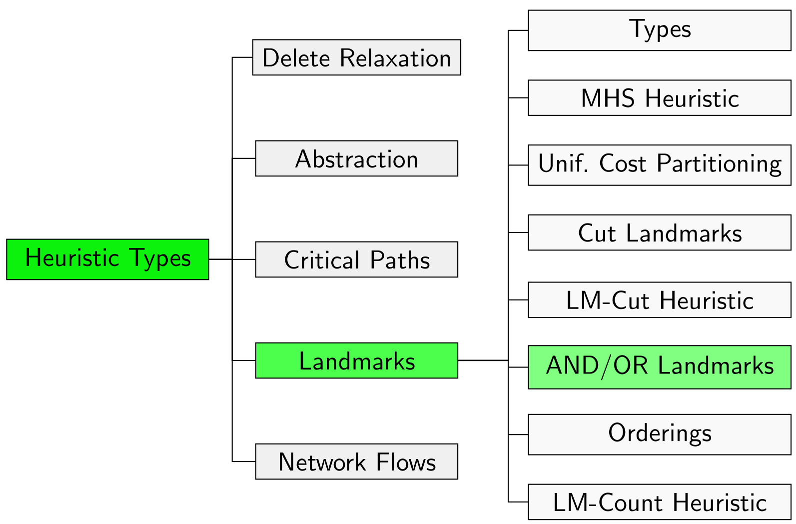
Planning and Optimization
E5. Landmarks: And/Or Landmarks

Malte Helmert and Gabriele Röger

Universität Basel

November 27, 2017

Content of this Course: Landmarks



Reminder

Definition (Disjunctive Action Landmark)

Let s be a state of planning task $\Pi = \langle V, I, O, \gamma \rangle$.

A disjunctive action landmark for s is a set of operators $L \subseteq O$ such that every label path from s to a goal state contains an operator from L.

Definition (Formula and Fact Landmark)

Let s be a state of planning task $\Pi = \langle V, I, O, \gamma \rangle$.

A formula landmark for s is a formula λ over V such that every state path from s to a goal state contains a state s' with $s' \models \lambda$.

If $\lambda \in V$ then λ is a fact landmark.

E5.1 Landmarks from RTGs

Incidental Landmarks

Example (Incidental Landmarks)

$\Pi = \langle \{a, b, c, d, e, f\}, \{a, b, e\}, \{o_1, o_2\}, \{e, f\} \rangle$ with

$$o_1 = \langle \{a\}, \{c, d, e\}, \{b\}, 1 \rangle, \text{ and}$$
$$o_2 = \langle \{d, e\}, \{f\}, \{a, b, c, d\}, 1 \rangle.$$

Single plan $\langle o_1, o_2 \rangle$ with state path $\{a, b, e\}, \{a, c, d, e\}, \{e, f\}$.

- All variables are fact landmarks for the initial state.
- Variable b is initially true but irrelevant for the plan.
- Variable c gets true as "side effect" of o_1 but it is not necessary for the goal or to make an operator applicable.

Causal Landmarks (1)

Definition (Causal Formula Landmark)

Let $\Pi = \langle V, I, O, \gamma \rangle$ be a planning task.

A formula λ over V is a causal formula landmark for I if $\gamma \models \lambda$ or if for all plans $\pi = \langle o_1, \dots, o_n \rangle$ there is an o_i with $pre(o_i) \models \lambda$.

Causal Landmarks (2)

Special case: Fact Landmark for STRIPS task

Definition (Causal Fact Landmark)

Let $\Pi = \langle V, I, O, G \rangle$ be a STRIPS planning task (in set representation).

A variable $v \in V$ is a causal fact landmark for I if $v \in G$ or if for all plans $\pi = \langle o_1, \dots, o_n \rangle$ there is an o_i with $v \in pre(o_i)$.

Causal Landmarks: Example

Example (Causal Landmarks)

$\Pi = \langle \{a, b, c, d, e, f\}, \{a, b, e\}, \{o_1, o_2\}, \{e, f\} \rangle$ with

$$o_1 = \langle \{a\}, \{c, d, e\}, \{b\}, 1 \rangle, \text{ and}$$
$$o_2 = \langle \{d, e\}, \{f\}, \{a, b, c, d\}, 1 \rangle.$$

Single plan $\langle o_1, o_2 \rangle$ with state path $\{a, b, e\}, \{a, c, d, e\}, \{e, f\}$.

- All variables are fact landmarks for the initial state.
- Only a, d, e and f are causal landmarks.

What We Are Doing Next

- Causal landmarks are the desirable landmarks.
- For STRIPS, we can use (a simpler version of) RTGs to compute them.
- We will define landmarks of AND/OR graphs, ...
- and show how they can be computed.
- Afterwards we establish that these are landmarks of the planning task.

Simplified Relaxed Task Graph

Definition

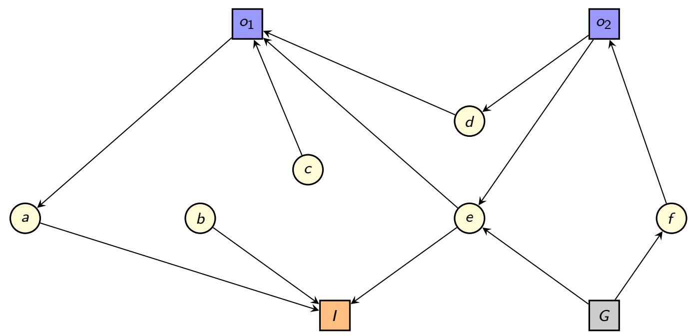
For a STRIPS planning task $\langle V, I, O, G \rangle$ (in set representation), the simplified relaxed task graph $sRTG(\Pi^+)$ is the AND/OR graph $\langle V_{\text{and}}, V_{\text{or}}, E \rangle$ with

- AND nodes $V_{\text{and}} = \{n_o \mid o \in O\} \cup \{v_I, v_G\}$,
- OR nodes $V_{\text{or}} = \{n_v \mid v \in V\}$, and
- $E = \{\langle n_a, n_o \rangle \mid o \in O, a \in add(o)\} \cup$
  $\{\langle n_o, n_p \rangle \mid o \in O, p \in pre(o)\} \cup$
  $\{\langle n_v, n_I \rangle \mid v \in I\} \cup$
  $\{\langle n_G, n_v \rangle \mid v \in G\}$

Simplified RTG: Example

$\Pi = \langle \{a, b, c, d, e, f\}, \{a, b, e\}, \{o_1, o_2\}, \{e, f\} \rangle$ with

$$o_1 = \langle \{a\}, \{c, d, e\}, \{b\}, 1 \rangle, \text{ and}$$
$$o_2 = \langle \{d, e\}, \{f\}, \{a, b, c, d\}, 1 \rangle.$$



Justification

Definition (Justification)

Let $G = \langle V_{\text{and}}, V_{\text{or}}, E\rangle$ be an AND/OR graph.

A subgraph $J = \langle V^J, E^J\rangle$ with $V^J \subseteq V_{\text{and}} \cup V_{\text{or}}$ and $E^J \subseteq E$ justifies $n_\star \in V_{\text{and}} \cup V_{\text{or}}$ iff

- $n_\star \in V^J$,
- $\forall n \in V^J \cap V_{\text{and}} : \forall \langle n, n'\rangle \in E : n' \in V^J$ and $\langle n, n'\rangle \in E^J$
- $\forall n \in V^J \cap V_{\text{or}} : \exists \langle n, n'\rangle \in E : n' \in V^J$ and $\langle n, n'\rangle \in E^J$, and
- J is acyclic.

"Proves" that $n_\star$ is forced true.

Landmarks in AND/OR Graphs

Definition (Landmarks in AND/OR Graphs)

Let $G = \langle V_{\text{and}}, V_{\text{or}}, E\rangle$ be an AND/OR graph. A node n is a landmark for reaching $n_\star \in V_{\text{and}} \cup V_{\text{or}}$ if $n \in V^J$ for all justifications J for $n_\star$.

But: exponential number of possible justifications

Characterizing Equation System

Theorem

Let $G = \langle V_{\text{and}}, V_{\text{or}}, E\rangle$ be an AND/OR graph. Consider the following system of equations:

$$LM(n) = \{n\} \cup \bigcap_{\langle n,n'\rangle \in E} LM(n') \qquad n \in V_{\text{or}}$$

$$LM(n) = \{n\} \cup \bigcup_{\langle n,n'\rangle \in E} LM(n') \qquad n \in V_{\text{and}}$$

The equation system has a unique maximal solution (maximal with regard to set inclusion), and for this solution it holds that

$n' \in LM(n)$ iff n' is a landmark for reaching n in G.

Computation of Maximal Solution

Theorem

Let $G = \langle V_{\text{and}}, V_{\text{or}}, E\rangle$ be an AND/OR graph. Consider the following system of equations:

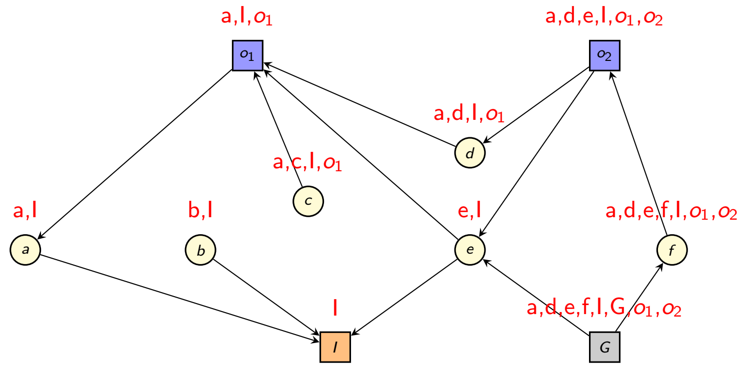
$$LM(n) = \{n\} \cup \bigcap_{\langle n,n'\rangle \in E} LM(n') \qquad n \in V_{\text{or}}$$

$$LM(n) = \{n\} \cup \bigcup_{\langle n,n'\rangle \in E} LM(n') \qquad n \in V_{\text{and}}$$

The equation system has a unique maximal solution (maximal with regard to set inclusion).

Computation: Initialize landmark sets as $LM(n) = V_{\text{and}} \cup V_{\text{or}}$ and apply equations as update rules until fixpoint.

Computation: Example



(cf. screen version of slides for step-wise computation)

Relation to Planning Task Landmarks

Theorem

Let $\Pi = \langle V, I, O, G\rangle$ be a STRIPS planning task and let $\mathcal{L}$ be the set of landmarks for reaching n_G in $sRTG(\Pi^+)$.

The set $\{v \in V \mid n_v \in \mathcal{L}\}$ is exactly the set of causal fact landmarks for I in Π^+.

For operators $o \in O$, if $n_o \in \mathcal{L}$ then $\{o\}$ is a disjunctive action landmark for I in Π^+. There are no other disjunctive action landmarks of size 1.

(Proofs omitted.)

Example

Example

$\Pi = \langle\{a, b, c, d, e, f\}, \{a, b, e\}, \{o_1, o_2\}, \{e, f\}\rangle$ with

$$o_1 = \langle\{a\}, \{c, d, e\}, \{b\}, 1\rangle, \text{ and}$$
$$o_2 = \langle\{d, e\}, \{f\}, \{a, b, c, d\}, 1\rangle.$$

- $LM(n_G) = \{a, d, e, f, I, G, o_1, o_2\}$
- a, d, e, and f are causal fact landmarks of Π^+.
- They are the only causal fact landmarks of Π^+.
- $\{o_1\}$ and $\{o_2\}$ are disjunctive action landmarks of Π^+.

(Some) Landmarks of Π^+ Are Landmarks of Π

Theorem

Let Π be a STRIPS planning task.

All fact landmarks of Π^+ are fact landmarks of Π and all disjunctive action landmarks of Π^+ are disjunctive action landmarks of Π.

Proof.

Let L be a disjunctive action landmark of Π^+ and π be a plan for Π. Then π is also a plan for Π^+ and, thus, π contains an operator from L.

Let f be a fact landmark of Π^+. If f is already true in the initial state, then it is also a landmark of Π. Otherwise, every plan for Π^+ contains an operator that adds f and the set of all these operators is a disjunctive action landmark of Π^+. Therefore, also each plan of Π contains such an operator, making f a fact landmark of Π. □

Not All Landmarks of Π^+ are Landmarks of Π

Example

Consider STRIPS task $\langle\{a, b, c\}, \emptyset, \{o_1, o_2\}, \{c\}\rangle$ with $o_1 = \langle\{\}, \{a\}, \{\}, 1\rangle$ and $o_2 = \langle\{a\}, \{c\}, \{a\}, 1\rangle$.

$a \wedge c$ is a formula landmark of Π^+ but not of Π.

E5.2 Landmarks from Π^m

Reminder: Π^m Compilation

Definition (Π^m)

Let $\Pi = \langle V, I, O, G\rangle$ be a STRIPS planning task.

For $m \in \mathbb{N}_1$, the task Π^m is the STRIPS planning task $\langle V^m, I^m, O^m, G^m\rangle$, where
$O^m = \{a_{o,S} \mid o \in O, S \subseteq V, |S| < m, S \cap (add(o) \cup del(o)) = \emptyset\}$
with

- $pre(a_{o,S}) = (pre(o) \cup S)^m$
- $add(a_{o,S}) = \{v_Y \mid Y \subseteq add(o) \cup S, |Y| \leq m, Y \cap add(o) \neq \emptyset\}$
- $del(a_{o,S}) = \emptyset$
- $cost(a_{o,S}) = cost(o)$

Landmarks from the Π^m Compilation (1)

Idea:

- Π^m is delete-free, so we can compute all causal (meta-)fact landmarks from the AND/OR graph.
- These landmarks correspond to formula landmarks of the original problem.

Landmarks from the Π^m Compilation (2)

Theorem

Let $\Pi = \langle V, I, O, G \rangle$ be a STRIPS planning task. If meta-variable v_S is a fact landmark for I^m in Π^m then $\bigwedge_{v \in S} v$ is a formula landmark for I in Π.

(Proof ommited.)

Π^m Landmarks: Example

Consider again our running example:

Example

$\Pi = \langle \{a, b, c, d, e, f\}, \{a, b, e\}, \{o_1, o_2\}, \{e, f\} \rangle$ with

$$o_1 = \langle \{a\}, \{c, d, e\}, \{b\}, 1 \rangle, \text{ and}$$
$$o_2 = \langle \{d, e\}, \{f\}, \{a, b, c, d\}, 1 \rangle.$$

Meta-variable $v_{\{d,e\}}$ is a causal fact landmark for I^2 in Π^2, so $d \wedge e$ is a causal formula landmark for Π.

Landmarks from the Π^m Compilation (3)

Theorem

Let $\Pi = \langle V, I, O, G \rangle$ be a STRIPS planning task. For $m \in \mathbb{N}_1$ let $\mathcal{L}^m = \{\wedge_{v \in C} v \mid C \subseteq V, v_C \text{ is a causal fact landmark of } \Pi^m\}$ be the set of formula landmarks derived from Π^m.

Let λ be a conjunction over V that is a causal formula landmark of Π. For sufficiently large m, $\mathcal{L}^m$ contains λ' with $\lambda' \equiv \lambda$.

(Proof omitted.)

$\rightsquigarrow$ can find all causal conjunctive formula landmarks

Π^m Landmarks: Discussion

- With the Π^m compilation, we can find causal fact landmarks of Π that are not causal fact landmarks of Π^+.
- In addition we can find conjunctive formula landmarks.
- The approach takes to some extent delete effects into account.
- However, the approach takes exponential time in m.
- Even for small m, the additional cost for computing the landmarks often outweights the time saved from better heuristic guidance.

E5.3 Summary

Summary

- We can efficiently compute all causal fact landmarks of a delete-free task from the (simplified) RTG.
- Fact landmarks of the delete relaxed task are also landmarks of the original task.
- We can use the Π^m compilation to find more landmarks.