On Preference-based Search in State Space Graphs

Patrice Perny
LIP6 - University of Paris VI
4 Place Jussieu
75252 Paris Cedex 05, France
patrice.perny@lip6.fr

Olivier Spanjaard
LAMSADE - University of Paris IX
Place du Maréchal de Lattre de Tassigny
75775 Paris Cedex 16, France
spanjaar@lamsade.dauphine.fr

Abstract

The aim of this paper is to introduce a general framework for preference-based search in state space graphs with a focus on the search of the preferred solutions. After introducing a formal definition of preference-based search problems, we introduce the PBA* algorithm, a generalization of the A* algorithm, designed to process quasi-transitive preference relations defined over the set of solutions. Then, considering a particular subclass of preference structures characterized by two axioms called *Weak Preadditivity* and *Monotonicity*, we establish termination, completeness and admissibility results for PBA*. We also show that previous generalizations of A* are particular instances of PBA*. The interest of our algorithm is illustrated on a preference-based web access problem.

Introduction

In heuristic search, the quality of a potential solution is often represented by a scalar-valued cost function to be minimized. This is the case in classical state space graphs problems, where the value of a path between two nodes is defined as the sum of the costs of its arcs. This particular feature makes it possible to resort to constructive search algorithms like A* and A$^*_\epsilon$ (Hart, Nilsson, and Raphael 1968; Pearl 1984), performing the implicit enumeration of feasible solutions, directed by a numerical evaluation function.

However, in practical situations, preferences over solutions are not always representable by such a numerical cost function and the traditional search algorithms do not fit. As an illustrative example, consider the following problem derived from (Etzioni *et al.* 1996; Papadimitriou and Yannakakis 2000):

The Web Access Problem. Suppose that you want to retrieve a list of documents from the world-wide web, requesting an "information marketplace" supported by automatic billing protocols. In order to gather the desired information, you have the possibility to query n information sources (web sites) simultaneously, each being accessible with charges. The following data are available for each site: the access cost c_i to site i (expressed in USD), the reliability r_i of the site (qualitative evaluation on a finite ordered scale), the probability p_i that a given information will be found on site i (this probability represents the richness of the site) and the access time t_i to site i. The problem is to determine which subset $S \subseteq \{1, \ldots, n\}$ of sites should we choose for launching a multiple query?

Assuming that your resources are bounded in time and money by $\bar{t}$ and $\bar{c}$ respectively, the search is restricted to subsets S verifying the following constraints:

$$\max_{i\in S}\{t_i\} \leq \bar{t} \text{ and } \sum_{i\in S} c_i \leq \bar{c} \tag{1}$$

Let us assume that, for any pair of feasible solutions S, S', the preferences in terms of reliability and richness are respectively represented by relations $\succ_1$, $\succ_2$. We introduce first the preferences in terms of reliability:

$$S \succ_1 S' \iff \begin{cases} \exists j \leq k,\ \forall i < j,\ L_i^S = L_i^{S'} \\ \text{and } L_j^S > L_j^{S'} \end{cases} \tag{2}$$

where $k = \min\{|S|, |S'|\}$ and L^S is the sublist of $(r_i)_{i\in S}$ containing the k greatest values sorted by decreasing order. The definition of $\succ_1$ aims at favoring subsets including at least one site of high reliability. This principle could be represented by a maximum operation, but the lexicographic comparison rule used here is a refinement, known as *LexiMax* (for more details, see (Dubois, Fargier, and Prade 1996)). We introduce now the preferences in terms of richness:

$$S \succ_2 S' \iff \prod_{i\in S}(1-p_i) < \prod_{i\in S'}(1-p_i) \tag{3}$$

The quantity $\prod_{i\in S}(1-p_i)$ represents the probability of failure of the multiple query characterized by S, under the assumption that the success of one source is independent of the success or failure of the other sources.

It can easily be checked that relations $\succ_1$ and $\succ_2$ are transitive. From these relations, we derive the following dominance relation:

$$S \succ S' \iff S \succ_1 S' \text{ and } S \succ_2 S' \tag{4}$$

The question under consideration is now to determine the subset of non-dominated solutions with respect to $\succ$, in other words the set of feasible solutions which are not dominated by any other feasible solution.

One can imagine various reformulations of this problem as a preferred-path problem in a state space graph. For example, consider a graph where the nodes represent all possible decisions concerning subsets of type $S_i = \{1, \ldots, i\}$ for $i = 1, \ldots, n$. Formally, each node is characterized by a vector $(s, b_1, \ldots, b_i)$ representing a state in which the decision concerning the i first sites has been made. The component b_k is a boolean which is true if and only if site k is selected, for every $k \in S_i$, and the starting node is (s) and corresponds to the initial state where no decision has been made. In this graph, each node of type $(s, b_1, \ldots, b_i)$ has two successors, $(s, b_1, \ldots, b_i, 0)$ and $(s, b_1, \ldots, b_i, 1)$ or less if some of them fail to satisfy the constraints (1). The goals are all feasible nodes of type $(s, b_1, \ldots, b_n)$. Clearly, we have to find the preferred paths from (s) to these nodes.

In this problem, we have to deal with a partial preference relation $\succ$ that cannot be represented by an additive scalar cost function to be minimized, and thus A* algorithm does not apply. Remark that none of the common preference-based extensions of A* applies to such a problem. The U* algorithm (White, Stewart, and Carraway 1992) which is a variation of A* designed to deal with utility-based preferences cannot be used because $\succ$ is not necessarily complete and thus, cannot be represented by a single utility function. The MOA* which is specially designed to process multiple objective does not apply either. Indeed, a preference relation like $\succ_1$ is not representable by a criterion function (note that, on that dimension, the value of a subset S may vary depending on the other subset it is compared to). Moreover, $\succ_1$ as many other partial preference orders cannot easily be represented using several criteria. Indeed, even if any partial order can be represented by an arbitrary large number of criteria (using a dominance relation), the computation of this representation is prohibitive due to the combinatorial number of elements to be compared. The same arguments apply to the ABC algorithm proposed in (Logan and Alechina 1998) for the search under flexible constraints.

For these reasons, we need a general framework to cope with partial preference relations in state space graphs. A similar statement has been discussed and clearly illustrated in (Dasgupta, Chakrabarti, and DeSarkar 1996; Müller 2001), in the context of game tree search. More generally, a systematic approach admitting any preference relation to direct the search is worth studying. In particular, this would make it possible to resort to useful qualitative preference models as those recently developed in the AI community, see e.g. (Boutilier 1994; Dubois, Fargier, and Prade 1996; Brafman and Tennenholtz 1997; 2000).

This idea is already present in the framework of constraint satisfaction problems. For example, the algebraic generalization of CSP algorithms proposed in (Schiex, Fargier, and Verfaillie 1995; Bistarelli, Montanari, and Rossi 1997) significantly increases the range of potential application of the algorithms by considering all ordered semiring structures on valuations. Our aim in this paper is to follow a similar line for search algorithms in state space graphs.

We are going to generalize and factorize various extensions of the A* algorithm at once, keeping only the key properties of preferences structures used to direct the search. The basic idea is to define a general framework where evaluation functions (like f, g and h in A*) are replaced by multi-sets of valuations, partially ordered by the preference $\succ$. In this framework, we will introduce the PBA* algorithm, a general preference-based extension of A*. Then, we will characterize a wide class of preferences structures for which our algorithm is admissible and show how this algorithm should be modified to cope with preference structures out of this class.

Preliminary Definitions

Let us first recall the following definitions about binary relations.

Definition 1 *For any binary relation $\succsim$ on a set X, the asymmetric and symmetric part of $\succsim$ are defined by:*
$\forall x, x' \in X, \quad (x \succ x') \iff ((x \succsim x')$ *and not*$(x' \succsim x))$
$\forall x, x' \in X, \quad (x \sim x') \iff ((x \succsim x')$ *and* $(x' \succsim x))$

Definition 2 *For any binary relation $\succsim$ defined on a set X, the set of* maximal *elements is defined by:*

$$M(X, \succsim) = \{x \in X \quad | \quad \forall x' \in X \quad \textit{not}(x' \succ x)\}$$

In this paper, $\succsim$ represents a weak-preference relation and therefore $\succ$ is the associated strict preference relation. The proposition $x \succsim x'$ means x is at least as good as x' whereas $x \succ x'$ means x is strictly preferred to x'.

Definition 3 *A binary relation $\succsim$ defined on a set X is said to be:*

- *complete iff* $\forall x, x' \in X,\ x \succsim x'$ *or* $x' \succsim x$
- *quasi-transitive iff* $\succ$ *is transitive*

We have to introduce the notion of *multi-set*, which is an unordered collection of values which may have duplicates. More formally:

Definition 4 *For any set E, the set $\mathcal{M}(E)$ of* multi-sets *of E is the set of functions $x : E \to \mathbb{N}$, representing the number of occurrences of each element. We call* support *of a multi-set x the set $E_x = \{e \in E \mid x(e) \neq 0\}$. The empty multi-set is denoted $\mathbb{1}_\emptyset$.*

We also have to define the sum of two multi-sets:

Definition 5 *Let x and y be two multi-sets in $\mathcal{M}(E)$. The addition and the difference of x and y are defined as follows:*

$$\forall e \in E, (x + y)(e) = x(e) + y(e)$$
$$\forall e \in E, (x - y)(e) = \max(0, x(e) - y(e))$$

The inclusion of a multi-set x in a multi-set y is defined as follows:

$$x \subseteq y \iff \forall e \in E_x,\ \ x(e) \leq y(e)$$

The cardinality of a multi-set x is defined as follows:

$$|x| = \sum_{e \in E_x} x(e)$$

Problem Formulation

A*-like search algorithms look for best paths in a state space graph. Let N be a finite set of nodes, $A \subseteq N \times N$ be a set of directed valued arcs, $N(P)$ be the set of all nodes on a path P and $S(n)$ be the set of all successors of a node n. We denote $\mathcal{P}(n, X)$ the set of all paths linking n to a node in X. Let $s \in N$ be the source of the graph and $\Gamma \subseteq N$ be the subset of goal nodes. Then, $\mathcal{P}(s, \Gamma)$ denotes the set of all paths from s to a goal node $\gamma \in \Gamma$, and $\mathcal{P}(n, n')$ the set of all paths linking n to n'. We call solution-path a path from s to a goal node $\gamma \in \Gamma$. Moreover, we assume we get a valuation function $v : A \to E$. Let $P \cap \mathcal{P}(n, n')$ be the segment of P linking n to n'. Let x_P be the multi-set of valuations of arcs on P. We assume a reflexive and quasi-transitive preference relation $\succsim$ is defined on $\mathcal{M}(E)$ (set of multi-sets of valuations). Notice that we will always consider finite multi-sets (*i.e.* with finite supports).

Concerning the definition of the preference relation, we can distinguish two main cases:

1. Most of the time, the preference relation $\succsim$ on $\mathcal{M}(E)$ is constructed from a commutative and associative internal composition operator $\otimes$ on the valuation space E, and a preference relation $\succsim_E$ on E. We denote $e^k = e \otimes \ldots \otimes e$ (k times). Then, $v(x)$ denotes the image of a multi-set x of elements in E, *i.e.*: $v(x) = \bigotimes_{e \in E_x} e^{x(e)}$. It leads to the following preference relation on $\mathcal{M}(E)$:
$$\forall x \in \mathcal{M}(E),\ x \succsim y \iff v(x) \succsim_E v(y)$$
For instance, for the usual A* algorithm, E is $\mathbb{R}$, $\otimes$ is the sum operator $+$ and $\succsim_E = \leq$. In such a case, we can work directly in the valuation space E.

2. Sometimes, it is not possible to represent the preference relation $\succsim$ by resorting to an internal composition operator on E. For example, in the web acces problem, no composition operator on the reliability scale E could induce a convenient representation of the *LexiMax* preference relation (see Equation (2)), which writes:
$$x \succ_1 y \iff L^x \neq L^y \text{ and } \max_{e \in E_{L^x - L^y}} e > \max_{e \in E_{L^y - L^x}} e$$
in terms of multi-sets. In such a case, we have to work in $\mathcal{M}(E)$ and therefore to design an algorithm able to operate in such a framework.

We denote $M(\mathcal{P}, \succsim)$ the set of maximal paths in a set $\mathcal{P}$:
$$M(\mathcal{P}, \succsim) = \{P \in \mathcal{P} \mid \forall P' \in \mathcal{P} \quad \text{not}(x_{P'} \succ x_P)\}$$

We call *multi-valuation* a multi-set of valuations. We introduce now the main issue of this paper:

The Preference-Based Search Problem. Consider a finite state space graph G, *i.e.* a graph containing a finite number of arcs and therefore a finite number of non-isolated vertices, and assume there exists at least one path P_0 with a finite length (number of arcs, denoted $|P_0|$) between s and a goal node $\gamma \in \Gamma$. The goal is to determine the entire set $M(\{x_P \mid P \in \mathcal{P}(s, \Gamma)\}, \succsim)$.

From now on, unless otherwise stated, we work in $\mathcal{M}(E)$ for the sake of generality. For example, coming back to the web access problem mentioned in the introduction, the multi-valuation of any path from (s) to a node $(s, b_1, \ldots, b_k)$ is the multi-set of pairs (r_i, p_i), $i \leq k$ such that $b_i =$ true.

The PBA* Algorithm

We propose here a variation of the A* search algorithm specifically designed to work with a preference relation on $\mathcal{M}(E)$. It is more general than an algebraic approach of the problem (Rote 1998; Bistarelli, Montanari, and Rossi 1997), that would consist in assuming there is a partially ordered semigroup on an evaluation space E. Indeed, we do not assume the transitivity of the symmetric part of the preference relation, neither the existence of an internal composition operator on E (which permits to consider the *LexiMax* preference relation). We call our algorithm PBA* for *Preference-Based A**. At any node n, we consider: $G^*(n)$ the set of maximal multi-valuations of paths P in $\mathcal{P}(s, n)$, $H^*(n)$ the set of maximal multi-valuations of paths P in $\mathcal{P}(n, \Gamma)$ and $F^*(n)$ the set of maximal multi-valuations of paths P in $\mathcal{P}(s, \Gamma)$ such that $n \in N(P)$. As soon as the Bellman principle is verified, this last set derives from the two other ones as follows:
$$F^*(n) = M(G^*(n) \odot H^*(n), \succsim)$$
where $\odot$ is an internal composition operator defined, for any sets X, Y of multi-valuations, by:
$$X \odot Y = \bigcup_{x \in X, y \in Y} (x + y)$$

As in the A* algorithm, $G^*(n)$, $H^*(n)$ and $F^*(n)$ are unknown during the search. Consequently, the evaluation of a node n is based on the following approximations: $G(n)$ the set of maximal multi-valuations of generated paths, $H(n)$ the set of multi-valuations resulting from a heuristic estimation of $H^*(n)$ and $F(n) = M(G(n) \odot H(n), \succsim)$. $H(n)$ is assumed to be *coincident*, in other words, the following property holds: $\forall \gamma \in \Gamma, H(\gamma) = \mathbb{1}_\emptyset$.

As in A*, the PBA* algorithm divides the set of generated nodes into a set O of open nodes (labeled but not yet developed) and a set C of closed nodes (labeled and already developed). At any iteration, we develop a node $n \in O$ such that $F(n)$ contains at least one maximal multi-valuation among labels. Formally, one chooses n in the set MAX of most promising nodes, defined as a subset of nodes $n \in O$ such that:
$$\exists f \in F(n),\ \begin{cases} \forall n' \in O,\ \forall f' \in F(n'),\ \text{not}(f' \succ f) \\ \forall c \in CHOICE,\ c \neq f \text{ and not}(c \succ f) \end{cases} \tag{5}$$
where $CHOICE$ denotes the current set of maximal labels at the goal nodes. The goal nodes which have already been selected for development are stored in a set denoted $GOALS$. More precisely, we propose Algorithm 1 given below:

Algorithm 1 *PBA**

Initialization: $O \leftarrow \{s\}$; $C \leftarrow \emptyset$; $G(s) \leftarrow \emptyset$; $MAX \leftarrow \{s\}$; $GOALS \leftarrow \emptyset$; $CHOICE \leftarrow \emptyset$; $n \leftarrow s$;
While [$MAX \neq \emptyset$]
  Move n from O to C
  If [$n \notin \Gamma$] then for $n' \in S(n)$ do
    If [$n' \notin O \cup C$] then:
      $G(n') \leftarrow M(G(n) \odot x_{(n,n')}, \succsim)$
      $F(n') \leftarrow M(G(n') \odot H(n'), \succsim)$
      Put n' in O
    end
    else n' is already labelled, then:
      $G(n') \leftarrow M(G(n') \cup (G(n) \odot x_{(n,n')}), \succsim)$
      $F(n') \leftarrow M(G(n') \odot H(n'), \succsim)$
      If $G(n')$ is modified, put n' in O
    end
  end
  If [$O \neq \emptyset$] then
    compute MAX according to Equation (5)
  end
  Else $MAX = \emptyset$
  If [$MAX \neq \emptyset$] choose $n \in MAX$ with respect to an heuristic specific to the application, with priority to $n \in \Gamma$.
  If [$MAX \neq \emptyset$] and [$n \in \Gamma$] then:
    $GOALS \leftarrow GOALS \cup \{n\}$
    $CHOICE \leftarrow M(CHOICE \cup G(n), \succsim)$
  end
end
If [$GOALS = \emptyset$] then exit with failure;
Exit with all the efficient paths obtained by backtracking from the labels in $CHOICE$;

end

Remark 1 *If a commutative and associative internal operator has been used to define the preference relation, then the entire search can be done in the valuation space (i.e. the labels are in E instead of $\mathcal{M}(E)$).*

Remark 2 *For the sake of simplicity, we have omitted the management of pointers allowing the preferred paths to be recovered. This can be easily implemented, as shown in (Stewart and White III 1991). This additional functionality is assumed to exist in the sequel.*

Axioms

As the preference relation used is not specified in our algorithm, we introduce here some axioms on preference which will be necessary to establish the termination, completeness and admissibility of the algorithm.

Weak Preadditivity (WP)

$$\forall x, y, z \in \mathcal{M}(E),\ x \succ y \Longrightarrow x + z \succ y + z$$

This axiom is a weak version of De Finetti's qualitative additivity (De Finetti 1974) and can also be seen as a qualitative counterpart of the monotonicity property considered in dynamic programming (Mitten 1964; Morin 1982). Moreover, if an internal composition operator $\otimes$ on E has been used to define the preference relation $\succsim$, this axiom translates into distributivity of $\otimes$ over the selection operation represented by $M(., \succsim)$ (Zimmermann 1981; Rote 1998). Note that WP holds within the framework of A* and its direct extensions mentioned above. Moreover, concerning the web access problem, this axiom is also satisfied by the preference relations defined in Equations (2), (3), and therefore (4).

Proposition 1 *When $\succsim$ satisfies WP, the Bellman principle is verified: any subpath of a maximal path is maximal.*

Proof. Let P be a path from a node n_1 to a node n_4. Let n_2, n_3 be two nodes along this path, $P' = P \cap \mathcal{P}(n_1, n_2)$, $P'' = P \cap \mathcal{P}(n_2, n_3)$ and $P''' = P \cap \mathcal{P}(n_3, n_4)$. Assume that P'' is not maximal, then there exists $Q \in \mathcal{P}(n_2, n_3)$ such that $x_Q \succ x_{P''}$ and by WP we get $x_{P' \cup Q \cup P'''} = x_Q + x_{P' \cup P'''} \succ x_{P''} + x_{P' \cup P'''} = x_P$. □

Remark 3 *Whenever $\succsim$ is transitive, $\sim$ is an equivalence relation defining indifference classes. Hence, one might be interested in determining only one maximal path by indifference class. For that purpose, the key property would be:*

$$\forall x, y, z \in \mathcal{M}(E),\ x \succ y \Longrightarrow x + z \succsim y + z \quad (\mathrm{WP}')$$

instead of Weak Preadditivity, thus opening new possibilities. For example, the preference relation defined by $x \succsim y$ iff $\max_{e \in E_x} e \geq \max_{e \in E_y}$ satisfies WP$'$ *but not* WP.

In the classical A* algorithm, the hypothesis of a strictly positive inferior bound on the valuations of the arcs insures that a cyclic path cannot be maximal. Therefore, as the graph is finite and each reopening of a node corresponds to the detection of a new acyclic path, the algorithm is guaranteed to terminate in finite time. However, in our algorithm, we have to reopen a node as soon as we detect a new path that is not worse than any other label at this node. Therefore, we need the following axiom to insure the termination of the algorithm:

Monotonicity (M)

$$\forall x, y \in \mathcal{M}(E),\ \ x \subset y \Longrightarrow x \succ y$$

This monotonicity axiom guarantees that a subpath is always preferred to the path it is extracted from. In other words, a path including a cycle cannot be maximal. Remark that in acyclic graphs (as in the web access problem), that axiom can be omitted.

Termination and Completeness

Termination and completeness are both algorithmic properties of main interest. As usual, an algorithm is said to *terminate* if it necessarily stops after a finite number of iterations. It is said to be *complete* if it outputs at least one solution-path as soon as a solution-path exists. The following lemma, valid for A*, still holds for PBA*, since its proof does not depend on the preference relation used.

Lemma 1 *Let $n \in N$ and $P \in \mathcal{P}(s, n)$. At any step of the algorithm, either at least one node on P is in O, or every node on P is in C.*

The following termination result holds thanks to the monotonicity axiom entailing the cancellation of any cyclic path during the search.

Theorem 1 *If $\succsim$ satisfies M, PBA* terminates for any problem such that at least one solution-path exists.*

Proof. Consider a yet developed node n. For this node to be redeveloped, it is necessary to find another path which is maximal with respect to $G(n)$. Such a path is necessarily acyclic due to axiom M. Since there exists only a finite number of acyclic paths in a finite graph, n can only be developed a finite number of times. Therefore PBA* terminates after a finite number of iterations. $\square$

The following result holds also for any best-first strategy (see (Pearl 1984) for the definition of a best-first strategy):

Theorem 2 *If $\succsim$ is quasi-transitive and satisfies M, PBA* is complete.*

Proof. While no solution-path is detected, there is necessarily a node $n \in N(P_0)$ which is in O (by Lemma 1). As $O \neq \emptyset$ and $CHOICE = \emptyset$, $MAX = M(\bigcup_{n \in O} F(n), \succsim)$. Hence, MAX cannot be empty due to the quasi-transitivity of $\succsim$. However, thanks to M and Theorem 1, we know that the algorithm terminates after a finite number of iterations. Therefore, as the termination rule is $MAX = \emptyset$, a solution-path is necessarily found. This establishes the completeness of the algorithm. $\square$

It can be shown that PBA* is complete even for infinite (but locally finite) graphs as soon as $\succsim$ is quasi-transitive and the following archimedean axiom is verified: If x is a finite multi-set of $\mathcal{M}(E)$, then there exists $k \in \mathbb{N}^*$, $\forall y \in \mathcal{M}(E)$, $|y| \geq k \Longrightarrow x \succ y$. Unfortunately, such a result is rather theoretical since the archimedean axiom is not satisfied by several natural preference relations. For instance, Equation (2) in our example fails to satisfy this axiom.

Admissibility

We now define the notion of *optimistic* heuristic in our framework, in order to establish the admissibility of PBA*. In this framework, an algorithm is said to be *admissible* if it guarantees to terminate with the whole set $M(\{x_P \,|\, P \in \mathcal{P}(s, \Gamma)\}, \succsim)$ for any problem such that at least one solution path exists.

Definition 6 *An optimistic heuristic is a set H of multi-valuations fulfilling the following conditions: $\forall n \in N$, $\forall h^* \in M(H^*(n), \succsim)$, $\exists h \in H(n)$ s.t. $h \succ h^*$ or $h = h^*$.*

For example, in the web access problem, we can choose as heuristic, at node $(s, b_1, \ldots, b_k)$, the multi-set of pairs (r_i, p_i) for $i > k$. Let us now introduce two intermediary results:

Lemma 2 *Let P be a maximal path from s to a node n (possibly outside Γ) in the graph, and n' be the first open node on this path. If $\succsim$ satisfies WP, then there exists $g \in G(n')$ such that $g = x_{P \cap \mathcal{P}(s,n')}$.*

Proof. The path $P' = P \cap \mathcal{P}(s, n')$ has been detected since all the predecessors of n' on P are closed. Moreover, by the Bellman principle which holds thanks to WP, $P \in M(\mathcal{P}(s,n), \succsim)$ implies $P' \in M(\mathcal{P}(s,n'), \succsim)$. Consequently: $\exists g \in G(n')$ such that $g = x_{P \cap \mathcal{P}(s,n')}$ $\square$

Lemma 3 *Let $\succsim$ be a preference relation which satisfies WP. At every step of the algorithm, if $P \in M(\mathcal{P}(s,\Gamma), \succsim)$ and P is not yet detected, there exists in O a node n' of P and $f \in F(n')$ such that $f \succ x_P$ or $f = x_P$.*

Proof. Let n' be the first open node on P. Let $P' = P \cap \mathcal{P}(s, n')$ and $P'' = P \setminus P'$. By Lemma 2, $\exists g \in G(n')$ such that $g = x_{P'}$. On the other hand, $\exists h \in H(n')$ such that $h \succ x_{P''}$ or $h = x_{P''}$ since h is optimistic. Therefore $\exists f \in F(n')$ such that $f = g + h = x_{P'} + h \succ x_{P'} + x_{P''}$ $= x_P$ by WP or $f = g+h = x_{P'}+h = x_{P'}+x_{P''} = x_P$. $\square$

We now present the main result of this section:

Theorem 3 *If $\succsim$ is quasi-transitive and satisfies WP and M, then PBA* is admissible.*

Proof. Thanks to M and Theorem 1, the algorithm terminates after a finite number of iterations. Assume that there exists x_P in $M(\{x_P | P \in \mathcal{P}(s,\Gamma)\}, \succsim)$ that is not in $CHOICE$ when PBA* stops. As P is not detected, $O \neq \emptyset$ (by Lemma 1). When PBA* stops, $MAX = \emptyset$ and therefore all the nodes in O satisfy: $\forall f \in F(n)$, $\exists c \in CHOICE$, $c = f$ or $c \succ f$. However, by Lemma 3, there exists a node n' of P in O and $f \in F(n')$ such that $f \succ x_P$ or $f = x_P$. Therefore $c \succ x_P$ or $c = x_P$, but $c \succ x_P$ contradicts the maximality of P and $c = x_P$ contradicts $x_P \notin CHOICE$. Hence, $M(\{x_P | P \in \mathcal{P}(s,\Gamma)\}, \succsim) \subseteq CHOICE$. Moreover, by construction, $\succ$ is empty on $CHOICE$, which completes the proof. $\square$

Considering the web access problem mentioned in the introduction, since the state space graph is acyclic and the preference relation $\succ$ defined by (4) is transitive and satisfies WP, Theorem 3 shows that PBA* can be used to determine the preferred solutions.

Approximation of the Preference Relation

It should be noticed that the previous results, despite their generality, do not cover the entire class of "rational" preference relations. For example, for bi-criteria optimization problems characterized by two valuations v_1 and v_2, the egalitarian preference: $x_P \succsim x_Q \Leftrightarrow \max\{v_1(x_P), v_2(x_P)\}$ $\leq \max\{v_1(x_Q), v_2(x_Q)\}$ (where $v_j(x_P) = \sum_{a \in P} v_j(a)$) fails to satisfy the WP axiom. Nevertheless, PBA* might be properly used with an approximation $\succsim'$ of $\succsim$ which satisfies the WP axiom (e.g. $x_P \succsim' x_Q \Leftrightarrow v_j(x_P) \leq v_j(x_Q)$ for $j = 1, 2$). For this reason, we introduce the following definition, which makes sense only in acyclic graphs (to escape the monotonicity problem):

Definition 7 *A preference relation $\succsim'$ is an approximation of $\succsim$ if and only if: $\forall X \subseteq \mathcal{M}(E)$, $M(X, \succsim) \subseteq M(X, \succsim')$*

Then, we have:

Proposition 2 *Let $\succsim$ be a preference relation and $\succsim'$ an approximation of $\succsim$ which is quasi-transitive and satisfies WP,* *applying PBA* with $\succsim'$ yields a superset of $M(\{x_P \mid P \in \mathcal{P}(s,\Gamma)\}, \succsim)$.*

Moreover, during the search with respect to $\succsim'$, we can remove labels f for which there exists a detected solution-path P such that $x_P \succ f$. By this way, we reduce computational efforts and we get exactly the set $M(\{x_P | P \in \mathcal{P}(s,\Gamma)\}, \succsim)$.

Comparison with Other A* Algorithms

In this section, we show that various well-known variations of the A* algorithm can be seen as particular instances of PBA*. Each variation is characterized by the choice of the valuation set E, the valuation space (E or $\mathcal{M}(E)$) and the preference relation (which mostly satisfies WP and M). This is also the case for the A* algorithm itself. Indeed, we can instantiate our model as follows: $E = \mathbb{R}$, the valuation space is E, $\otimes = +$ and $x_P \succsim x_Q \Leftrightarrow \sum_{a \in P} v(a) \leq \sum_{a \in Q} v(a)$, and the goal is to find one maximal path.

The multi-criteria variation of A* algorithm, namely MOA* (Stewart and White III 1991), gives the following instance: $E = \mathbb{R}^n$, the valuation space is E, $x_P \succsim x_Q \Leftrightarrow \forall i = 1, \ldots, n, \sum_{a \in P} v_i(a) \leq \sum_{a \in Q} v_i(a)$, and the goal is to determine $M(\mathcal{P}(s,\Gamma), \succsim)$.

The multiattribute utility variation of A* algorithm, namely U* (White, Stewart, and Carraway 1992), gives the following instance (here, $v(x)$ denotes a multiattribute reward vector): $E = \mathbb{R}^n$, the valuation space is E, $x_P \succsim x_Q \Leftrightarrow u(v(x_P)) \geq u(v(x_Q))$, approximated by: $x_P \succsim' x_Q \Leftrightarrow \forall i = 1, \ldots, n, \sum_{a \in P} v_i(a) \leq \sum_{a \in Q} v_i(a)$, and we are looking for one maximal path.

The ABC algorithm (Logan and Alechina 1998) is a variation of U* designed for multicriteria problems where the overall objective is expressed by n soft constraints on criteria. E is the criteria space and the preference relation is $x \succsim y \Leftrightarrow (x_1, \ldots, x_n) \sqsubseteq (y_1, \ldots, y_n)$ where x_j (resp. y_j) is a boolean representing the satisfaction index of constraint C_j and $\sqsubseteq$ is any reflexive and transitive extension of the dominance order on boolean vectors (induced by $1 \sqsubseteq 0$).

Conclusion

We have proposed a new algorithm for preference-based search which extends previous A*-like algorithms in a very natural way. The termination, completeness and admissibility results established in the paper prove its practical interest for a wide class of preference relations characterized by axioms WP and M. When preferences escape this class, we still have the possibility to determine the preferred paths, provided a convenient approximation of the preference relation is found. A more elaborate study on the construction of efficient approximations seems to be of main interest.

References

Bistarelli, S.; Montanari, U.; and Rossi, F. 1997. Semiring-based constraint satisfaction and optimization. *Journal of the Association for Computing Machinery* 44(2).

Boutilier, C. 1994. Toward a logic for qualitative decision theory. In *Proceedings of the Fourth International Conference on Principles of Knowledge Representation and Reasoning, KR-94*, 75–86.

Brafman, R., and Tennenholtz, M. 1997. On the axiomatization of qualitative decision theory. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence, AAAI-97*, 76–81. AAAI Press/MIT Press.

Brafman, R., and Tennenholtz, M. 2000. An axiomatic treatment of three qualitative decision criteria. *Journal of the ACM* 47(3):452–482.

Dasgupta, P.; Chakrabarti, P.; and DeSarkar, S. 1996. Searching game trees under a partial order. *Artificial Intelligence* 82:237–257.

De Finetti, B. 1974. *Probability Theory Vol. I.* London: Wiley.

Dubois, D.; Fargier, H.; and Prade, H. 1996. Refinements of the maximin approach to decision-making in fuzzy environment. *Fuzzy Sets and Systems* 81:103–122.

Etzioni, O.; Hanks, S.; Jiang, T.; Karp, R. M.; Madari, O.; and Waarts, O. 1996. Efficient information gathering on the internet. In *Proceedings of the 37th IEEE Symposium on Foundation of Computer Science*, 234–243.

Hart, P. E.; Nilsson, N. J.; and Raphael, B. 1968. A formal basis for the heuristic determination of minimum cost paths. *IEEE Trans. Syst. and Cyb.* SSC-4 (2):100–107.

Logan, B., and Alechina, N. 1998. A* with bounded costs. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence, AAAI-98*. AAAI Press/MIT Press.

Mitten, L. 1964. Composition principles for the synthesis of optimal multi-stage processes. *Operations Research* 12.

Morin, T. 1982. Monotonicity and the principle of optimality. *J. of Math. Analysis and Applications* 86:665–674.

Müller, M. 2001. Partial order bouding: a new approach to evaluation in game tree search. *Artificial Intelligence* 129:279–231.

Papadimitriou, C. H., and Yannakakis, M. 2000. On the approximability of trade-offs and optimal access of web sources. In *Proc. of the 41th IEEE Symp. on FOCS*, 86–92.

Pearl, J. 1984. Intelligent search strategies for computer problem solving.

Rote, G. 1998. Paths problems in graphs. Technical report, Institute of mathematics, technical university of Graz.

Schiex, T.; Fargier, H.; and Verfaillie, G. 1995. Valued constraint satisfaction problems: Hard and easy problems. In Mellish, C., ed., *Proceedings of the International Joint Conference on Artificial Intelligence, IJCAI-95*.

Stewart, B., and White III, C. 1991. Multiobjective A*. *Journal of the Association for Computing Machinery* 38(4):775–814.

White, C.; Stewart, B.; and Carraway, R. 1992. Multiobjective, preference-based search in acyclic OR-graphs. *European Journal of Operational Research* 56:357–363.

Zimmermann, U. 1981. *Linear and combinatorial optimization in ordered algebraic structures*. Number 10 in Annals of discrete mathematics. North holland publishing company.