Landmark-Aware Strategies for Hierarchical Planning

Mohamed Elkawkagy, Pascal Bercher,
Bernd Schattenberg and Susanne Biundo

Institute of Artificial Intelligence

June 12th, 2011

ulm university universität ulm

HTN Planning – Problem Formalization

$\pi = \langle s_{init}, P_{init}, D \rangle$ is an HTN planning problem with

- s_{init} is the initial state
- P_{init} is the initial partial plan that needs to be decomposed
- $D = \langle T, M \rangle$ is the domain model with
  - T is a set of task schemata of the form $t(\bar{\tau}) = \langle pre, eff \rangle$, $\bar{\tau}$ is the parameter list of t
  - M is a set of decomposition methods of the form $m = \langle t(\bar{\tau}), P \rangle$, P is a partial plan

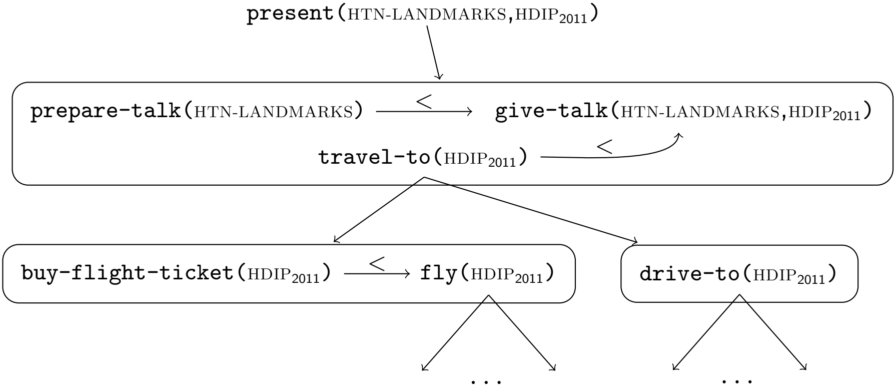
HTN Planning – Solution Formalization

Let $\pi = \langle s_{init}, P_{init}, D \rangle$ an HTN planning problem.

The partial plan P_{sol} is a solution to π if and only if

- P_{sol} contains only primitive tasks
- P_{sol} is obtained from P_{init} by decomposing non-primitive tasks
- Every linearization of P_{sol} is executable in s_{init}

HTN Planning – Example



Motivation

- Development of novel HTN planning search strategies other than progression:
  - Strategies estimate future branching factor based on landmark information.
- Other well-known HTN planners and their strategies:
  - UMCP (make primitive, then resolve conflicts)
  - EMS (expand, then make sound)
  - SHOP (progression)
  - SHOP2 (progression)

Search Procedure

Algorithm: Generic Refinement Planning Algorithm

Input : The sequence $\texttt{Fringe} = \langle P_{\text{init}} \rangle$.

Output : A solution or $\texttt{fail}$.

while $\texttt{Fringe} = \langle P_1 \ldots P_n \rangle \neq \varepsilon$ **do**

  $F \leftarrow f^{\text{FlawDet}}(P_1)$

  if $F = \emptyset$ **then return** P_1

  $\langle \texttt{m}_1 \ldots \texttt{m}_k \rangle \leftarrow f^{ModOrd}\big(\bigcup_{\texttt{f} \in F} f^{\text{ModGen}}(\texttt{f}) \big)$

  $\texttt{succ} \leftarrow \langle \text{app}(\texttt{m}_1, P_1) \ldots \text{app}(\texttt{m}_k, P_1) \rangle$

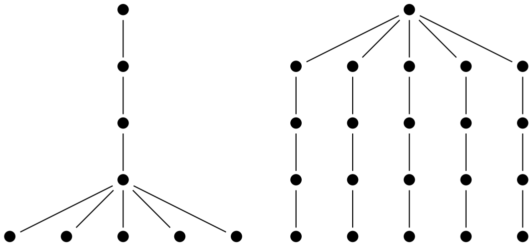
  $\texttt{Fringe} \leftarrow f^{PlanOrd}(\texttt{succ} \circ \langle P_2 \ldots P_n \rangle)$

return $\texttt{fail}$

Landmark-Aware Strategies – Motivation

Chosen ordering functions:

- Plan ordering function $f^{PlanOrd}$: *Fewer Modifications First*
- Modification ordering function f^{ModOrd}: *Use landmarks to reduce branching factor by deciding which non-primitive task to decompose first.*



Decomposition Graph & Landmark Table

Figure: Task Decomposition Graph

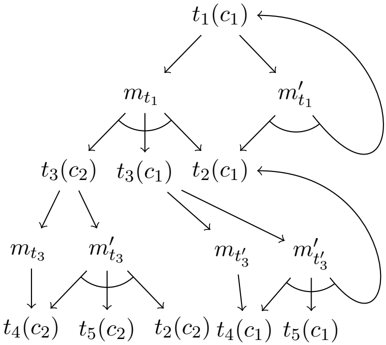


Table: Landmark Table

t	$M(t)$	$O(t)$
$t_1(c_1)$	$\{t_2(c_1)\}$	$\{\{t_3(c_2), t_3(c_1)\}, \{t_1(c_1)\}\}$
$t_3(c_2)$	$\{t_4(c_2)\}$	$\{\emptyset, \{t_5(c_2), t_2(c_2)\}\}$
$t_3(c_1)$	$\{t_4(c_1)\}$	$\{\emptyset, \{t_5(c_1), t_2(c_1)\}\}$

Mandatory Tasks:
Intersection of method's plans

Optional Tasks:
Remaining tasks

Landmark-Aware Strategies – $\mathrm{lm}_1, \mathrm{lm}_2$

Let P be a partial plan, $t_i, t_j \in P$ be ground non-primitive tasks.
Let LT be a landmark table and

- $\langle\ t_i\ ,\ M(t_i)\ ,\ O(t_i)\ \rangle \in LT$,
- $\langle\ t_j\ ,\ M(t_j)\ ,\ O(t_j)\ \rangle \in LT$

Definition (Landmark-aware strategy lm_2)

f^{ModOrd} *decomposes* t_i *before* t_j *if and only if*

$$\sum_{o \in O(t_i)} |o| \quad < \sum_{o \in O(t_j)} |o|$$

Landmark-Aware Strategies – $\mathrm{lm}_1, \mathrm{lm}_2$

Let P be a partial plan, $t_i, t_j \in P$ be ground non-primitive tasks.
Let LT be a landmark table and

- $\langle\ t_i\ ,\ M(t_i)\ ,\ O(t_i)\ \rangle \in LT$,
- $\langle\ t_j\ ,\ M(t_j)\ ,\ O(t_j)\ \rangle \in LT$

Definition (Landmark-aware strategy lm_1)

f^{ModOrd} decomposes t_i before t_j if and only if

$$\sum_{o \in O(t_i)} |o|_{LT} < \sum_{o \in O(t_j)} |o|_{LT}$$

for $|o|_{LT} := |\{t \in o \mid \langle t, M(t), O(t)\rangle \in LT\}|$.

Landmark-Aware Strategies – $\mathrm{lm}_1, \mathrm{lm}_2$ (Examples)

Let a plan P contain $t_1(c_1)$ and $t_3(c_2)$. It holds:

- $\langle\ t_1(c_1)\ ,\ \{t_2(c_1)\}\ ,\ \{\ \{t_3(c_2), t_3(c_1)\}\ ,\ \{t_1(c_1)\}\ \}\ \rangle \in LT$,
- $\langle\ t_3(c_1)\ ,\ \{t_4(c_1)\}\ ,\ \{\ \emptyset\ ,\ \{t_5(c_1), t_2(c_1)\}\ \}\ \rangle \in LT$

Definition (Landmark-aware strategy lm_2)

f^{ModOrd} *decomposes t_i before t_j if and only if*

$$\sum_{o \in O(t_i)} |o| \quad < \sum_{o \in O(t_j)} |o|$$

$$\sum_{o \in O(t_3)} |o| \quad < \sum_{o \in O(t_1)} |o| \quad \Leftrightarrow 0 + 2 < 2 + 1 \Rightarrow \text{decompose } t_3!$$

Landmark-Aware Strategies – $\mathrm{lm}_1, \mathrm{lm}_2$ (Examples)

Let a plan P contain $t_1(c_1)$ and $t_3(c_2)$. It holds:

- $\langle\ t_1(c_1)\ ,\ \{t_2(c_1)\}\ ,\ \{\ \{t_3(c_2), t_3(c_1)\}\ ,\ \{t_1(c_1)\}\ \}\ \rangle \in LT$,
- $\langle\ t_3(c_1)\ ,\ \{t_4(c_1)\}\ ,\ \{\ \emptyset\ ,\ \{t_5(c_1), t_2(c_1)\}\ \}\ \rangle \in LT$

Definition (Landmark-aware strategy lm_1)

f^{ModOrd} *decomposes* t_i *before* t_j *if and only if*

$$\sum_{o \in O(t_i)} |o|_{LT} < \sum_{o \in O(t_j)} |o|_{LT}$$

for $|o|_{LT} := |\{t \in o \mid \langle t, M(t), O(t)\rangle \in LT\}|$.

$$\sum_{o \in O(t_3)} |o|_{LT} < \sum_{o \in O(t_1)} |o|_{LT} \Leftrightarrow 0 + 0 < 2 + 1 \Rightarrow \text{decompose } t_3!$$

Landmark-Aware Strategies – $\mathrm{lm}_1^*, \mathrm{lm}_2^*$

Closure of the strategies traverse the graph to the very last level:

$$O^*(t) = O(t) \cup \bigcup_{o \in O(t)} \Big(\bigcup_{t' \in o} O^*(t') \Big)$$
$$\text{with } \langle t, M(t), O(t) \rangle \in LT$$

t	$M(t)$	$O(t)$	$O^*(t)$
$t_1(c_1)$	$\{t_2(c_1)\}$	$\{\{t_3(c_2), t_3(c_1)\}, \{t_1(c_1)\}\}$	$O(t_1(c_1)) \cup O(t_3(c_2)) \cup O(t_3(c_1))$
$t_3(c_2)$	$\{t_4(c_2)\}$	$\{\emptyset, \{t_5(c_2), t_2(c_2)\}\}$	$O(t_3(c_2))$
$t_3(c_1)$	$\{t_4(c_1)\}$	$\{\emptyset, \{t_5(c_1), t_2(c_1)\}\}$	$O(t_3(c_1))$

Evaluation – Domains

Conducted experiments on two different HTN planning domains:

- UM Translog
  - Logistics domain – originally formulated for UMCP.
  - 13 qualitatively different problem instances.
  - Deep hierarchy! $\rightarrow \mathrm{lm}_i/\mathrm{lm}_i^*$ strategies well-informed.
- Satellite
  - Satellite observation tasks – originally designed for classical planning – hierarchy put on top of it.
  - Problem structure: x observations – y satellites – z modi.
  - Shallow hierarchy! $\rightarrow \mathrm{lm}_i/\mathrm{lm}_i^*$ strategies less informed.

Evaluation – Some Notes

- Comparisons with:
  - standard HTN strategies like *lcf* (least commitment first) and *HotSpot*/*HotZone* (modifications that don't effect many plan fragments).
  - planning systems mentioned in motivation: our generic refinement algorithm can simulate behavior of (almost) any system when using the according modification and plan ordering functions f^{ModOrd} and $f^{PlanOrd}$.
- Tables' syntax:
  - #n == number of generates search nodes
  - blue == best result / red == second-best result

Evaluation – UM Translog

f^{ModOrd}	1 #n	2 #n	3 #n	4 #n	5 #n	6 #n	7 #n	8 #n	9 #n	10 #n	11 #n	12 #n	13 #n
UMCP	58	156	177	55	57	308	63	75	220	161	92	90	70
EMS	147	405	211	127	114	–	1571	113	–	2558	879	500	784
SHOP	89	164	146	106	83	926	98	95	–	247	121	173	150
lm_1	52	133	145	62	53	291	63	71	158	183	75	72	142
lm_1^*	51	135	154	52	65	266	61	61	304	158	78	89	189
lm_2	62	135	141	53	55	339	109	73	420	211	84	91	104
lm_2^*	124	146	137	57	51	305	110	81	367	142	87	84	114
lcf	55	155	162	78	127	327	62	86	–	227	79	90	247
da-HotSpot	144	644	239	114	148	723	99	120	–	184	641	588	172
du-HotSpot	101	459	1508	160	117	–	1047	75	–	1390	424	307	643
HotZone	55	197	191	55	55	–	159	122	–	701	81	76	345

Evaluation – Satellite

f^{ModOrd}	1–1–1 #n	1–2–1 #n	2–1–1 #n	2–1–2 #n	2–2–1 #n	2–2–2 #n
UMCP	83	36	883	1558	278	1062
EMS	65	47	1586	–	1219	–
SHOP	62	105	138	–	1406	–
lm_1	73	194	560	352	693	295
lm_1^*	78	34	847	1803	739	619
lm_2	78	128	4890	200	–	483
lm_2^*	73	91	–	1905	–	146
lcf	86	71	1120	3022	407	–
da-HotSpot	56	68	782	832	2186	142
du-HotSpot	100	139	–	–	–	–
HotZone	61	57	1281	–	1094	871

Summary

- Partitioning of method's tasks in mandatory and optional sets.
- Landmark strategies use size of (closure of) optional sets to estimate branching factor.
- Evaluation looks promising.