Planning and Optimization

G2. Landmarks: RTG Landmarks

Malte Helmert and Gabriele Röger

Universität Basel

November 29, 2023

Planning and Optimization

November 29, 2023 — G2. Landmarks: RTG Landmarks

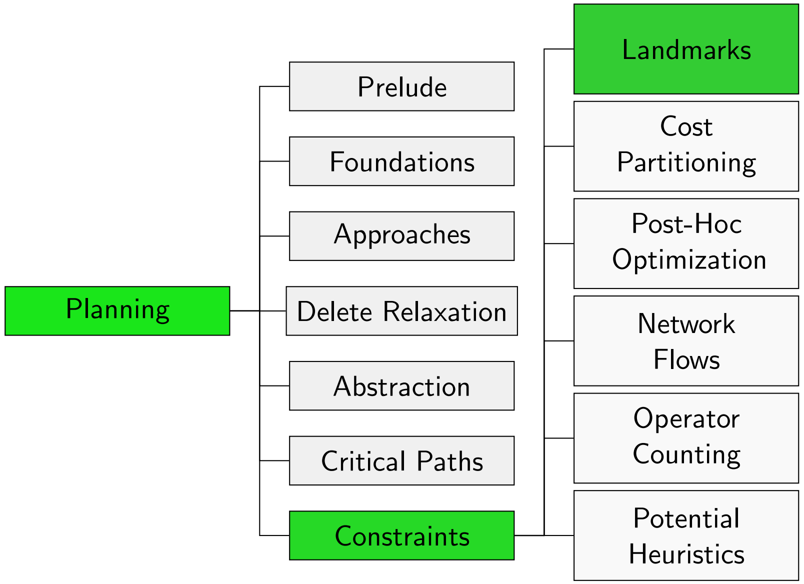
G2.1 Landmarks

G2.2 Landmarks from RTGs

G2.3 Landmarks from Π^m

G2.4 Summary

Content of this Course



G2.1 Landmarks

Landmarks

Basic Idea: Something that must happen in every solution

For example

- some operator must be applied (action landmark)
- some atomic proposition must hold (fact landmark)
- some formula must be true (formula landmark)

→ Derive heuristic estimate from this kind of information.

We mostly consider fact and disjunctive action landmarks.

Reminder: Terminology

Consider sequence of transitions $s^0 \xrightarrow{\ell_1} s^1, \dots, s^{n-1} \xrightarrow{\ell_n} s^n$ such that $s^0 = s$ and $s^n = s'$.

- $s^0, \dots, s^n$ is called (state) path from s to s'
- $\ell_1, \dots, \ell_n$ is called (label) path from s to s'

Disjunctive Action Landmarks

Definition (Disjunctive Action Landmark)

Let s be a state of a propositional or FDR 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.
The cost of landmark L is $cost(L) = \min_{o \in L} cost(o)$.

If we talk about landmarks for the initial state, we omit "for I".

Fact and Formula Landmarks

Definition (Formula and Fact Landmark)

Let s be a state of a propositional or FDR 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 λ is an atomic proposition then λ is a fact landmark.

If we talk about landmarks for the initial state, we omit "for I".

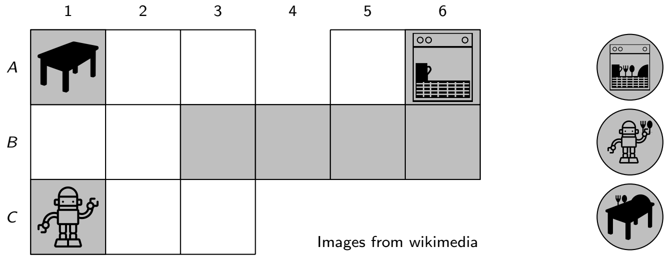
Landmarks: Example

Example

Consider a FDR planning task $\langle V, I, O, \gamma \rangle$ with

- $V = \{\textit{robot-at}, \textit{dishes-at}\}$ with
  - $\mathrm{dom}(\textit{robot-at}) = \{\mathrm{A1}, \dots, \mathrm{C3}, \mathrm{B4}, \mathrm{A5}, \dots, \mathrm{B6}\}$
  - $\mathrm{dom}(\textit{dishes-at}) = \{\mathrm{Table}, \mathrm{Robot}, \mathrm{Dishwasher}\}$
- $I = \{\textit{robot-at} \mapsto \mathrm{C1}, \textit{dishes-at} \mapsto \mathrm{Table}\}$
- operators
  - move-x-y to move from cell x to adjacent cell y
  - pickup dishes, and
  - load dishes into the dishwasher.
- $\gamma = (\textit{robot-at} = B6) \wedge (\textit{dishes-at} = \mathrm{Dishwasher})$

Fact and Formula Landmarks: Example



Images from wikimedia

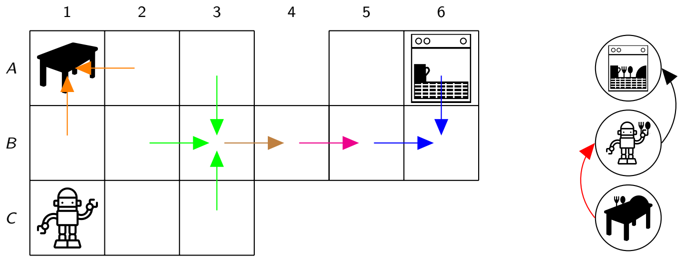
Each fact in gray is a fact landmark:

- `robot-at` $= x$ for $x \in \{\mathrm{A1}, \mathrm{A6}, \mathrm{B3}, \mathrm{B4}, \mathrm{B5}, \mathrm{B6}, \mathrm{C1}\}$
- `dishes-at` $= x$ for $x \in \{\mathrm{Dishwasher}, \mathrm{Robot}, \mathrm{Table}\}$

Formula landmarks:

- `dishes-at` $= \mathrm{Robot} \wedge$ `robot-at` $= \mathrm{B4}$
- `robot-at` $= \mathrm{B1} \vee$ `robot-at` $= \mathrm{A2}$

Disjunctive Action Landmarks: Example



Actions of same color form disjunctive action landmark:

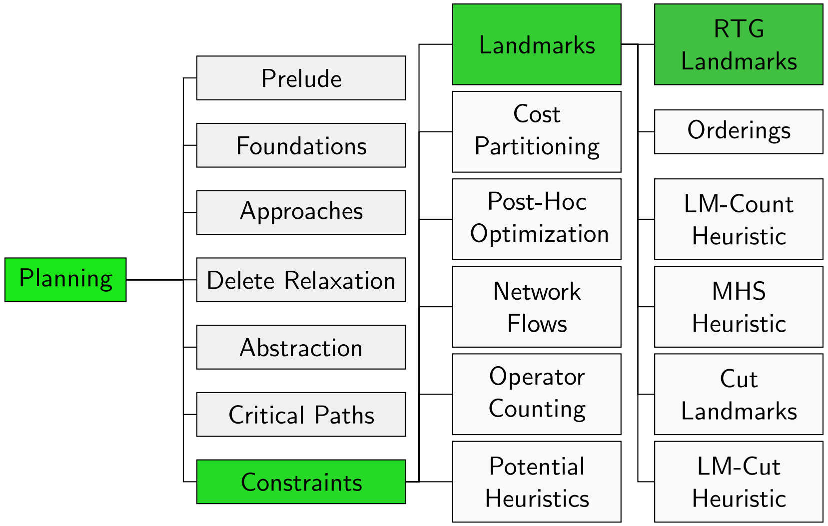
- {pickup}
- {load}
- {move-B3-B4}
- {move-B4-B5}
- {move-A6-B6, move-B5-B6}
- {move-A3-B3, move-B2-B3, move-C3-B3}
- {move-B1-A1, move-A2-A1}
- ...

Remarks

- Not every landmark is informative. Some examples:
  - The set of all operators is a disjunctive action landmark unless the initial state is already a goal state.
  - Every variable that is initially true is a fact landmark.
  - The goal formula is a formula landmark.
- Deciding whether a given atomic proposition is a fact landmark is as hard as the plan existence problem.
- Deciding whether a given operator set is a disjunctive action landmark is as hard as the plan existence problem.
- Every fact landmark v that is initially false induces a disjunctive action landmark consisting of all operators that possibly make v true.

G2.2 Landmarks from RTGs

Content of this Course



Planning

Prelude

Foundations

Approaches

Delete Relaxation

Abstraction

Critical Paths

Constraints

Landmarks

Cost Partitioning

Post-Hoc Optimization

Network Flows

Operator Counting

Potential Heuristics

RTG Landmarks

Orderings

LM-Count Heuristic

MHS Heuristic

Cut Landmarks

LM-Cut Heuristic

Computing Landmarks

How can we come up with landmarks?

Most landmarks are derived from the relaxed task graph:

- RHW landmarks: Richter, Helmert & Westphal. Landmarks Revisited. (AAAI 2008)
- LM-Cut: Helmert & Domshlak. Landmarks, Critical Paths and Abstractions: What's the Difference Anyway? (ICAPS 2009)
- h^m landmarks: Keyder, Richter & Helmert: Sound and Complete Landmarks for And/Or Graphs (ECAI 2010)

We will now discuss h^m landmarks restricted to to STRIPS planning tasks, starting with $m = 1$.

Incidental Landmarks: Example

Example (Incidental Landmarks)

Consider a STRIPS planning task $\langle V, I, \{o_1, o_2\}, G\rangle$ with

$$\begin{aligned} V &= \{a, b, c, d, e, f\}, \\ I &= \{a \mapsto \mathbf{T}, b \mapsto \mathbf{T}, c \mapsto \mathbf{F}, d \mapsto \mathbf{F}, e \mapsto \mathbf{T}, f \mapsto \mathbf{F}\}, \\ o_1 &= \langle \{a\}, \{c, d, e\}, \{b\}\rangle, \\ o_2 &= \langle \{d, e\}, \{f\}, \{a\}\rangle, \text{ and} \\ G &= \{e, f\}. \end{aligned}$$

Single solution: $\langle o_1, o_2\rangle$

- All variables are fact landmarks.
- 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 propositional or FDR 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)

Consider a STRIPS planning task $\langle V, I, \{o_1, o_2\}, G\rangle$ with

$$\begin{aligned}
V &= \{a, b, c, d, e, f\},\\
I &= \{a \mapsto \mathbf{T}, b \mapsto \mathbf{T}, c \mapsto \mathbf{F}, d \mapsto \mathbf{F}, e \mapsto \mathbf{T}, f \mapsto \mathbf{F}\},\\
o_1 &= \langle \{a\}, \{c, d, e\}, \{b\}\rangle,\\
o_2 &= \langle \{d, e\}, \{f\}, \{a\}\rangle, \text{ and}\\
G &= \{e, f\}.
\end{aligned}$$

Single solution: $\langle o_1, o_2\rangle$

- 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.
- We can use the simplified version of RTGs for STRIPS to compute causal landmarks for STRIPS planning tasks.
- 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.

Reminder: Simplified Relaxed Task Graph

Definition

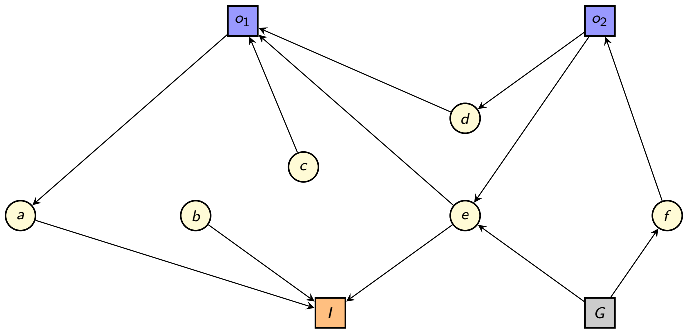
For a STRIPS planning task $\Pi = \langle V, I, O, G\rangle$ (in set representation), the simplified relaxed task graph $sRTG(\Pi^+)$ is the AND/OR graph $\langle N_{\text{and}} \cup N_{\text{or}}, A, type\rangle$ with

- $N_{\text{and}} = \{n_o \mid o \in O\} \cup \{v_I, v_G\}$ with $type(n) = \wedge$ for all $n \in N_{\text{and}}$,
- $N_{\text{or}} = \{n_v \mid v \in V\}$ with $type(n) = \vee$ for all $n \in N_{\text{or}}$, and
- $A = \{\langle n_a, n_o\rangle \mid o \in O, a \in add(o)\} \cup$ $\{\langle n_o, n_p\rangle \mid o \in O, p \in pre(o)\} \cup$ $\{\langle n_v, n_I\rangle \mid v \in I\} \cup$ $\{\langle n_G, n_v\rangle \mid v \in G\}$

Like RTG but without extra nodes to support arbitrary conditions.

Simplified RTG: Example

The simplified RTG for our example task is:



Justification

Definition (Justification)

Let $G = \langle N, A, type\rangle$ be an AND/OR graph.

A subgraph $J = \langle N^J, A^J, type^J\rangle$ with $N^J \subseteq N$ and $A^J \subseteq A$ and $type^J = type|_{N^J}$ justifies $n_\star \in N$ iff

- $n_\star \in N^J$,
- $\forall n \in N^J$ with $type(n) = \wedge$: $\forall \langle n, n'\rangle \in A : n' \in N^J$ and $\langle n, n'\rangle \in A^J$
- $\forall n \in N^J$ with $type(n) = \vee$: $\exists \langle n, n'\rangle \in A : n' \in N^J$ and $\langle n, n'\rangle \in A^J$, and
- J is acyclic.

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

Landmarks in AND/OR Graphs

Definition (Landmarks in AND/OR Graphs)

Let $G = \langle N, A, type\rangle$ be an AND/OR graph.

A node $n \in N$ is a landmark for reaching $n_\star \in N$ if $n \in V^J$ for all justifications J for $n_\star$.

But: exponential number of possible justifications

Characterizing Equation System

Theorem

Let $G = \langle N, A, type\rangle$ be an AND/OR graph. Consider the following system of equations:

$$LM(n) = \{n\} \cup \bigcap_{\langle n,n'\rangle \in A} LM(n') \quad type(n) = \vee$$

$$LM(n) = \{n\} \cup \bigcup_{\langle n,n'\rangle \in A} LM(n') \quad type(n) = \wedge$$

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 $G = \langle N, A, type\rangle$ be an AND/OR graph. Consider the following system of equations:

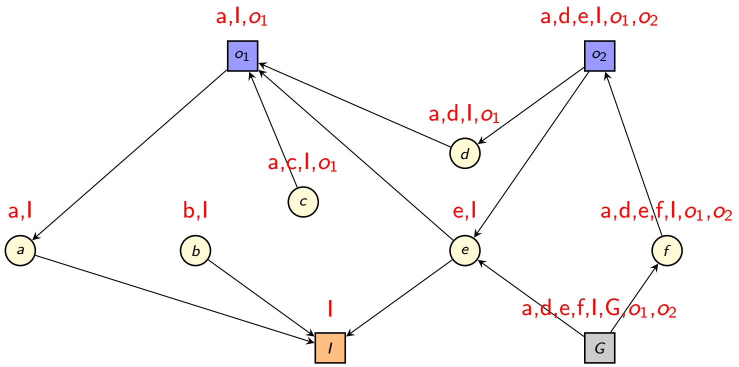
$$LM(n) = \{n\} \cup \bigcap_{\langle n,n'\rangle \in A} LM(n') \quad type(n) = \vee$$

$$LM(n) = \{n\} \cup \bigcup_{\langle n,n'\rangle \in A} LM(n') \quad type(n) = \wedge$$

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

Computation: Initialize landmark sets as $LM(n) = N$ 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 $\Pi = \langle V, I, O, \gamma\rangle$ 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 in Π^+.

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

(Proofs omitted.)

Computed RTG Landmarks: Example

Example (Computed RTG Landmarks)

Consider a STRIPS planning task $\langle V, I, \{o_1, o_2\}, G\rangle$ with

$$\begin{aligned} V &= \{a, b, c, d, e, f\},\\ I &= \{a \mapsto \mathbf{T}, b \mapsto \mathbf{T}, c \mapsto \mathbf{F}, d \mapsto \mathbf{F}, e \mapsto \mathbf{T}, f \mapsto \mathbf{F}\},\\ o_1 &= \langle \{a\}, \{c, d, e\}, \{b\}\rangle,\\ o_2 &= \langle \{d, e\}, \{f\}, \{a\}\rangle, \text{ and}\\ G &= \{e, f\}. \end{aligned}$$

- $LM(n_G) = \{a, d, e, f, I, G, o_1, o_2\}$
- a, d, e, and f are 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.

G2.3 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

STRIPS planning task $\Pi = \langle V, I, \{o_1, o_2\}, G \rangle$ with

$$\begin{aligned} V &= \{a, b, c, d, e, f\}, \\ I &= \{a \mapsto \mathbf{T}, b \mapsto \mathbf{T}, c \mapsto \mathbf{F}, d \mapsto \mathbf{F}, e \mapsto \mathbf{T}, f \mapsto \mathbf{F}\}, \\ o_1 &= \langle \{a\}, \{c, d, e\}, \{b\} \rangle, \\ o_2 &= \langle \{d, e\}, \{f\}, \{a\} \rangle, \text{ and} \\ G &= \{e, f\}. \end{aligned}$$

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 $\mathcal{L}^m = \{\bigwedge_{v\in C} v \mid C \subseteq V, v_C$ 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.

G2.4 Summary

Summary

- Fact landmark: atomic proposition that is true in each state path to a goal
- Disjunctive action landmark: set L of operators such that every plan uses some operator from L
- We can efficiently compute all causal fact landmarks of a delete-free STRIIPS 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.