about change: the qualification problem. For example, a coin may be rejected by the machine.

6 Atomicity of actions

It is sometimes the case that an atomic action is later refined into *smaller* ones. That is to say, actions which were initially taken to be atomic may be later regarded as composed of more primitive actions. For example, the action $Puton(A,B)$ may be regarded as a compound action $(Clear(A); Clear(B); Moveon(A,B))$, where $Clear(x)$ designates the action of clearing the object x and $Moveon(x,y)$ designates the action of moving x onto top of y. The effect of $Switchon||Puton(A,B)$ is thus that of $Switchon||(Clear(A); Clear(B); Moveon(A,B))$. In general, to assume that a is $(a_1; a_2; \cdots; a_m)$ and b is $(b_1; b_2; \cdots; b_n)$ for $m, n \geq 1$, how should we evaluate the effect of $a||b$ in terms of the effects of a_i, b_j for $1 \leq i \leq m$ and $1 \leq j \leq n$?

Before proceeding on, we define the ordered interleaving traces as follows. First we define a projection operation π of sequential actions on sequential actions. Let $c = (c_1; c_2; \cdots; c_k)$, $k \geq 1$, be any sequential action and d an atomic action, then the projection of sequential action (d;x) on c is defined as follows:

$$\pi((d;x),c) = \begin{cases} x & \text{if } d \neq c_i \text{ for any } 1 \leq i \leq k \\ (d; \pi(x,c)) & \text{if } d = c_i \text{ for some } 1 \leq i \leq k \end{cases}$$

We define the ordered interleaving traces $OPer(a,b)$ of a and b as follows:

$$OPer(a,b) = \{x \mid x \in Q \text{ and } \pi(x,a) = a \text{ and } \pi(x,b) = b\}$$

where $Q = Per(\{a_i \mid 1 \leq i \leq m\} \cup \{b_j \mid 1 \leq j \leq n\})$. It is easy to extend the domain of $OPer$ for any finite number of sequential actions. For example, let a, b, c be three sequential actions. Then, we can define

$$OPer(a,b,c) = \{x \mid x \in Q \text{ and } \pi(x,a) = a \text{ and } \pi(x,b) = b \text{ and } \pi(x,c) = c\}$$

where Q is set of all the interleaving traces of atomic actions appearing in a, b, c.

By using ordered interleaving traces, for concurrent actions $a_1|| \cdots ||a_n$, each of which may be a sequential action composed of more primitive actions, the default rule (ITn) is modified as follows:

$$\frac{\Pi^o(p,A,s) \;:\; MH(p, R(a_1||\cdots||a_n, s))}{H(p, R(a_1||\cdots||a_n, s))}$$

where $A = \{a_1, \cdots, a_n\}$ and $\Pi^o(p,A,s) = \bigwedge_{e \in OPer(A)} H(p, R(e,s))$. It is easy to see that $\Pi(p,A,s)$ is a special case of $\Pi^o(p,A,s)$ when each element of A is itself an atomic one. Thus, the original default rule (ITn) is a special case of the modified default rule (ITn). Let a be $a_1; a_2; \cdots; a_m$ and let b be $b_1; b_2; \cdots; b_n$. Substituting a and b in the original $(IT2)$ with $a_1; a_2; \cdots; a_m$ and $b_1; b_2; \cdots; b_n$, respectively, then we have a default rule. That is to say, the original $(IT2)$ can also be applied to actions composed of more primitive actions. However, its deductive power is different from that of the modified $(IT2)$, as indicated by the following proposition.

Proposition 6.1 Assume an extended situation calculus C is consistent. Let W be all the theorems of C. Let D_1 be all the Skolemized default rules of the original (ITn), and D_2 all the Skolemized default rules of the modified (ITn). Then, for any extension E_2 for the default theory (D_2, W) there is an extension E_1 for the default theory (D_1, W) such that $E_2 \subseteq E_1$.

This proposition can be easily proved by use of Proposition 7.3 in the next section.

7 Believability

As said before, conclusions from default rules should be understood as beliefs, which can be defeated by new facts. In previous sections we have proposed some default rules for reasoning about concurrent actions. In particular, we have modified the default rules (ITn) to deal with actions which were initially taken as atomic and are later composed of more primitive actions. The modified default rules (ITn) imply that beliefs from the original default rules (ITn) should not be believed if they do not follow from the modified default rules (ITn). That is to say, conclusions from the modified default rules (ITn) are *more believable* than those from the original default rules (ITn). Note that the prerequisites of all the (original or modified) default rules (ITn) are conjunctions of some formulas of the form $H(p, R(e, s))$, where $e \in Per(A)$ in the original defaults (ITn) or $e \in OPer(A)$ in the modified defaults (ITn). More credulous people may think that the prerequisites might be too strong, and thus the conclusions might be too conservative. A simple modification of the prerequisites in (ITn) is just to change conjunction into disjunction. That is to say, for concurrent action $a_1 \| \cdots \| a_n$, we may use the following default rules:

$$\frac{\Sigma(p, A, s) \;:\; MH(p, R(a_1 \| \cdots \| a_n, s))}{H(p, R(a_1 \| \cdots \| a_n, s))}$$

where $A = \{a_1, \cdots, a_n\}$ and $\Sigma(p, A, s) = \bigvee_{e \in Per(A)} H(p, R(e, s))$. The beliefs from the above defaults are less believable (or more credulous) than those from the original defaults.

From the above simple discussion it can be seen that we can define a partial ordering relation among default theories to indicate some beliefs are more believable than others. In the following we start with the general definition of believability, then discuss our extended situation calculus.

Definition 7.1 Let $D = \{D_i \mid D_i$ is a set of defaults for any $i \geq 1\}$ be a (finite or infinite) family of sets of defaults. We define a partial ordering relation $\preceq$ on D as follows: $D_i \preceq D_j$ iff for every extension E_i for (W, D_i) there is an extension E_j for (W, D_j) such that $E_i \subseteq E_j$, where W is any set of formulas. The partial ordering relation $\preceq$ is called more-believable-than relation, or simply believability relation.

Obviously, if $D_i \preceq D_j$, then beliefs from (W, D_i) are less defeasible than those from (W, D_j). In particular, $\{\} \preceq D_j$ for any D_j. Extensions for $(W, \{\})$ are just $Th(W)$, which cannot be defeated by any new facts. Hence, $Th(W)$ is the most believable.

Lemma 7.2 Let D_1 and D_2 be two sets of normal defaults. If $D_1 \subseteq D_2$, then, $D_1 \preceq D_2$.

The above lemma says that the more normal defaults are used, the less believable the conclusions from them. The empty set of normal defaults is the most believable. The following proposition is particularly useful for the following discussion.

Proposition 7.3 Let D_1 and D_2 be two sets of closed normal defaults. Suppose for any normal default $\{\frac{\phi : M\psi}{\psi}\} \in D_1$ there is a normal default $\{\frac{\varphi : M\psi}{\psi}\} \in D_2$ such that $\vdash \phi \to \varphi$. Then, $D_1 \preceq D_2$.

Let C be an extended situation calculus. Then, by admitting different set of default rules we have different default theories. Some of them are more believable than others. Given a concurrent action $a_1 \| \cdots \| a_n$, let $\Upsilon \subseteq Per(\{a_1, \cdots, a_n\})$ be any subset of ordered permutations of the set $\{a_1, \cdots, a_n\}$. We generalize the denotation Σ and write $\Sigma(p, \Upsilon, s)$ to stand for $\bigvee_{\varsigma \in \Upsilon} H(p, R(\varsigma, s))$. Let $\Upsilon_1, \cdots, \Upsilon_k \subseteq Per(\{a_1, \cdots, a_n\})$ be any k subsets of ordered permutations of the set $\{a_1, \cdots, a_n\}$. Then, we may use the following default rule for $a_1 \| \cdots \| a_n$:

$$\frac{\bigwedge_{1 \leq i \leq k} \Sigma(p, \Upsilon_i, s) \;:\; MH(p, R(a_1 \| \cdots \| a_n, s))}{H(p, R(a_1 \| \cdots \| a_n, s))}$$

Given different subsets $\Upsilon_1, \cdots, \Upsilon_k$ of $Per(\{a_1, \cdots, a_n\})$, we have an instance of the above default. It can be verified that (ITn) and the default in the beginning of this section are both special instances of the above default. Let $\varsigma_1, \varsigma_2, \cdots, \varsigma_{n!}$ be all the elements of $Per(\{a_1, \cdots, a_n\})$, then among all the instances of the above default, the following $n!+2$ defaults $\delta_\forall, \delta_{\varsigma_1}, \cdots, \delta_{\varsigma_{n!}}, \delta_\exists$ are particularly interesting:

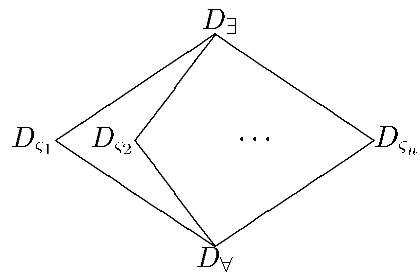
$$\delta_\forall : \quad \frac{\bigwedge_{\varsigma \in Per(\{a_1,\cdots,a_n\})} H(p,R(\varsigma,s)) : MH(p,R(a_1||\cdots||a_n,s))}{H(p,R(a_1||\cdots||a_n,s))}$$

$$\delta_{\varsigma_1} : \quad \frac{H(p,R(\varsigma_1,s)) : MH(p,R(a_1||\cdots||a_n,s))}{H(p,R(a_1||\cdots||a_n,s))}$$

$$\cdots$$

$$\delta_{\varsigma_{n!}} : \quad \frac{H(p,R(\varsigma_{n!},s)) : MH(p,R(a_1||\cdots||a_n,s))}{H(p,R(a_1||\cdots||a_n,s))}$$

$$\delta_\exists : \quad \frac{\bigvee_{\varsigma \in Per(\{a_1,\cdots,a_n\})} H(p,R(\varsigma,s)) : MH(p,R(a_1||\cdots||a_n,s))}{H(p,R(a_1||\cdots||a_n,s))}$$



Using the believability relation, we have a partial ordering among $D_\forall = \{\delta_\forall\}$, $D_\exists = \{\delta_\exists\}$, $D_{\varsigma_i} = \{\delta_{\varsigma_i}\}$ for $i = 1, 2, ..., n!$, which is illustrated by the above flat lattice.

Similarly, for the concurrent actions, each of which may be sequentially composed of other more primitive actions, substituting $OPer$ for Per in the above we can have many choices of defaults, among which a flat lattice may be particularly interesting. A short example may be found in the extended version of this paper. Since some beliefs may be more believable than others, we may wonder whether a more believable default theory is preferred to another less believable default theory. Actually, the answer depends on the understanding of *preferred* and practical needs in applications. For the space limitation we will not go into deeper discussions.

8 Comparison and discussion

We have extended the situation calculus of [11] with concurrent actions, identified the composition problem in formalisms for reasoning about concurrent actions, and proposed some normal default rules to solve the composition problem. Reasoning about concurrent actions is thus reduced to computation of extensions for normal default theories. After the work of this paper had been finished, we found some other attempts at dealing with concurrent actions in the situation calculus, namely, Lin and Shoham [9] and Baral and Gelfond [1], both of which are in line with Gelfond, Lifschitz and Rabinov's pioneering work [4]. Although there are some similarities between our work and theirs (the exact and precise relationship is yet to be investigated), some technical differences also exist. In [1], a language denoted by $\mathcal{A}_C$ and a translation methodology of theories from $\mathcal{A}_C$ to extended logic programs ELPs of [3] are proposed. In the translation, the following axioms (see [1] for details), which are also among major contributions of [1], play a vital rôle:

$$\begin{array}{lll}
(a) & H(f, R(a, s)) & \leftarrow subsetof(b, a), H(f, R(b, s)), not\ Noninherit(f, a, b, s) \\
(b) & \neg H(f, R(a, s)) & \leftarrow subsetof(b, a), \neg H(f, R(b, s)), not\ Noninherit(\overline{f}, a, b, s) \\
(c) & Noninherit(\overline{F}, x, y, s) & \leftarrow subsetof(y, x), subsetof(A, x), \neg subsetof(A, y), \\
 & & \quad not\ \overline{H(P_1, s)}, \cdots, not\ \overline{H(P_n, s)}
\end{array}$$

According to these axioms, concurrent actions normally inherit their component actions' effects. This is the case when subactions are independent of each other. For example, the effect of $Switchon||Opendoor$ is the *net aggregation* of $Switchon$ and $Opendoor$ effects. When subactions are independent of each other, both Baral and Gelfond [1] and our default rules can achieve the same conclusions. When subactions are dependent on each other, some effect descriptions of concurrent actions can be omitted by use of our default rules and some cannot, as shown before. In the chocolate slot machines example, however, it is not clear whether and how to use the axioms to derive effects of $insert(n_1)||insert(n_2)$ from effects of $insert(x)$. It seems that the default rules (ITn) are more expressive than the above axioms.

In both [9] and [1] a concurrent action is defined to be a finite set of actions. Thus, it is not possible in [9] and [1] to distinguish between actions and occurrences of actions. For example, our formalism takes $Buy(sweepstake)||Buy(sweepstake)$ as a concurrent action, where $Buy(sweepstake)$ should be understood as an *occurrence* of action $Buy(sweepstake)$. In [9] and [1] it is not possible to express two concurrent occurrences of the same action $Buy(sweepstake)$ with the help of $\{Buy(sweepstake), Buy(sweepstake)\}$, since it is regarded as the same as $\{Buy(sweepstake)\}$ by set theory. This problem could be simply solved by adding an extra parameter to all the atomic actions to denote the number of occurrences of the action. However, it is not clear how to express $(a;b)||(a;b)$ if we allow sequential actions of the form $a;b$.

The focus of Lin and Shoham [9] is on the epistemological completeness proposed by themselves in [8]. Intuitively, a theory of an action is epistemologically complete if, given a complete description of the initial situation, the theory enables us to predict a complete description of the resulting situation when the action is performed. Lin and Shoham use some circumscriptive minimization techniques to deal with their generalized frame problem, which looks similar to our composition problem (a study of the precise relationship is still needed). Lin and Shoham's formalism enjoys the epistemological completeness. It seems that both our formalism and Baral and Gelfond's formalism [1] do not have precisely this property. It is because of epistemologically completeness that $H(Open, R(Opendoor||Closedoor, s))$, for example, could be derivable in Lin and Shoham's formalism by using inertia, but it cannot be derived in either our formalism or Baral and Gelfond's: the effect of $Opendoor||Closedoor$ is unknown in our formalism and Baral and Gelfond's. Of course, if one really wants something like this in our formalism, he may just assert:

$$H(Open, s) \rightarrow H(Open, R(Opendoor||Closedoor, s))$$
$$H(Closed, s) \rightarrow H(Closed, R(Opendoor||Closedoor, s))$$

The main point of this paper is the composition problem and the proposed defeasible solution. We may have many defeasible solutions, some of which are more believable (or skeptical) than others. All the solutions may be compared by a lattice given by the believability relation. There might be other solutions to the composition problem. We could benefit from the research in verification of concurrent programs in the field of Software Engineering. For example, Lamport and Owick's proof rule for parallel statement composed of interference-free statements [5, 15] may be applied. It seems that there is a very close relationship between Lamport and Owick's proof rule and Baral and Gelfond's axioms. But the precise relationship is yet to be investigated. We should emphasize that there is a difference between verification of concurrent programs and reasoning about concurrent actions in Artificial Intelligence. In verification of concurrent programs, the problems such as synchronization, message-passing and resource-sharing need to be considered. The verification of concurrent programs cannot be non-monotonic. Everything there has to be exact and correct. The concept of *validity* plays the central rôle. If the correctness of concurrent programs were not absolutely guaranteed, the programs would be useless. In common-sense reasoning studied in Artificial Intelligence, however, conclusions might be defeated after new knowledge is learnt. In common-sense reasoning the concept of *consistency*, instead of validity, plays the central rôle. All the knowledge and beliefs are to be consistent.

The result of this work has been adapted and used in an abductive planner [6] based on extended logic programs with explicit negation. Currently we are working on temporal knowledge representation and belief revision in a variant of extended situation calculus by use of logic programs with well-founded semantics with explicit negation [2, 16].

Acknowledgements

We thank Esprit Project Compulog 2 (No. 6810) and JNICT - Portugal for their support. We would also like to thank António Porto, Carlos Damásio, and Michael Schroeder for helpful discussions.

References

[1] Baral, C. and Gelfond, M., Representing concurrent actions in extended logic programming, *IJCAI*, 1993, pp. 866-871

[2] Damásio, C. V., Wolfgang, N. and Pereira, L.M., REVISE: An Extended Logic Programming System for Revising Knowledge Bases, KR'94.

[3] Gelfond, M. and Lifschitz, V., Logic programs with classical negation, In. *Proc. of 7th ICLP*, D. Warren and P. Szeredi, eds., 1990, pp. 579-597

[4] Gelfond, M., Lifschitz, V., and Rabinov, A., What are the limitaions of the situation calculus?, Automated Reasoning: Essays in Honor of Woody Bledsoe, R. Moore (ed.), 1991, 167-179

[5] Lamport, L., Proving the correctness of multiprocess programs, *IEEE Trans. on Software Engeneering*, SE-3, 2, 1977, pp.125-143

[6] Li, R. and Pereira, L.M., Abductively Make a Good Plan with Extended Logic Programming, Technical Report, AI Center, UNINOVA, 1994

[7] Li, R. and Sernadas, A., Reasoning about objects using a tableau method, *J. of Logic and Computation*, 1:5, 1991, pp. 575-611

[8] Lin, F. and Shoham, Y., Provably correct theories of actions: preliminary report, *AAAI*, 1991

[9] Lin, F. and Shoham, Y., Concurrent actions in the situation calculus, *Proc. of AAAI-92*, 1992

[10] Maibaum, T.S.E., A logic for the formal requirements specificaiton of real-time embedded systems, Dept. of Computing, Imperial College, 1987

[11] McCarthy, J. and Hayes, P.J., Some philosophical problems from the standpoint of artificial intelligence, in: B. Meltzer and D. Michie, (eds.), *Machine Intelligence*, 4, 1969, pp.463-502

[12] McCarthy, J., Circumscription – A form of non-monotonic reasoning, *Artificial Intelligence*, 13, 1980, pp. 27-39

[13] McDermott, D. and Doyle, J., Non-monotonic logic I, *Artificial Intelligence*, 13, 1980, pp. 41-72

[14] Moore, R., Semantical considerations on nonmonotonic logic, *Artificial Intelligence*, 25, 1985

[15] Owick, S. and Gries, D., Verifying properties of parallel programs: An axiomatic approach, *Comm. ACM*, 19:5, 1976, pp. 279-285

[16] Pereira, L. M., and Alferes, J. J., Well-founded semantics for logic programs with explicit negation, in *ECAI'92*, B. Neumann (ed.), John Wiley & Sons, 1992, 102-106

[17] Reiter, R., A logic for Default Reasoning, *Artificial Intelligence*, 13, 1980, pp. 81-132

[18] Reiter, R., Proving properties of states in the situation calculus, *Artificial Intelligence*, 64:2, 1993, pp. 337-351

[19] Sernadas, A., Fiadeiro, J., Sernadas, C., Ehrich, H.-D., Abstract object types: a temporal perspective , in: *Temporal Logic in Specification*, B. Banieqbal et al. (eds.), Springer-Verlag

[20] Shoham, Y., Ten requirements for a theory of change, *New Generation Computing*, 3, 1985

[21] Shoham, Y. and N. Goyal, Temporal reasoning in Artificial Intelligence, in: *Exploring Artificial Intelligence*, H. E. Shrobe (ed.), Morgan Kaufmann Publishers, Inc. , 1988