A General Framework for Expressing Preferences in Causal Reasoning and Planning

James P. Delgrande
School of Computing Science
Simon Fraser University
Burnaby, B.C.
Canada V5A 1S6
jim@cs.sfu.ca

Torsten Schaub*
Institut für Informatik
Universität Potsdam
Postfach 90 03 27
D–14439 Potsdam, Germany
torsten@cs.uni-potsdam.de

Hans Tompits
Institut für Informationssysteme
Technische Universität Wien
Favoritenstraße 9–11
A–1040 Vienna, Austria
tompits@kr.tuwien.ac.at

Abstract

We consider the problem of incorporating arbitrary preferences in planning systems. A preference may be seen as a goal or constraint that is desirable, but not necessary, to satisfy. We work within the context of transition systems; however, our results are applicable to general planning formalisms. To begin, we define a query language for specifying arbitrary conditions that may be satisfied by a history, or interleaved sequence of world states and actions. Given this, we specify a second language in which preferences are defined. A single preference defines a binary relation on histories, so that in an ordered pair of histories the second history is preferred to the first. From this, one can define global preference orderings on the set of histories, the maximal elements of which are the preferred histories. The approach is very general and flexible; thus it constitutes a "base" language in terms of which higher-level operators may be defined. The approach can be used to express others, and so serves as a common basis in which such approaches can be expressed and compared.

1 Introduction

Planning, as traditionally formulated, involves attaining a particular goal, given an initial state of the world and a description of actions. A plan succeeds just when it is executable and attains the goal; otherwise it fails. However, in realistic situations, things are not quite so simple. Thus, there may be requirements specifying that a plan should be as short as possible or that total cost, where costs are associated with actions, be minimised.

As well, there may be *preferred* conditions, that are desirable to attain, but not necessary. For an example, consider an extension of the monkeys and bananas problem. In addition to the usual information, where the monkey can push a box and climb on the box to grasp some bananas, consider where the monkey has a range of choices for food. Perhaps the monkey prefers to have an appetizer before the main course, although this preference need not be satisfied in attaining the overall goal of eating a full meal. Perhaps too the monkey prefers soup to grubs as an appetizer, although either will do. If the monkey has soup, then it prefers to have a spoon before it has the soup; and if it does not have a spoon before having soup, then it prefers to have a spoon as soon as possible after getting the soup. Clearly, such preferences can be arbitrarily complex, and range over temporal constraints as well as relations among fluents and actions. In this setting, the goal of a planning problem now shifts to determining a *preferred* plan, in which a maximal set of preferences is satisfied, along with the goal.

Such preferences also make sense outside of planning domains, and in fact apply to arbitrary sequences of temporal events. Hence it is perfectly rational to prefer that it rains during the next several work days (since the plants need the water) but that it be sunny for the weekend.

In this paper, we consider the problem of incorporating general preferences in temporal histories. While we focus on histories as they are used in action description languages [Gelfond and Lifschitz, 1998], our approach is readily applicable to any planning formalism. We begin by specifying a query language in which one can determine if an arbitrary expression is true in a given history. Given this language, we define a *preference-specification language* that enables the definition of preference relations between histories. We obtain a powerful means of specifying preferences in temporal, causal, or planning frameworks. As well, the approach provides a very general language in which other "higher-level" constructs can be encoded, and in which other approaches can be expressed and so compared.

2 Background

Reasoning with preferences is an active area that is receiving increasing attention in AI. The literature is extensive; we mention only Oztürk *et al.* [2005] as an introduction to preference modelling in general, and a recent special issue of *Computational Intelligence* [CI, 2004]

*Affiliated with the School of Computing Science at Simon Fraser University, Burnaby, Canada.

for a selection of papers addressing preferences in Artificial Intelligence and constraints solving.

Our interests lie with preferences in planning (and more generally, temporal) formalisms. Here, research is more recent. One approach to preferences in planning is given by Son and Pontelli [2004], where a language for specifying preferences between histories is presented. This language is an extension of action language $\mathcal{B}$ [Gelfond and Lifschitz, 1998]. The notion of preference explored is based on so-called *desires*, expressed via formulas built by means of propositional as well as temporal connectives such as *always*, *until*, etc. From desires, preferences among histories are induced as follows: Given a desire ϕ, a history H is preferred to H' if $H \models \phi$ but $H' \not\models \phi$.

Eiter *et al.* [2003] describe planning in an answer-set programming framework where action costs are taken into account. The approach allows the specification of desiderata such as computing the shortest plan, or the cheapest plan, or some combination of these criteria. This is achieved by employing weak constraints, which filter answer sets, and thus of plans, based on *quantitative* criteria.

Delgrande *et al.* [2004] propose two types of domain-specific preferences, *choice* and *temporal* preferences. For a choice preference $\psi <_c \phi$, involving fluent and action formulas ϕ and ψ, a history in which ϕ is true at some point is preferred over another in which this is not the case but ψ is true at some point. For a temporal preferences $\psi <_t \phi$, a history in which ϕ becomes true after ψ is preferred to one where this is not the case. While these types of preferences are perhaps compelling, it is clear, as discussed in the introduction, that preferences can be arbitrarily more complex.

The goal of this research is an approach for addressing preferences in temporal and planning settings. A prerequisite to the success of this endeavour is the development of languages for combining preference relations; for work to this end, see for example, Brewka [2004].

3 The Approach

Our central notion is that of a *preference framework*, consisting of a pair $(\boldsymbol{H}, \boldsymbol{P})$, where $\boldsymbol{H}$ is a set of *histories* and $\boldsymbol{P}$ is a set of *preferences* on histories. A history is a sequence of states and transitions between states, representing some evolution of the world. A preference specifies an individual criterion for distinguishing among histories. It defines a binary relation, consisting of pairs of histories where one is preferred to the other, according to the preference. The goal is to determine, in a sense to be established, the most preferred histories according to the set of preferences.

We use the syntax that a history $H \in \boldsymbol{H}$ is a sequence $(s_0, a_1, s_1, a_2, s_2, \ldots, s_{n-1}, a_n, s_n)$, where s_0 is an initial state, and the subsequence s_i, a_{i+1}, s_{i+1} indicates that action a_{i+1} takes the world from state s_i to s_{i+1}. This notation is for convenience only; we could as well have based our approach on, for example, *situations* [Levesque *et al.*, 1998] or any other notation that carries the same information. $\boldsymbol{H}$ can be equated with a complete description of a planning problem, with members of $\boldsymbol{H}$ corresponding to complete plans, but it need not.

We define a preference among two histories directly in terms of a formula ϕ. That is, we define that H_h is not less preferred than H_l, $H_l \preceq_\phi H_h$, just if $\langle H_l, H_h \rangle \models \phi$. The intent is that ϕ expresses a preference condition between two histories, and $H_l \preceq_\phi H_h$ holds if ϕ is true by evaluating it with respect to H_l and H_h. This in turn requires that we are able to refer to fluent and action names at specific time points in histories, and specify their truth values at time points and in histories. Preferences are expressed by means of a formula composed of

- Boolean combinations of fluents and actions indexed by time points in a history, and by a history, and
- quantifications over time points.

Indexing with respect to time points and histories is achieved via *labelled atoms* of form $\ell : b(i)$. Here, ℓ is a *label*, either $\boldsymbol{l}$ or $\boldsymbol{h}$, referring to a history which is considered to be lower or higher ranked, respectively; b is an action or fluent name; and i is a time point. Semantically, $\langle H_l, H_h \rangle \models \boldsymbol{l} : b(i)$ holds if b holds at time point i in history H_l; and analogously for $\langle H_l, H_h \rangle \models \boldsymbol{h} : b(i)$. This is extended to labelled formulas in the expected fashion.

For example, we would express that history H_h is preferred to history H_l if fluent f is true at some point in H_h but never true in H_l by the formula

$$\phi = (\boldsymbol{h} : \exists i f(i)) \ \wedge \ (\boldsymbol{l} : \forall i \neg f(i)), \qquad (1)$$

providing $\langle H_l, H_h \rangle \models \phi$ holds.

We remark that labels, as employed here, serve a similar purpose as labels used in tableau systems [Fitting, 1990] or in labelled deduction [Gabbay, 1996].

Each preference $\phi \in \boldsymbol{P}$ induces a binary relation $\preceq_\phi$ on $\boldsymbol{H}$. This binary relation of course has no properties (such as transitivity) since any such properties will depend on the formula ϕ. Depending on the type of preference encoded in $\boldsymbol{P}$, one would supply a strategy from which a maximally preferred history is selected. Thus for preferences only of the form (1), indicating which fluents are desirable, the maximally preferred history might be the one which was ranked as "preferred" by the greatest number of preferences in $\boldsymbol{P}$.

4 Expressing Preferences on Histories

4.1 Histories and Queries on Histories

In specifying histories, we begin with notation adapted from Gelfond and Lifschitz [1998] in their discussion of *transition systems*. As described, virtually any general planning system (or indeed causal-reasoning formalism) could be used to provide a setting for our approach; as well, the approach is more broadly applicable than just to planning problems.

Definition 1 *An* action signature Σ *is a triple* $\langle V, F, A\rangle$, *where* V *is a set of* value names, F *is a set of* fluent names, *and* A *is a set of* action *names.*

If $V = \{1, 0\}$, *then* Σ *is called* propositional. *If* V, F, *and* A *are finite, then* Σ *is called* finite.

For simplicity we will assume throughout that action signatures are finite and propositional.

Definition 2 *Let* $\Sigma = \langle V, F, A\rangle$ *be an action signature.*

A history, H, *over* Σ *is a sequence*

$$(s_0, a_1, s_1, a_2, s_2, \ldots, s_{n-1}, a_n, s_n),$$

where

- $n \geq 0$,
- *each* s_i, $0 \leq i \leq n$, *is a mapping assigning each fluent* $f \in F$ *a value* $v \in V$, *and*
- $a_1, \ldots, a_n \in A$.

The functions $s_0, \ldots, s_n$ *are called* states, *and* n *is the* length *of history* H, *symbolically* $|H|$.

The states of a history may be thought of as possible worlds, and the actions take one possible world into another. For a propositional action signature $\Sigma = \langle V, F, A\rangle$, fluent $f \in F$ is said to be *true at* state s iff $s(f) = 1$, otherwise f is *false at* s.

We need to be able to refer to fluent and action names in a history. We also need to be able to refer to time points and their relations, as well as to the truth value of a fluent at a time point. Further, we need to be able to refer to histories, since fluents at the same time point may have different values depending on which history is under consideration. For simplicity, we first define a query language on histories of maximum length n over an action signature Σ, named $\mathcal{Q}_{\Sigma,n}$. In the next subsection, we extend this language to deal with formulas that refer to more than one history.

Definition 3 *Let* $\Sigma = \langle V, F, A\rangle$ *be an action signature and* $n \geq 0$ *a natural number. We define the query language* $\mathcal{Q}_{\Sigma,n}$ *as follows:*

1. *The alphabet of* $\mathcal{Q}_{\Sigma,n}$ *consists of*
   - (a) *a set* $\mathcal{V}$ *of* time-stamp variables, *or simply* variables,
   - (b) *the set* $\{0, \ldots, n\}$ *of natural numbers,*
   - (c) *the primitive sentential connectives '*$\neg$*' and '*$\supset$*',*
   - (d) *the primitive quantifier symbol '*$\exists$*',*
   - (e) *the set* $A \cup F$ *of action and fluent names,*
   - (f) *the arithmetic function symbols '*$+$*' and '*$\cdot$*',*
   - (g) *the primitive arithmetic relation symbol '*$<$*',*
   - (h) *the equality symbol '*$=$*', and*
   - (i) *the parentheses '(' and ')'.*
2. *A* time term *is an arithmetic term recursively built from variables and numbers in* $\mathcal{V} \cup \{0, \ldots, n\}$, *employing* $+$ *and* $\cdot$ *(as well as parentheses) in the usual manner. A* time atom *is an arithmetic expression of the form* $(t_1 < t_2)$ *or* $(t_1 = t_2)$, *where* t_1, t_2 *are time terms. We use* TT_n *and* TA_n *to refer to the set of time terms and time atoms, respectively.*
3. *An* atom *is either a time atom, or an expression of form* $b(t)$, *where* $b \in A \cup F$ *and* $t \in TT_n$. *If* b *is an action name, then* $b(t)$ *is called an* action atom, *and if* b *is a fluent name, then* $b(t)$ *is called a* fluent atom. *An atom containing no variable is* ground.
4. *A* literal *is an atom possibly preceded by the sign* $\neg$.
5. *A* formula *is a Boolean combination of atoms, along with quantifier expressions of form* $\exists v$, *for* $v \in \mathcal{V}$, *formed in the usual recursive fashion.*
6. *A* query *is a closed formula, i.e., containing no free time-stamp variables.*

We define the operators $\wedge$, $\vee$, and $\leq$, and the universal quantifier $\forall$, in the usual way. We allow to drop parentheses in formulas if no ambiguity arises, and we may write quantified formulas like $\mathsf{Q}v_1\mathsf{Q}v_2\,\alpha$ as $\mathsf{Q}v_1, v_2\,\alpha$, for $\mathsf{Q} \in \{\forall, \exists\}$. For formula α, variables $v_1, \ldots, v_k$, and numbers $i_1, \ldots, i_k$, $\alpha[v_1/i_1, \ldots, v_k/i_k]$ is the result of uniformly substituting v_j by i_j in α, for each $j \in \{1, \ldots, k\}$. Thus, if $v_1, \ldots, v_k$ are the free variables in α, then $\alpha[v_1/i_1, \ldots, v_k/i_k]$ is a closed formula. For ground time term t, $val(t)$ is the value of t according to standard integer arithmetic.

Variables range over time points, and so quantification applies to time points only. Atoms consist of actions or fluents indexed by a time point, or of a predicate on arithmetic (time point) expressions. Atoms are used to compose formulas in the standard fashion, and queries consist of closed formulas. This means that we remain within the realm of propositional logic, since quantified expressions $\forall v$ and $\exists v$ can be replaced by the conjunction or disjunction (respectively) of their instances.

As an example, let $pickup \in A$, $red \in F$, and $i, j \in \mathcal{V}$. Then $pickup(4)$, $red(i + j)$, $i < j + 2$ are atoms. As well,

$$red(j) \wedge (\forall k\ (k < j) \supset \neg red(k))$$

is a formula, and

$$\exists i, j((i + 2 < j) \wedge pickup(i) \wedge \neg red(j))$$

is a closed formula and so a query. The intent of this last formula is that it be true in a history in which $pickup$ is true at some time point and three or more time points later red is false.

The definition of truth of a query is as follows.

Definition 4 *Let* $H = (s_0, a_1, s_1, \ldots, a_k, s_k)$ *be a history over* Σ *of length* $k \leq n$, *and let* Q *be a query of* $\mathcal{Q}_{\Sigma,n}$.

We define $H \models_{\mathcal{Q}_{\Sigma,n}} Q$ *recursively as follows:*

1. *If* $Q = a(t)$ *is a ground action atom, then* $H \models_{\mathcal{Q}_{\Sigma,n}} Q$ *iff* $a = a_j$, *where* $j = \min(val(t), n)$.
2. *If* $Q = f(t)$ *is a ground fluent atom, then* $H \models_{\mathcal{Q}_{\Sigma,n}} Q$ *iff* $s_j(f) = 1$, *where* $j = \min(val(t), n)$.

3. *If Q is a ground time atom, then $H \models_{\mathcal{Q}_{\Sigma,n}} Q$ iff Q is true according to the rules of integer arithmetic.*
4. *If $Q = \neg\alpha$, then $H \models_{\mathcal{Q}_{\Sigma,n}} Q$ iff $H \not\models_{\mathcal{Q}_{\Sigma,n}} \alpha$.*
5. *If $Q = \alpha \supset \beta$, then $H \models_{\mathcal{Q}_{\Sigma,n}} Q$ iff $H \not\models_{\mathcal{Q}_{\Sigma,n}} \alpha$ or $H \models_{\mathcal{Q}_{\Sigma,n}} \beta$.*
6. *If $Q = \exists v\alpha$, then $H \models_{\mathcal{Q}_{\Sigma,n}} Q$ iff, for some $0 \leq i \leq n$, $H \models_{\mathcal{Q}_{\Sigma,n}} \alpha[v/i]$.*

If $H \models_{\mathcal{Q}_{\Sigma,n}} Q$ holds, then H *satisfies* Q. For simplicity, if $\mathcal{Q}_{\Sigma,n}$ is unambiguously fixed, we also write $\models$ instead of $\models_{\mathcal{Q}_{\Sigma,n}}$.

Note that the rationale of taking $j = \min(val(t), n)$ in Items 1 and 2 of Definition 4 is to take into account that a time term may refer to a time point which lies outside the interval determined by the length n of a given history. Intuitively, if $val(t)$ is greater than the length n of history H, then a ground atomic query $b(t)$, where b is either an action name or a fluent name, is satisfied by H if it is satisfied at the last state of H.

It is convenient to define certain additional operators in our languages. For instance, we define the following abbreviations, which basically correspond to well-known operators from linear temporal logic (LTL):

- $\Box b = \forall i\, b(i)$;
- $\Diamond b = \exists i\, b(i)$; and
- $(b \,\mathsf{U}\, g) = \exists i \big(g(i) \wedge \forall j((j < i) \supset b(j))\big)$.

Here, b and g are fluent or action names. Informally, $\Box b$ expresses that b *always* holds, $\Diamond b$ that b holds *eventually*, and $b \,\mathsf{U}\, g$ that b holds continually *until* g holds. Other LTL operators are likewise expressible.

4.2 Expressing Preferences among Histories

As described, we define a preference among two histories, H_l and H_h, directly in terms of a formula ϕ:

$$H_l \preceq_\phi H_h \quad \text{iff} \quad \langle H_l, H_h\rangle \models \phi. \tag{2}$$

The intent with $\langle H_l, H_h\rangle \models \phi$ is that ϕ expresses a condition in which H_h is at least as preferred as H_l. This requires that we be able to talk about the truth values of fluents and actions in H_l and H_h. In the previous subsection, we defined a query language on histories, $\mathcal{Q}_{\Sigma,n}$, and a notion of truth in a history for a query. Given these definitions, we are now in a position to introduce a *preference-specification language*, enabling the definition of preference relations between histories, as in (2).

Definition 5 *Let $\Sigma = \langle V, F, A\rangle$ be an action signature and $n \geq 0$ a natural number. We define the* preference-specification language *$\mathcal{P}_{\Sigma,n}$ over $\mathcal{Q}_{\Sigma,n}$ as follows:*

1. *The alphabet of $\mathcal{P}_{\Sigma,n}$ consists of the alphabet of the query language $\mathcal{Q}_{\Sigma,n}$, together with the symbols $\boldsymbol{l}$ and $\boldsymbol{h}$, called* history labels*, or simply* labels*.*
2. *Atoms of $\mathcal{P}_{\Sigma,n}$ are either time atoms of $\mathcal{Q}_{\Sigma,n}$ or expressions of the form $\ell : b(t)$, where $\ell \in \{\boldsymbol{l}, \boldsymbol{h}\}$ is a label and $b(t)$ is an action or fluent atom of $\mathcal{Q}_{\Sigma,n}$.*
   Atoms of the form $\ell : b(t)$ are also called labelled atoms*, with ℓ being the label* of *$\ell : b(t)$. We call $\ell : b(t)$* ground *iff $b(t)$ is ground.*
3. *Formulas of $\mathcal{P}_{\Sigma,n}$ are built from atoms, as introduced above, in a similar fashion as formulas of $\mathcal{Q}_{\Sigma,n}$. We call formulas of $\mathcal{P}_{\Sigma,n}$ also* preference formulas.
4. *A* preference axiom*, or simply* axiom*, is a closed preference formula, i.e., containing no free timestamp variables.*

For a formula α of $\mathcal{Q}_{\Sigma,n}$ and a history label $\ell \in \{\boldsymbol{l}, \boldsymbol{h}\}$, by $\ell : \alpha$ we understand the formula resulting from α by replacing each action and fluent atom $b(t)$ of α by the labelled atom $\ell : b(t)$. Informally, a labelled atom $\ell : b(t)$ expresses that b holds in a history associated with label ℓ, at time point t. The idea is that histories associated with label $\boldsymbol{h}$ are at least as preferred as histories associated with label $\boldsymbol{l}$. This is made precise as follows.

Definition 6 *Let Σ be an action signature and $n \geq 0$. Let ϕ be a preference axiom of $\mathcal{P}_{\Sigma,n}$ and H_l, H_h histories over Σ with $|H_i| \leq n$, for $i = l, h$. The relation $\langle H_l, H_h\rangle \models_{\mathcal{P}_{\Sigma,n}} \phi$ is recursively defined as follows:*

1. *If $\phi = \ell : b(t)$ is a ground labelled atom, for $\ell \in \{\boldsymbol{l}, \boldsymbol{h}\}$, then $\langle H_l, H_h\rangle \models_{\mathcal{P}_{\Sigma,n}} \phi$ iff*
   (a) *$H_l \models_{\mathcal{Q}_{\Sigma,n}} b(t)$, for $\ell = \boldsymbol{l}$, and*
   (b) *$H_h \models_{\mathcal{Q}_{\Sigma,n}} b(t)$, for $\ell = \boldsymbol{h}$.*
2. *If ϕ is a time atom, then $\langle H_l, H_h\rangle \models_{\mathcal{P}_{\Sigma,n}} \phi$ iff ϕ is true according to the rules of integer arithmetic.*
3. *If $\phi = \neg\psi$, then $\langle H_l, H_h\rangle \models_{\mathcal{P}_{\Sigma,n}} \phi$ just if $\langle H_l, H_h\rangle \not\models_{\mathcal{P}_{\Sigma,n}} \psi$.*
4. *If $\phi = \psi \supset \eta$, then $\langle H_l, H_h\rangle \models_{\mathcal{P}_{\Sigma,n}} \phi$ just if $\langle H_l, H_h\rangle \not\models_{\mathcal{P}_{\Sigma,n}} \psi$ or $\langle H_l, H_h\rangle \models_{\mathcal{P}_{\Sigma,n}} \eta$.*
5. *If $\phi = \exists v\, \psi$, then $\langle H_l, H_h\rangle \models_{\mathcal{P}_{\Sigma,n}} \phi$ iff, for some $0 \leq i \leq n$, $\langle H_l, H_h\rangle \models_{\mathcal{P}_{\Sigma,n}} \psi[v/i]$.*

If $\langle H_l, H_h\rangle \models_{\mathcal{P}_{\Sigma,n}} \phi$ holds, then $\langle H_l, H_h\rangle$ is said to *satisfy* ϕ. If Σ and n are clear from the context, we may simply write $\models$ instead of $\models_{\mathcal{P}_{\Sigma,n}}$.

Definition 7 *Let ϕ be a preference axiom of $\mathcal{P}_{\Sigma,n}$. For histories H_l, H_h over Σ of maximum length n, we define*

$$H_l \preceq_\phi H_h \quad \text{iff} \quad \langle H_l, H_h\rangle \models_{\mathcal{P}_{\Sigma,n}} \phi.$$

Note that the employment of the symbol $\preceq_\phi$ is purely suggestive at this point, since $\preceq_\phi$ may have none of the properties of an ordering.

We give some illustrations next.

Example 1 *The formula*

$$\phi = \big(\boldsymbol{h} : (\exists i f_1(i) \wedge \forall i \neg f_2(i))\big) \wedge \big(\boldsymbol{l} : (\exists i f_2(i) \wedge \forall i \neg f_1(i))\big)$$

expresses a preference of f_1 over f_2 in the sense that, for all histories H_l, H_h, we prefer H_h over H_l whenever it holds that H_h satisfies f_1 but not f_2, whilst H_l satisfies f_2 but not f_1.

Example 2 *For fluent or action name b, and a variable i, let $min[b, i]$ be given as follows:*

$$min[b, i] = \big(b(i) \wedge \forall k((k < i) \supset \neg b(k))\big).$$

Furthermore, for fluent or action names f_1, f_2, define

$$\phi = \big(\boldsymbol{l}\,{:}\,\exists i, j\big((i \geq j) \wedge min[f_1, i] \wedge min[f_2, j]\big)\big) \wedge \\ \big(\boldsymbol{h}\,{:}\,\exists i, j\big((i < j) \wedge min[f_1, i] \wedge min[f_2, j]\big)\big). \quad (3)$$

Then, $H_l \preceq_\phi H_h$ holds iff both f_1, f_2 are true in H_l and H_h, but f_1 is established earlier in H_h than in H_l.

In an easy extension of the preceding example, we can express that we prefer first that f_1 and f_2 occur together, and then that f_1 occur before f_2. As well, conditional preferences are trivially expressible in our approach.

Having the preference-specification language at hand, we define a *preference framework* as follows:

Definition 8 *Let Σ be an action signature and $n \geq 0$.*

A preference framework over Σ with horizon n *is a pair $(\boldsymbol{H}, \boldsymbol{P})$, where*

- *$\boldsymbol{H}$ is a set of histories over Σ having maximum length n, and*
- *$\boldsymbol{P}$ is a set of preference axioms over $\mathcal{P}_{\Sigma,n}$.*

The question then is how to select maximally preferred histories, given a preference framework $(\boldsymbol{H}, \boldsymbol{P})$. If $\boldsymbol{P}$ contains more than one axiom, this question involves the general problem of combining different relations and is actually independent from the concrete form of our preference language. We say more on this in Section 6.

5 Modelling Notions of Previous Approaches

We briefly outline how some extant preference notions in the context of planning, as well as other temporal notions from the literature, can be captured by our approach.

We start with the approach due to Son and Pontelli [2004]. As pointed out in Section 2, their language is based on *desires*, i.e., formulas constructed from $\wedge$, $\vee$, $\neg$, and the temporal (LTL) operators $\mathbf{always}$, $\mathbf{until}$, $\mathbf{next}$, and $\mathbf{eventually}$. Furthermore, they define a satisfaction relation $\models$ between histories and desires, and moreover define that H_h is preferred to H_l, given a desire ϕ, iff $H_h \models \phi$ but $H_l \not\models \phi$. Let us write $H_l \preceq_\phi^{PS} H_h$ if H_h is preferred to H_l in this sense. We can show the following result:

Theorem 1 *There is a translation τ mapping desires into query formulas of $\mathcal{Q}_{\Sigma,n}$, for some n, such that, given a desire ϕ, $H_l \preceq_\phi^{PS} H_h$ iff*

$$\langle H_l, H_h\rangle \models (\boldsymbol{h} : \tau(\phi)) \wedge (\boldsymbol{l} : \neg\tau(\phi)).$$

In fact, translation τ is constructible in polynomial time, and employs similar constructs as the abbreviations $\Box$, $\Diamond$, and U in Section 4.1.

In the approach of Delgrande *et al.* [2004] (cf. Section 2), a choice or temporal preference between actions and fluents induces a corresponding preference between histories. Considering the case of temporal preference $<_t$ for simplicity, the following result can be shown:

Theorem 2 *A history H_h is temporally preferred to a history H_l in the sense of Delgrande* et al. *[2004] under a single temporal preference $f_1 <_t f_2$ just in case that $\langle H_l, H_h\rangle \models \phi$, where ϕ is Formula (3) from Example 2.*

Finally, we consider the well-known *interval algebra* [Allen, 1983], another dominant approach in temporal reasoning, in which time intervals are the primitive objects. There are 13 basic relations between intervals, including relations such as *before, meets, overlaps*, etc. One could envisage a temporal preference language based on the interval algebra, wherein one may assert, for example, that the interval during which coffee is drunk overlaps with or starts before the interval in which a seminar takes place. To this end, one might define that a fluent f constitutes an interval just if it is true only for a contiguous set of time points:

$$interval(f) = \exists i, j\big((i \leq j) \wedge \\ \forall k(f(k) \equiv (i \leq k) \wedge (k \leq j))\big).$$

Then, the relation that an interval *meets* another can be defined by:

$$meets(f, g) = interval(f) \wedge interval(g) \wedge \exists i(f(i) \wedge \\ \neg f(i+1) \wedge \neg g(i) \wedge g(i+1)).$$

Other relations follow analogously.

6 Ordering on Histories

We consider here how, given a preference framework, one may determine those histories that are maximally preferred. In a preference framework, each preference formula defines a binary relation whose instances are pairs of relatively less- and more-preferred histories. Thus, one can express various independent preference relations that must in some sense be combined in order to come up with maximally preferred histories. However this problem, of combining differing preference orderings, is a general and difficult problem in and of itself, and is the object of ongoing research. Nonetheless, it is instructive to consider ways in which one may determine an overall preference ordering on histories, given a preference framework.

To begin, we can identify two base or generic approaches for determining (maximally) preferred histories. Recall that each $\phi \in \boldsymbol{P}$ induces a binary relation over $\boldsymbol{H}$ by $H_l \preceq_\phi H_h$ iff $\langle H_l, H_h\rangle \models_{\mathcal{P}_{\Sigma,n}} \phi$. As a base approach we can define:

Definition 9 *Let $(\boldsymbol{H}, \boldsymbol{P})$ be a preference framework over action signature Σ with horizon $n \geq 0$. Then, $H \in \boldsymbol{H}$ is a* (general) maximally preferred history *iff H is a maximal element of*

$$(\cup\{\preceq_\phi \mid \phi \in \boldsymbol{P}\})^*,$$

where, for binary relation R, R^* denoted the transitive closure of R.

Similarly, we can define a base approach founded on cardinality:

Definition 10 *Let $(\boldsymbol{H}, \boldsymbol{P})$ be a preference framework over action signature Σ with horizon $n \geq 0$. Furthermore, for $H \in \boldsymbol{H}$, let*

$$c(H) = |\{H' \preceq_\phi H \mid H' \in \boldsymbol{H} \text{ and } \phi \in \boldsymbol{P}\}|.$$

Then, $H \in \boldsymbol{H}$ is a (general) cardinality-based maximally preferred history *iff for every $H' \in \boldsymbol{H}$, we have $c(H') \leq c(H)$.*

We do not give a full discussion of the above approaches here—rather, for illustrative purposes, we elaborate on the cardinality-based approach.

Consider where one is given a set of desirable outcomes, of which the goal is to determine the history which satisfies the maximum number. Examples include fluents which are simply preferred to be true somewhere in a history, and *temporal preferences* [Delgrande *et al.*, 2004] in which one prefers that (pairs of) fluents become true in a specific order. In such cases, one wants to maximise the set of these desiderata. Assume then that we are given a set of (for simplicity) fluents $D = \{f, g, h, \ldots\}$, where we wish to prefer a history in which as many of these fluents are true as possible.

Given a set of histories $\boldsymbol{H}$ and preferences D, we define our preference framework by:

$$\boldsymbol{P} = \{(\boldsymbol{l} : \Box\neg d) \wedge (\boldsymbol{h} : \Diamond d) \mid d \in D\}. \quad (4)$$

Definition 10 yields a total preorder on histories, the maximum of which constitute the set of preferred histories. A refinement of this approach is to use set containment on satisfied preferences, rather than cardinality:

Definition 11 *Let $(\boldsymbol{H}, \boldsymbol{P})$ be a preference framework over action signature Σ with horizon $n \geq 0$, and assume that $\boldsymbol{P}$ is given by (4). For $H \in \boldsymbol{H}$, let*

$$s(H) = \{H' \preceq_\phi H \mid H' \in \boldsymbol{H} \text{ and } \phi \in \boldsymbol{P}\}.$$

Then, $H \in \boldsymbol{H}$ is a (general) set containment-based maximally preferred history *iff for every $H' \in \boldsymbol{H}$, we have $s(H') \subseteq s(H)$.*

Example 3 *Consider a preference framework, where we have simple preferences given by $D = \{f, g, h\}$. Assume that we have histories H_1, H_2, H_3, such that the following hold:*

$$H_1 \models \Diamond f \wedge \Diamond h, \quad H_2 \models \Diamond g, \quad H_3 \models \Diamond h.$$

According to Definition 10, H_1 is preferred; according to Definition 11, H_1 and H_2 are preferred.

Besides the base approaches for determining maximally preferred histories, as pointed out above, another possibility for generating a global ordering on histories would be to employ methods based on the approach by Brewka [2004] for building complex combinations of different preference strategies. An elaboration of such techniques is an issue for future work.

7 Conclusion

We have addressed the problem of expressing arbitrary preferences over histories (or linear temporal sequences interspersed with actions), inter alia addressing preferences in planning systems. We first defined a query language for specifying arbitrary conditions that may be satisfied by a history. Given this, we specified a second language for defining preferences. A preference induces a binary relation on histories, so that in an ordered pair of histories the second history is preferred to the first. From this, one can define a global ordering on the set of histories, the maximal elements of which are the preferred histories. The overall approach is very general and flexible; specifically we argue that previous approaches to preferences in planning are expressible in our formalism.

Acknowledgements

The second author was partially supported by the German Science Foundation (DFG) under grant SCHA 550/6, TP C. The collaboration of the second and third author was partially supported by the Information Society Technologies programme of the European Commission, Future and Emerging Technologies under the IST-2001-37004 WASP project.

References

[Allen, 1983] J. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(1):832-843, 1983.

[Brewka, 2004] G. Brewka. A rank based description language for qualitative preferences. *Proc. ECAI 2004*, p. 303-307. IOS Press, 2004.

[Delgrande *et al.*, 2004] J. Delgrande, T. Schaub, and H. Tompits. Domain-specific preferences for causal reasoning and planning. *Proc. KR 2004*, p. 673-682, 2004. AAAI Press.

[Eiter *et al.*, 2003] T. Eiter, W. Faber, N. Leone, G. Pfeifer, and A. Polleres. Answer set planning under action costs. *Journal of Artificial Intelligence Research*, 19:25-71, 2003.

[Fitting, 1990] M. Fitting. *First-Order Logic and Automated Theorem Proving*. Springer, New York, 1990.

[Gabbay, 1996] D. M. Gabbay. *Labelled Deductive Systems: Volume 1*, volume 33 of *Oxford Logic Guides*. Oxford University Press, 1996.

[Gelfond and Lifschitz, 1998] M. Gelfond and V. Lifschitz. Action languages. *Electronic Transactions on AI*, 3, 1998. Available at `http://www.ep.liu.se/rs/cis/1998/016/`.

[CI, 2004] Special issue on preferences in CP and AI. *Computational Intelligence*, 2(20), 2004.

[Levesque *et al.*, 1998] H. Levesque, F. Pirri, and R. Reiter. Foundations for the situation calculus. *Linkoping*

Electronic Articles in Computer and Information Science, 3(18), 1998. Available at `http://www.ep.liu.se/ea/cis/1998/018/`.

[Oztürk *et al.*, 2005] M. Oztürk, A. Tsoukiàs, and Ph. Vincke. Preference modelling. In *Multiple Criteria Decision Analysis*, p. 27-72. Springer, 2005.

[Son and Pontelli, 2004] T. Son and E. Pontelli. Planning with preferences in logic programming. *Proc. LPNMR 2004*, p. 247-260. Springer, 2004.