

Planning and Optimization

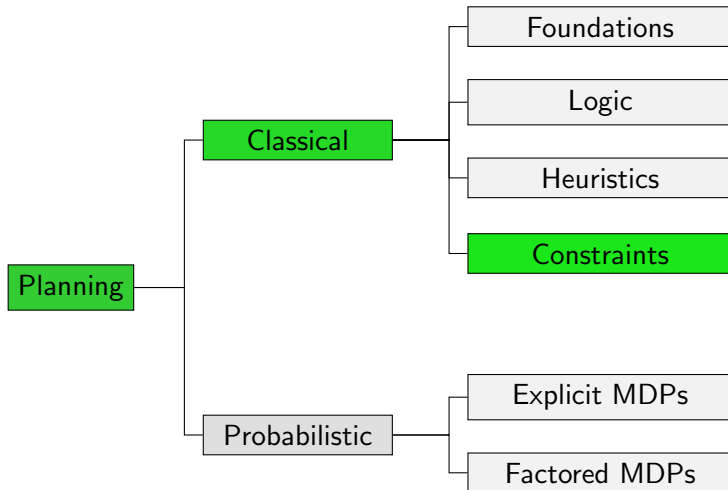
E2. Landmarks: RTG Landmarks & MHS Heuristic

Malte Helmert and Thomas Keller

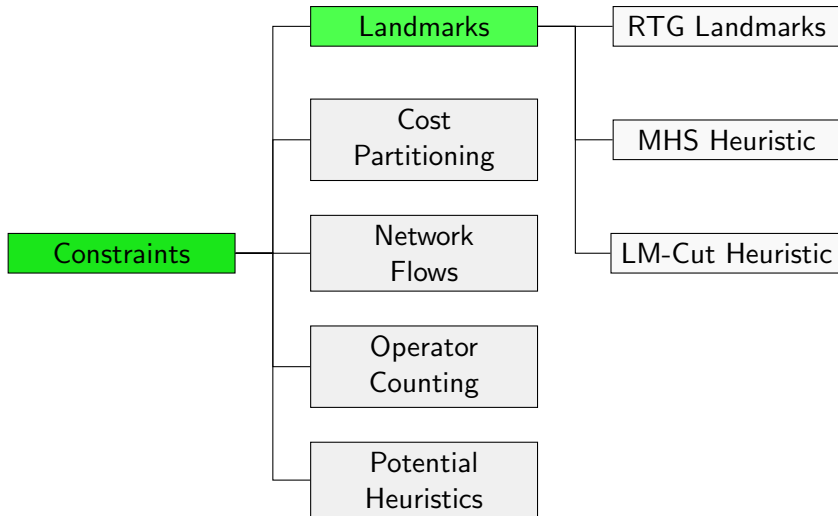
Universität Basel

November 11, 2019

Content of this Course



Content of this Course: Constraints



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.

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 only consider fact and disjunctive action landmarks.

Definition

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 .

The **cost** of landmark L is $cost(L) = \min_{o \in L} cost(o)$.

Definition (Fact Landmark)

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

An atomic proposition $v = d$ for $v \in V$ and $d \in \text{dom}(v)$ is a **fact landmark** for s if every state path from s to a goal state contains a state s' with $s'(v) = d$.

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

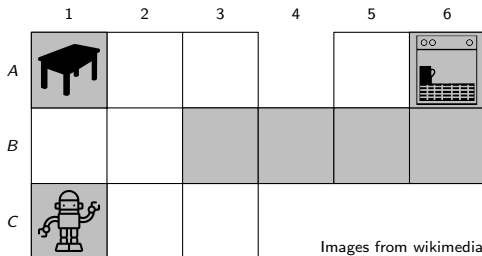
Landmarks: Example

Example

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

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

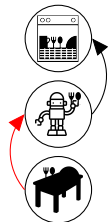
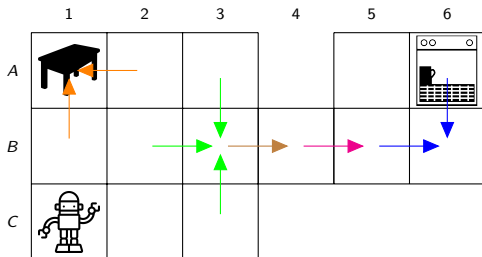
Fact Landmarks: Example



Each fact in gray is a fact landmark:

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

Disjunctive Action Landmarks: Example



Actions of same color form disjunctive action landmark:

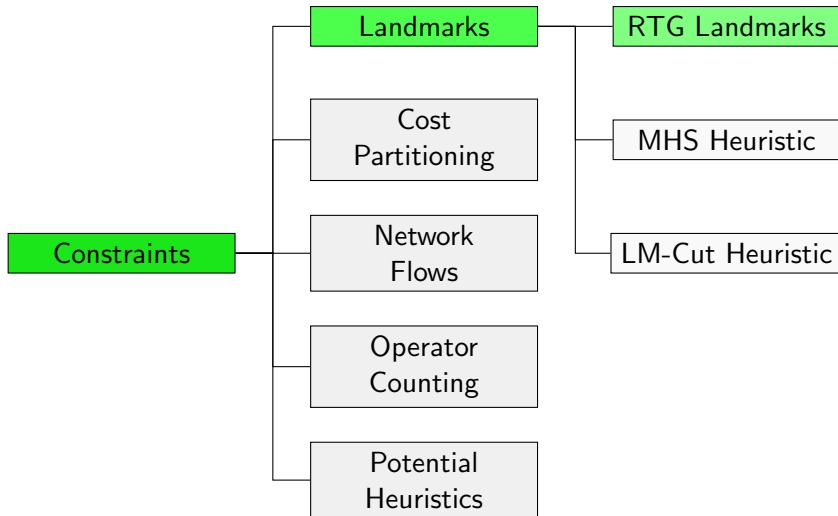
- {pickup}
- {load}
- {move-B3-B4}
- {move-B4-B5}
- {move-A6-B6, move-B5-B6}
- {move-A3-B3, move-B2-B3, move-C3-B3}
- {move-B1-A1, move-A2-A1}
- ...

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.
- Deciding whether a given variable 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.

Landmarks from RTGs

Content of this Course: Constraints



Computing Landmarks

How can we come up with landmarks?

Most landmarks are derived from the **relaxed task graph**:

- RHW landmarks: Richter, Helmert & Matthias 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 discuss **h^m landmarks** restricted to $m = 1$ and to STRIPS planning tasks.

Incidental Landmarks: Example

Example (Incidental Landmarks)

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

$$V = \{a, b, c, d, e, f\},$$

$$I = \{a \mapsto \mathbf{T}, b \mapsto \mathbf{T}, c \mapsto \mathbf{F}, d \mapsto \mathbf{F}, e \mapsto \mathbf{T}, f \mapsto \mathbf{F}\},$$

$$o_1 = \langle \{a\}, \{c, d, e\}, \{b\} \rangle,$$

$$o_2 = \langle \{d, e\}, \{f\}, \{a\} \rangle, \text{ and}$$

$$\gamma = \{e, f\}.$$

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

Definition (Causal Fact Landmark)

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

An atomic proposition $v = \mathbf{T}$ for $v \in V$ is a **causal fact landmark**

- if $v \in \gamma$
- or if for all goal paths $\pi = \langle o_1, \dots, o_n \rangle$ there is an o_i with $v \in \text{pre}(o_i)$.

Causal Landmarks: Example

Example (Causal Landmarks)

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

$$V = \{a, b, c, d, e, f\},$$

$$I = \{a \mapsto \mathbf{T}, b \mapsto \mathbf{T}, c \mapsto \mathbf{F}, d \mapsto \mathbf{F}, e \mapsto \mathbf{T}, f \mapsto \mathbf{F}\},$$

$$o_1 = \langle \{a\}, \{c, d, e\}, \{b\} \rangle,$$

$$o_2 = \langle \{d, e\}, \{f\}, \{a\} \rangle, \text{ and}$$

$$\gamma = \{e, f\}.$$

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 a simplified version of RTGs 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.

Simplified Relaxed Task Graph

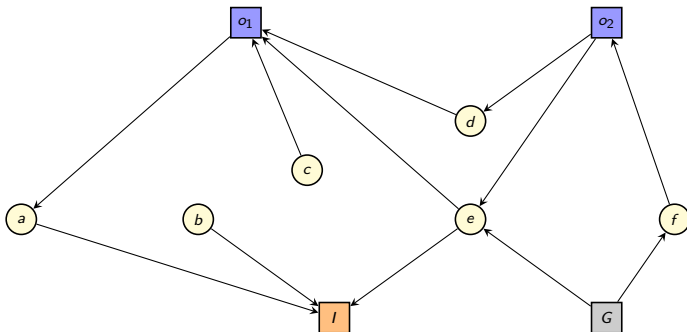
Definition

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

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

Simplified RTG: Example

The simplified RTG for our example task is:



Characterizing Equation System

Theorem

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

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

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

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

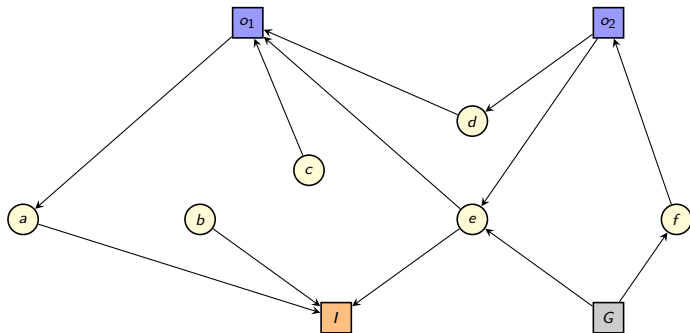
$$LM(n) = \{n\} \cup \bigcap_{\langle n, n' \rangle \in A} LM(n') \quad \text{type}(n) = \vee$$

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

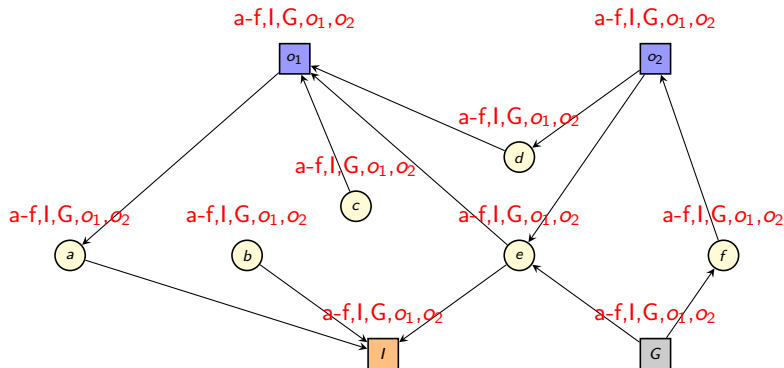
The equation system has a unique maximal solution (maximal with regard to set inclusion).

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

Computation: Example

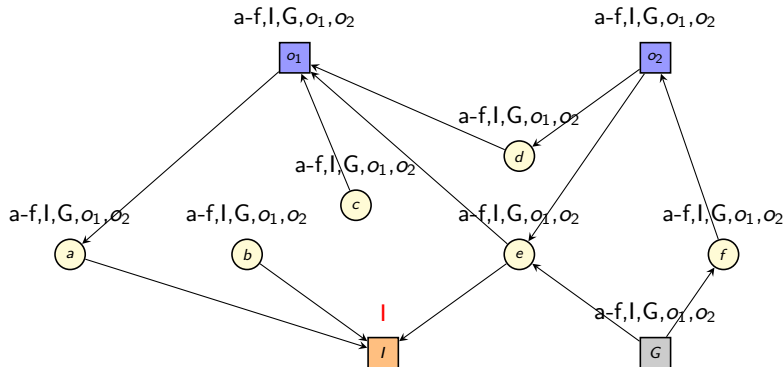


Computation: Example



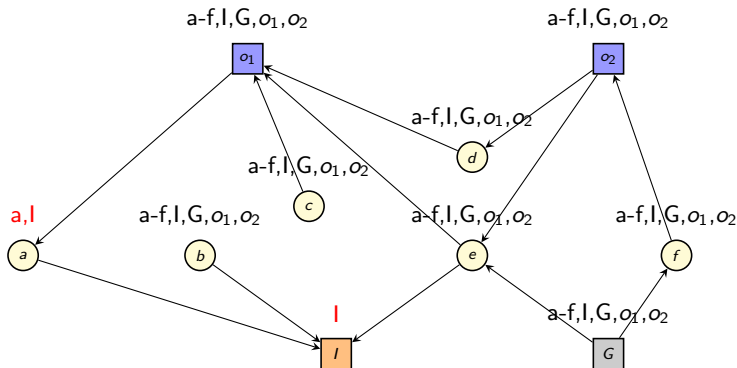
Initialize with all nodes

Computation: Example



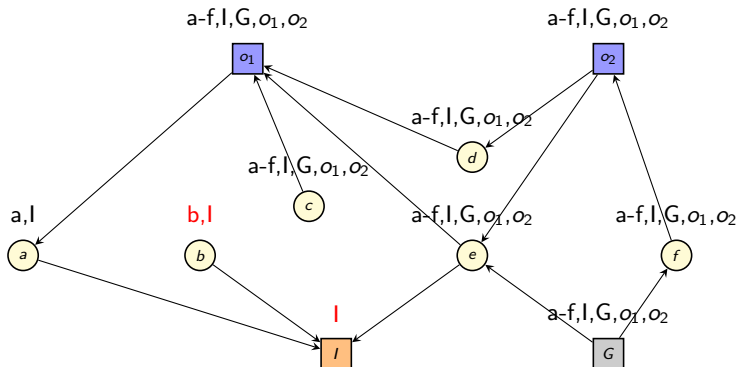
$$LM(I) = \{I\}$$

Computation: Example



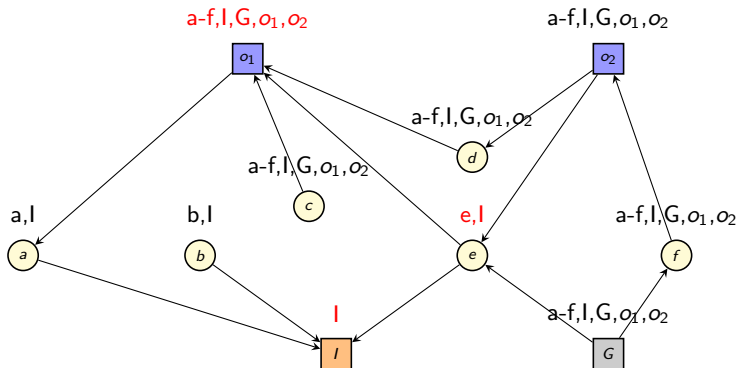
$$LM(a) = \{a\} \cup LM(I)$$

Computation: Example



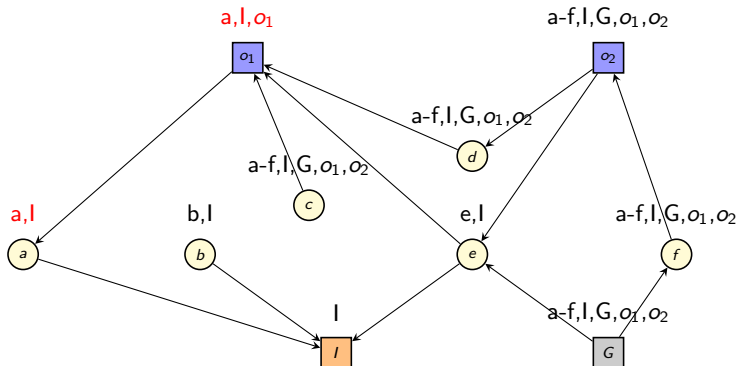
$$LM(b) = \{b\} \cup LM(I)$$

Computation: Example



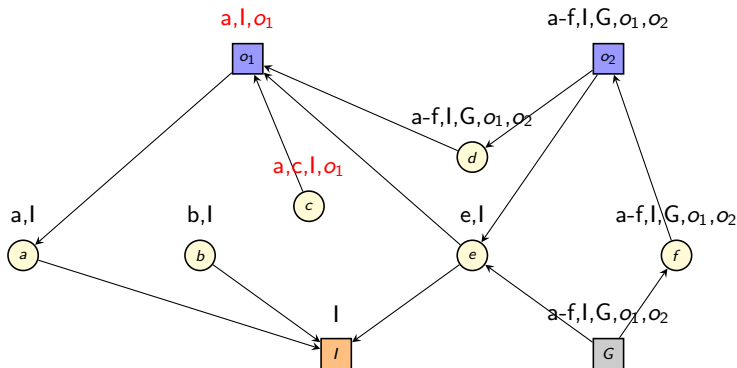
$$LM(e) = \{e\} \cup (LM(I) \cap LM(o_1))$$

Computation: Example



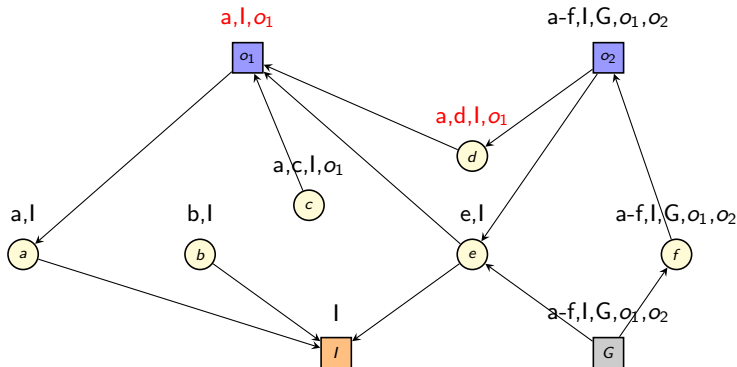
$$LM(o_1) = \{o_1\} \cup LM(a)$$

Computation: Example



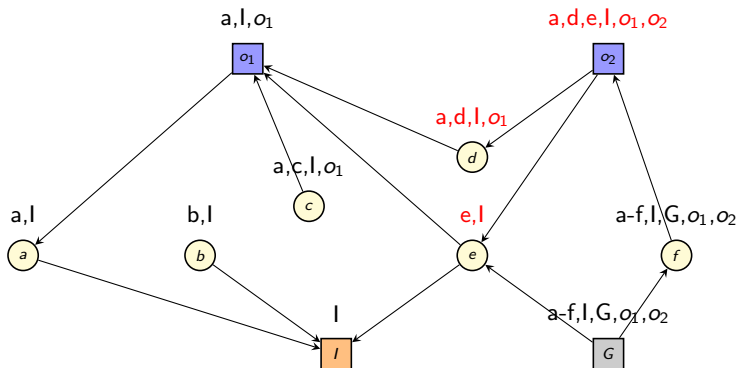
$$LM(c) = \{c\} \cup LM(o_1)$$

Computation: Example



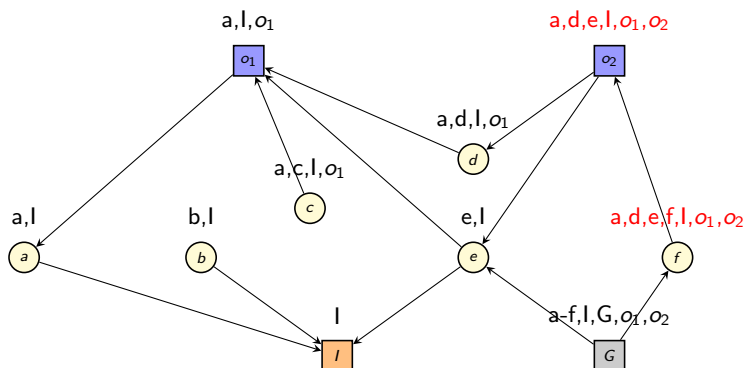
$$LM(d) = \{d\} \cup LM(o_1)$$

Computation: Example



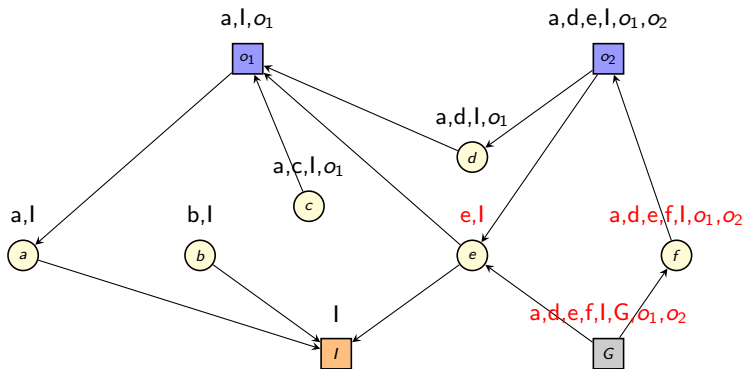
$$LM(o_2) = \{o_2\} \cup LM(d) \cup LM(e)$$

Computation: Example



$$LM(f) = \{f\} \cup LM(o_2)$$

Computation: Example



$$LM(G) = \{G\} \cup LM(e) \cup LM(f)$$

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 $\{v = \mathbf{T} \mid v \in V \text{ and } n_v \in \mathcal{L}\}$ 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 $\langle V, I, \{o_1, o_2\}, \gamma \rangle$ with

$$V = \{a, b, c, d, e, f\},$$

$$I = \{a \mapsto \mathbf{T}, b \mapsto \mathbf{T}, c \mapsto \mathbf{F}, d \mapsto \mathbf{F}, e \mapsto \mathbf{T}, f \mapsto \mathbf{F}\},$$

$$o_1 = \langle \{a\}, \{c, d, e\}, \{b\} \rangle,$$

$$o_2 = \langle \{d, e\}, \{f\}, \{a\} \rangle, \text{ and}$$

$$\gamma = \{e, f\}.$$

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

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 .

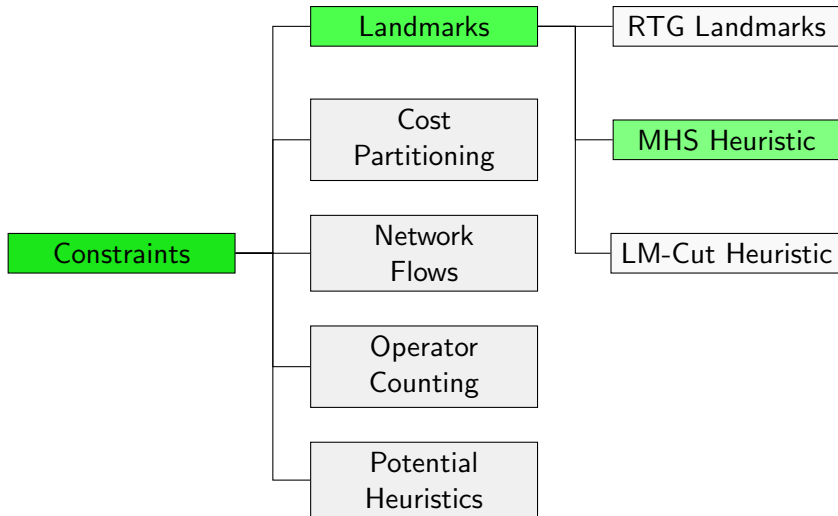
Proof.

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 . $\square$

Minimum Hitting Set Heuristic

Content of this Course: Constraints



Exploiting Disjunctive Action Landmarks

- The cost $cost(L)$ of a disjunctive action landmark L is an admissible heuristic, but it is usually not very informative.
- Landmark heuristics typically aim to combine multiple disjunctive action landmarks.

How can we exploit a given set $\mathcal{L}$ of disjunctive action landmarks?

- Sum of costs $\sum_{L \in \mathcal{L}} cost(L)$?
 $\rightsquigarrow$ **not admissible!**
- Maximize costs $\max_{L \in \mathcal{L}} cost(L)$?
 $\rightsquigarrow$ **usually very weak heuristic**
- **better:** Hitting sets

Hitting Sets

Definition (Hitting Set)

Let X be a set, $\mathcal{F} = \{F_1, \dots, F_n\} \subseteq 2^X$ be a family of subsets of X and $c : X \rightarrow \mathbb{R}_0^+$ be a cost function for X .

A **hitting set** is a subset $H \subseteq X$ that “hits” all subsets in $\mathcal{F}$, i.e., $H \cap F \neq \emptyset$ for all $F \in \mathcal{F}$. The **cost** of H is $\sum_{x \in H} c(x)$.

A **minimum hitting set (MHS)** is a hitting set with minimal cost.

MHS is a “classical” NP-complete problem (Karp, 1972)

Example: Hitting Sets

Example

$$X = \{o_1, o_2, o_3, o_4\}$$

$$\mathcal{F} = \{\{o_4\}, \{o_1, o_2\}, \{o_1, o_3\}, \{o_2, o_3\}\}$$

$$c(o_1) = 3, \quad c(o_2) = 4, \quad c(o_3) = 5, \quad c(o_4) = 0$$

What is minimum hitting set?

Example: Hitting Sets

Example

$$X = \{o_1, o_2, o_3, o_4\}$$

$$\mathcal{F} = \{\{o_4\}, \{o_1, o_2\}, \{o_1, o_3\}, \{o_2, o_3\}\}$$

$$c(o_1) = 3, \quad c(o_2) = 4, \quad c(o_3) = 5, \quad c(o_4) = 0$$

What is minimum hitting set?

Solution: $\{o_1, o_2, o_4\}$ with cost $3 + 4 + 0 = 7$

Hitting Sets for Disjunctive Action Landmarks

Idea: **disjunctive action landmarks** are interpreted as instance of **minimum hitting set**

Definition (Hitting Set Heuristic)

Let $\mathcal{L}$ be a set of disjunctive action landmarks. The **hitting set heuristic** $h^{MHS}(\mathcal{L})$ is defined as the cost of a minimum hitting set for $\mathcal{L}$ with $c(o) = cost(o)$.

Proposition (Hitting Set Heuristic is Admissible)

Let $\mathcal{L}$ be a set of disjunctive action landmarks for state s . Then $h^{MHS}(\mathcal{L})$ is an admissible estimate for s .

Hitting Set Heuristic: Discussion

- The hitting set heuristic is the **best possible** heuristic that only uses the given information...
- ...but is NP-hard to compute.
- $\rightsquigarrow$ Use approximations that can be efficiently computed.
 $\Rightarrow$ **LP-relaxation**, **cost partitioning** (both discussed later)

Summary

Summary

- **Fcat 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
- **Relaxed task graphs** allows efficient computation of landmarks
- **Hitting sets** yield the most accurate heuristic for a given set of disjunctive action landmarks
- Computation of **minimal hitting set** is NP-hard