Solving Non-deterministic Planning Problems with Pattern Database Heuristics

Pascal Bercher
Institute of Artificial Intelligence
University of Ulm, Germany

Robert Mattmüller
Department of Computer Science
University of Freiburg, Germany

KI 2009, Paderborn

Given:	A non-deterministic planning problem. (Informally: Initial state, actions, goal states. Nondeterminism: actions may have several outcomes.)
Desired:	A solution to that problem. (Informally: How to reach a goal state, using the actions?)

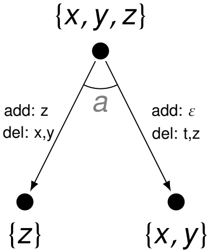
Given:

Non-deterministic planning problem $\mathcal{P} = (Var, A, s_0, G)$ with:

- Var, finite set of *state variables*.
  $S = 2^{Var}$ is the state space.
- A, finite set of *actions* $a = \langle pre(a), eff(a) \rangle$ and:
  - $pre(a) \subseteq Var$ and
  - $eff(a) = \{ \langle add_i, del_i \rangle \mid add_i, del_i \subseteq Var \text{ and } i \in \{1, \dots, n\} \}$.
  - Its application (if $pre(a) \subseteq s$) leads to:
    $app(s, a) = \{ (s \setminus del) \cup add \mid \langle add, del \rangle \in eff(a) \}$
- $s_0 \in S$, the *initial state*.
- $G \subseteq Var$, the *goal description*.
  A state $s \in S$ is a goal state iff $s \supseteq G$.

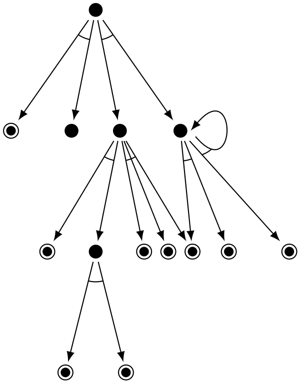
Let $s = \{x, y, z\} \in S$ be a state and $a \in A$ be an action with:

$a = \langle pre(a), eff(a)\rangle$ and
$pre(a) = \{x, y\} \subseteq s$,
$eff(a) = \{ \langle\{z\}, \{x, y\}\rangle, \langle\emptyset, \{t, z\}\rangle \}$.



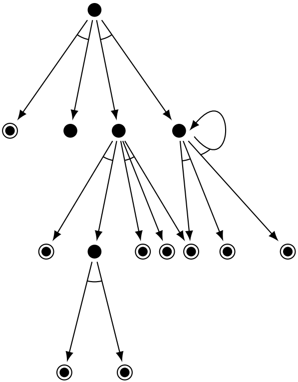
Desired:

Strong plan. (Success, regardless of non-deterministic outcome.)



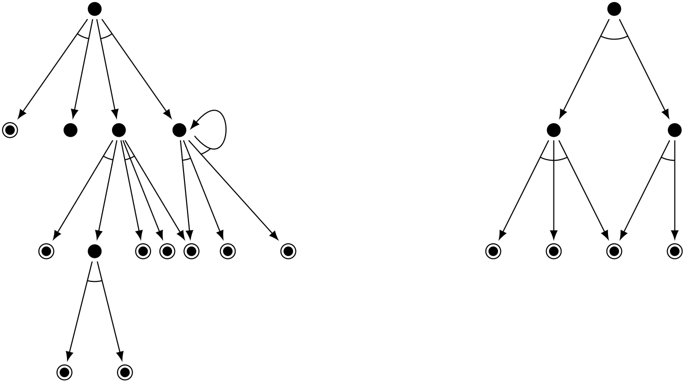
Desired:

Strong plan. (Success, regardless of non-deterministic outcome.)



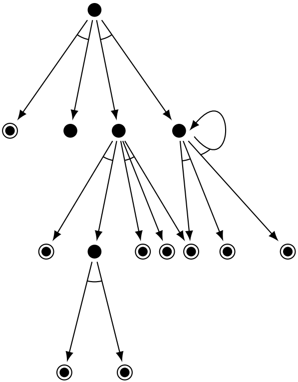
Desired:

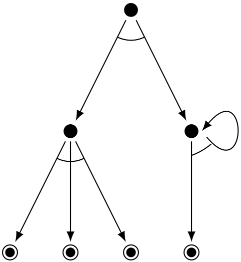
Strong plan. (Success, regardless of non-deterministic outcome.)



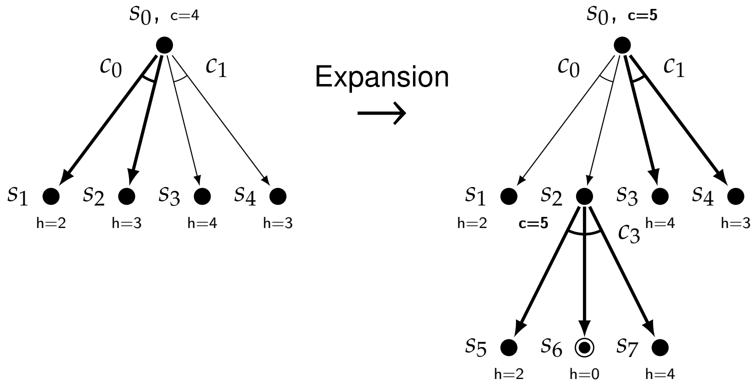
Desired:

Strong plan. (Success, regardless of non-deterministic outcome.)

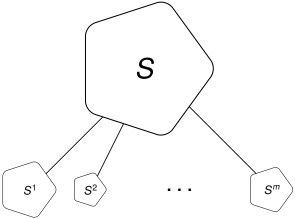




Search Algorithm, modification of AO*



Use abstraction to simplify the problem:



Map the search space S to abstract search spaces S^i with $|S^i| \ll |S|$.

Compute $h(s), s \in S,$ on basis of all $h^i(s^i)$.
Calculation of the h^i is done *before* the search.

Idea: Disregard some (or rather *most of the*) state variables.

The *abstraction* $\mathcal{P}^i = (Var^i, A^i, s_0^i, G^i)$ is the planning problem $\mathcal{P}$, restricted to the pattern $P_i \subseteq Var$:

- $Var^i := Var \cap P_i = P_i$,
- For $var \subseteq Var$ let $var^i := var \cap P_i$. Then:
  $a^i := \langle pre(a)^i, \{ \langle add^i, del^i \rangle \mid \langle add, del \rangle \in eff(a) \} \rangle$ for $a \in A$.
  Now, $A^i := \{ a^i \mid a \in A \}$.
- $s_0^i := s_0 \cap P_i$
- $G^i := G \cap P_i$.

Recall:

- A *pattern* is a set of state variables $P_i \subseteq Var$.
  Then, a *pattern collection* P is a set of patterns.
- Compute $h(s), s \in S$, on basis of all $h^i(s^i), P_i \in P$, P finite pattern collection, i.e. set of patterns.

How to calculate those $h^i(s^i), s^i \in S^i$?

$h^i(s^i)$ is the *true* cost value *cost** of the planning problem $\mathcal{P}^i$.
Calcuation is done by a complete exhaustive search.
(Thus, S^i and therefore P_i have to be small!)

(*True* means: prefer shallow solution graphs.)

How to calculate $h(s), s \in S$?

By using additivity!
A pattern collection P is called *additive*, if for all states $s \in S$:
$\sum_{P_i \in P} h^i(s^i) \leq cost^*(s)$, i.e. if this sum is still admissible.

Known from classical planning:

Theorem (textual description)

If there is no action $a \in A$ that affects variables in more than one pattern from P, then P is additive.

How to calculate $h(s), s \in S$?

By using additivity!
A pattern collection P is called *additive*, if for all states $s \in S$: $\sum_{P_i \in P} h^i(s^i) \leq cost^*(s)$, i.e. if this sum is still admissible.

Known from classical planning:

Theorem (mathematical description)

If for all $a \in A$ and for all patterns $P_i \in P$ holds:

If $P_i \cap effvar(a) \neq \emptyset$, then $P_j \cap effvar(a) = \emptyset$ for all $P_j \in P$ with $P_j \neq P_i$, where $effvar(a) = \bigcup_{\langle add, del \rangle \in eff(a)} add \cup del$.

Then P is additive.

Additivity (Example)

$\mathcal{P} = (\{a, b, c, d, e\}, A, \{a\}, \{b, c, d, e\})$ with $A = \{a_1, \ldots, a_9\}$ and:

$a_1 = \langle\{a\}, \{\langle\{b\}, \{a\}\rangle, \langle\{c\}, \{a\}\rangle\}\rangle$

$a_2 = \langle\{b\}, \{\langle\{e\}, \emptyset\rangle, \langle\{d\}, \emptyset\rangle\}\rangle$

$a_3 = \langle\{c\}, \{\langle\{e\}, \emptyset\rangle, \langle\{d\}, \emptyset\rangle\}\rangle$

$a_4 = \langle\{b, d\}, \{\langle\{c\}, \emptyset\rangle\}\rangle$

$a_5 = \langle\{c, d\}, \{\langle\{b\}, \emptyset\rangle\}\rangle$

$a_6 = \langle\{b, e\}, \{\langle\{c\}, \emptyset\rangle\}\rangle$

$a_7 = \langle\{c, e\}, \{\langle\{b\}, \emptyset\rangle\}\rangle$

$a_8 = \langle\{b, c, d\}, \{\langle\{e\}, \emptyset\rangle\}\rangle$

$a_9 = \langle\{b, c, e\}, \{\langle\{d\}, \emptyset\rangle\}\rangle$

$\mathcal{P} = (\{a, b, c, d, e\}, A, \{a\}, \{b, c, d, e\})$ with $A = \{a_1, \ldots, a_9\}$ and:

$$a_1 = \langle\{a\}, \{\langle\{b\}, \{a\}\rangle, \langle\{c\}, \{a\}\rangle\}\rangle$$
$$a_2 = \langle\{b\}, \{\langle\{e\}, \emptyset\rangle, \langle\{d\}, \emptyset\rangle\}\rangle$$
$$a_3 = \langle\{c\}, \{\langle\{e\}, \emptyset\rangle, \langle\{d\}, \emptyset\rangle\}\rangle$$
$$a_4 = \langle\{b, d\}, \{\langle\{c\}, \emptyset\rangle\}\rangle$$
$$a_5 = \langle\{c, d\}, \{\langle\{b\}, \emptyset\rangle\}\rangle$$
$$a_6 = \langle\{b, e\}, \{\langle\{c\}, \emptyset\rangle\}\rangle$$
$$a_7 = \langle\{c, e\}, \{\langle\{b\}, \emptyset\rangle\}\rangle$$
$$a_8 = \langle\{b, c, d\}, \{\langle\{e\}, \emptyset\rangle\}\rangle$$
$$a_9 = \langle\{b, c, e\}, \{\langle\{d\}, \emptyset\rangle\}\rangle$$

Now, consider the pattern collection $P = \{\{a, b, c\}, \{d, e\}\}$.

Additivity (Example)

$\mathcal{P} = (\{a, b, c, d, e\}, A, \{a\}, \{b, c, d, e\})$ with $A = \{a_1, \ldots, a_9\}$ and:

$$a_1 = \langle \{a\}, \{\langle \{b\}, \{a\} \rangle, \langle \{c\}, \{a\} \rangle\} \rangle$$
$$a_2 = \langle \{b\}, \{\langle \{e\}, \emptyset \rangle, \langle \{d\}, \emptyset \rangle\} \rangle$$
$$a_3 = \langle \{c\}, \{\langle \{e\}, \emptyset \rangle, \langle \{d\}, \emptyset \rangle\} \rangle$$
$$a_4 = \langle \{b, d\}, \{\langle \{c\}, \emptyset \rangle\} \rangle$$
$$a_5 = \langle \{c, d\}, \{\langle \{b\}, \emptyset \rangle\} \rangle$$
$$a_6 = \langle \{b, e\}, \{\langle \{c\}, \emptyset \rangle\} \rangle$$
$$a_7 = \langle \{c, e\}, \{\langle \{b\}, \emptyset \rangle\} \rangle$$
$$a_8 = \langle \{b, c, d\}, \{\langle \{e\}, \emptyset \rangle\} \rangle$$
$$a_9 = \langle \{b, c, e\}, \{\langle \{d\}, \emptyset \rangle\} \rangle$$

Now, consider the pattern collection $P = \{\{a, b, c\}, \{d, e\}\}$.

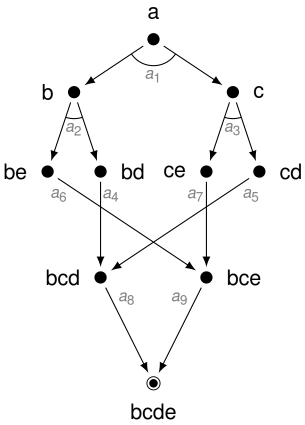
Additivity (Example)

$\mathcal{P} = (\{a, b, c, d, e\}, A, \{a\}, \{b, c, d, e\})$ with $A = \{a_1, \ldots, a_9\}$ and:

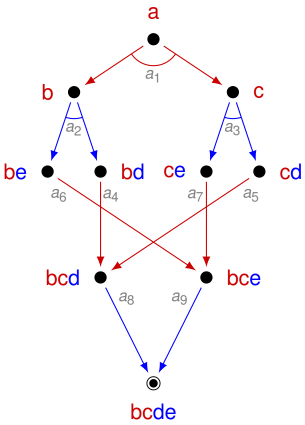
$$a_1 = \langle\{a\}, \{\langle\{b\}, \{a\}\rangle, \langle\{c\}, \{a\}\rangle\}\rangle$$
$$a_2 = \langle\{b\}, \{\langle\{e\}, \emptyset\rangle, \langle\{d\}, \emptyset\rangle\}\rangle$$
$$a_3 = \langle\{c\}, \{\langle\{e\}, \emptyset\rangle, \langle\{d\}, \emptyset\rangle\}\rangle$$
$$a_4 = \langle\{b, d\}, \{\langle\{c\}, \emptyset\rangle\}\rangle$$
$$a_5 = \langle\{c, d\}, \{\langle\{b\}, \emptyset\rangle\}\rangle$$
$$a_6 = \langle\{b, e\}, \{\langle\{c\}, \emptyset\rangle\}\rangle$$
$$a_7 = \langle\{c, e\}, \{\langle\{b\}, \emptyset\rangle\}\rangle$$
$$a_8 = \langle\{b, c, d\}, \{\langle\{e\}, \emptyset\rangle\}\rangle$$
$$a_9 = \langle\{b, c, e\}, \{\langle\{d\}, \emptyset\rangle\}\rangle$$

Now, consider the pattern collection $P = \{\{a, b, c\}, \{d, e\}\}$.
Only the effect variables matter!

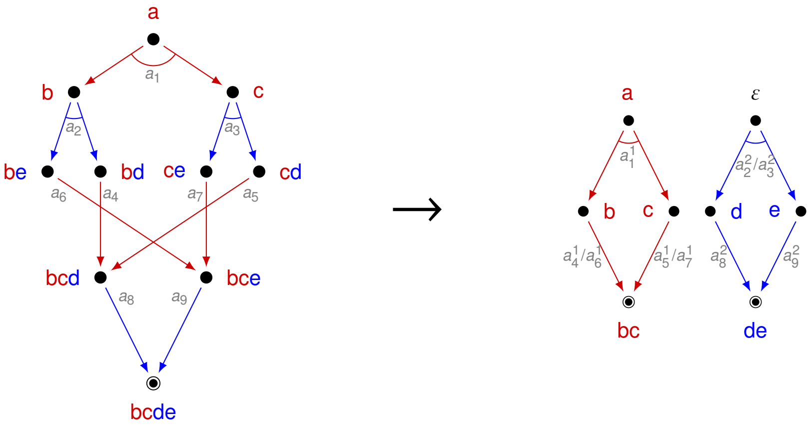
Additivity (Example, cont'd)



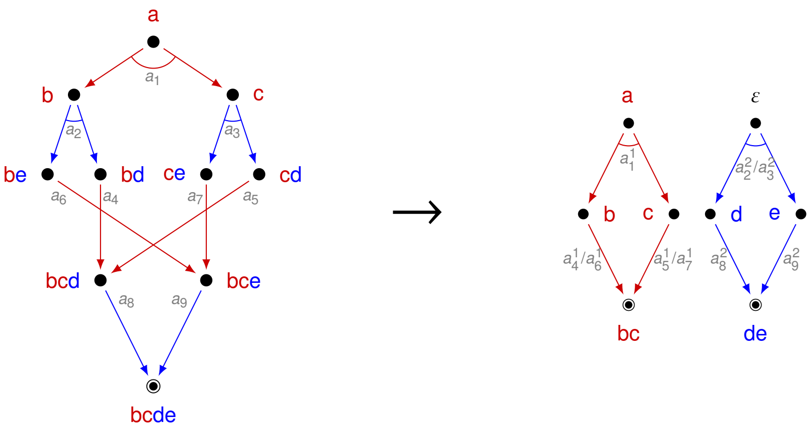
Additivity (Example, cont'd)



Additivity (Example, cont'd)



Additivity (Example, cont'd)



Example:

$$h(\{a\}) = h^1(\{a\}^1) + h^2(\{a\}^2) = h^1(\{a\}) + h^2(\emptyset) = 2 + 2 = 4 = cost^*(\{a\}).$$

Let $\mathcal{M}$ be a set of additive pattern collections.

$h^{\mathcal{M}}(s) := \max_{P \in \mathcal{M}} \sum_{P_i \in P} h^i(s^i)$.

$h^{\mathcal{M}}$ (and in particular, every single h^i) is admissible.

How to find $\mathcal{M}$?

Current research. (Here: still domain-dependent by hand.)

Compared Systems

Encoded two domains and compared:

- Our planner with the heuristic of FF.
- Our planner with the presented pattern database heuristics.
- GAMER.

Important differences between GAMER and our system:

- Optimal solutions vs. suboptimal solutions.
- Regression vs. progression.

Results

- Pattern database heuristic quality is problem dependent:

Domain 1:	Pattern database heuristics about 25% more node expansions than FF heuristic.
Domain 2:	Pattern database heuristics calculate true (perfect) cost value (as opposed to the FF heuristic).

- Calculation time of pattern database heuristic is much smaller than the FF heuristic's. Thus, more problems could be solved.
- Progression with heuristic search seems promising approach. (Note: No comparison to sub-optimal planner, yet.)

- Presented fomalization for domain-independent pattern database heuristics in non-deterministic planning.
- Generalization of additivity criterion.
- Benchmarks look promising.

Future Work

- Automatic pattern selection.
- Strong plans → strong *cyclic* plans.
  - Search algorithm, LAO*.
  - Pattern database heuristics: Admissibility/Additivity?
- Multi-valued state variables.

Thank you!