Proceedings of the Twenty-Fourth International Conference on Automated Planning and Scheduling

Landmark-Based Plan Distance Measures for Diverse Planning

Daniel Bryce
SIFT, LLC.
dbryce@sift.net

Abstract

Prior approaches to generating diverse plans in domain-independent planning seek out variations on plan structure such as actions or causal links used, or states entered. Measuring such syntactic differences between plans can be misleading because syntactically different plans can be semantically identical. We develop a landmark-based plan distance measure that captures semantic differences between plans.

The landmark-based distance measure focuses on the disjunctive landmarks satisfied by each plan. We develop a simple algorithm for finding diverse plans that is based upon the LAMA planner. We illustrate that, in comparison with plan distance measures, landmark-based plan distance is not as susceptible to including irrelevant or redundant actions in plans to increase plan distance. Through extensive empirical evaluation, we find that high landmark distance between plans implies high action set distance, but not vice versa. Landmark-based plan distance overcomes some of the weaknesses of syntactic plan distance measures and can be used to find plan sets that are both landmark diverse and action set diverse.

1 Introduction

Diverse planning is an important tool for decision support scenarios where a human analyst wants to understand the space of possible plans or has difficulty specifying the planning domain model that exactly matches their application (a similar motivation underlies information retrieval systems that return multiple results). Diversity is measured by the average distance between plans in a plan set, and many domain-independent distance measures have been explored.

Prior work (Nguyen et al. 2012; Coman and Muñoz-Avila 2011) defines the distance between two plans in terms of the symmetric difference of the respective action, causal link, or state sets. The most scalable approaches make use of the action set distance measure and adjust the heuristic function of either FF (Hoffmann and Nebel 2001) or LPG (Gerevini, Saetti, and Serina 2003) to bias search away from previously generated plans. These algorithms greedily add new plans to a plan set until the set reaches a given size or has a minimum average inter-plan distance. We note that these prior distance measures all suffer from a common problem: adding irrelevant or redundant actions to the plan can increase plan distance. In most cases, it is possible to extend a plan by a single irrelevant action to attain a "new" plan.

We find that for this reason, prior plan distance measures can produce plans that are semantically identical, but syntactically different. That is, they deem plans different even when they have the same causal proof or validation structure (Kambhampati and Kedar 1994). Said another way, they can count properties of a plan that are not essential to proving its correctness. It is debatable whether varying the non-essential properties of plans contributes to the overall plan set diversity. However, we note that there has been no study of how to guarantee diversity in the essential properties of plans. For this reason we consider landmarks.

Landmarks (Hoffmann, Porteous, and Sebastia 2004) are typically phrased as atomic propositions or disjunctive sets of propositions that must be satisfied by all plans solving a planning instance. For example, each goal proposition is a landmark. If a landmark is achieved by a set of actions that share a common precondition, then that precondition is landmark. Even when the actions supporting a landmark do not share a common precondition, a disjunction of propositions appearing in their preconditions is a landmark. Thus, a disjunctive landmark represents fundamentally different subgoals that must be established by plans.

By definition, all landmarks for a planning instance are satisfied by any plan for the instance. Furthermore, two plans must have different causal proofs if they satisfy alternative landmark disjuncts. Counting the number of uniquely satisfied landmark disjuncts leads to us to a landmark-based distance measure. This distance measure guarantees that if it is non-zero then the plans must have different causal proofs and are thus semantically different.

We develop an extension of the LAMA planner, called DLAMA, that finds a set of plans by selecting alternative sets of landmark disjuncts and finding a plan for each. Each set of landmark disjuncts is greedily chosen to maximize the distance from previously selected sets. DLAMA can then invoke multiple parallel instances of LAMA to find solutions satisfying the different landmark disjunct sets. We modify LAMA to count only the chosen landmark disjuncts in its landmark count heuristic and reject solutions that do not satisfy all chosen landmark disjuncts. DLAMA is greedy prior to planning (in selecting landmark disjunct sets), whereas

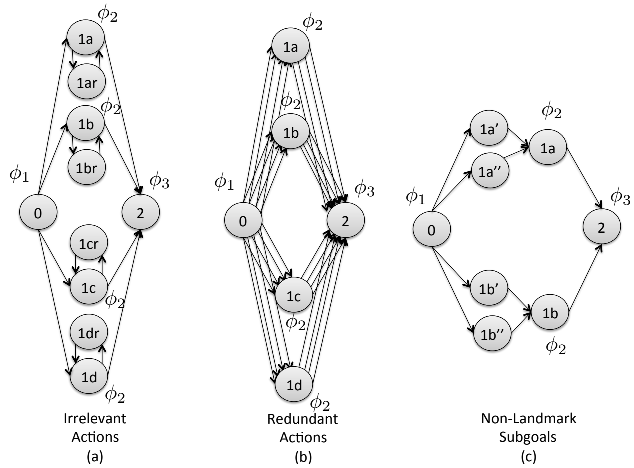

Figure 1: Road network planning examples with irrelevant actions, redundant actions, or non-landmark subgoals. In all cases, the initial is node 0 and the goal is node 2. The nodes related to different landmarks are labeled ϕ_1 to ϕ_3.

prior works are greedy during planning (biasing search away from the current plan set). The primary benefit of this type of greediness is that DLAMA can be parallelized trivially.

We compare DLAMA to LPG-d (Nguyen et al. 2012) and FF^{GrDIV} (Coman and Muñoz-Avila 2011), two contemporary diverse planners that employ action set-based distance measures in different capacities. We show that high landmark distance implies high action set distance and that DLAMA can often overcome misleading forms of diversity that plague other diverse planners. We also show that DLAMA is superior in maximizing landmark distance.

2 Diversity Examples

We illustrate the relative strengths and weaknesses of landmark-based and action set-based distance measures for diverse planning with three examples (see Figure 1). We define the distance measures in the following section. The examples highlight the following points:

- Irrelevant actions increase action set distance, but not necessarily landmark distance.
- Redundant actions increase action set distance, but not necessarily landmark distance.
- Landmark-based distance measures can fail to capture variations in satisfied subgoals that are necessary in a *subset* of all plans.

Irrelevant Actions: Consider a simple navigation task that involves the first road network depicted in Figure 1a. The initial state is location 0, and the goal is location 2. The available actions change the agent's location, as dictated by the edges. There are many solutions to this problem. It is possible to take different paths through 1a, 1b, 1c, and 1d from location 0. It is also possible to apply a reversible action in locations 1a, 1b, 1c, and 1d.

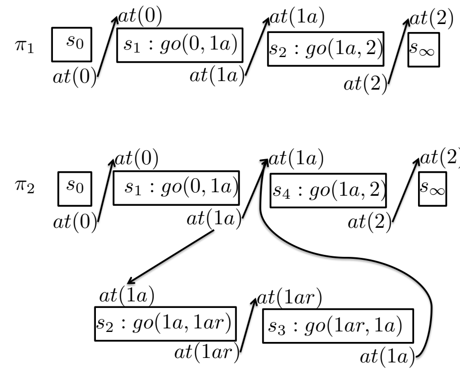

Figure 2: Causal proofs of two plans π_1 and π_2 in the road network instance with irrelevant actions.

In extracting the landmarks for this instance, we attain the followings set of landmarks $\mathcal{L}$:

$$\{\phi_1 : \{at(0)\},\\ \phi_2 : \{at(1a), at(1b), at(1c), at(1d)\},\\ \phi_3 : \{at(2)\}\}$$

where landmarks ϕ_1 and ϕ_3 are atomic and ϕ_2 is disjunctive.

Solutions to this problem illustrate how irrelevant actions can increase action set-based distance measures. Consider the plans:

$$\pi_1 = \langle go(0, 1a), go(1a, 2)\rangle$$
$$\pi_2 = \langle go(0, 1a), go(1a, 1ar), go(1ar, 1a), go(1a, 2)\rangle$$
$$\pi_3 = \langle go(0, 1b), go(1b, 2)\rangle$$

which satisfy the respective landmark disjuncts

$$\mathcal{L}(\pi_1) = \{\{at(0)\}, \{at(1a)\}, \{at(2)\}\}$$
$$\mathcal{L}(\pi_2) = \{\{at(0)\}, \{at(1a)\}, \{at(2)\}\}$$
$$\mathcal{L}(\pi_3) = \{\{at(0)\}, \{at(1b)\}, \{at(2)\}\}$$

We summarize the size of the symmetric difference ($\triangle$) between the action sets and satisfied landmark disjuncts plans in the following table.

Pair	$\triangle$ Action Sets	$\triangle$ Landmark Disjuncts
(π_1, π_2)	2	0
(π_1, π_3)	4	2
(π_2, π_3)	6	2

Plans π_1 and π_2 satisfy the same number of unique landmark disjuncts when compared with π_3, but have different numbers of unique actions. Plan π_2 seems preferable in terms of unique actions, but at the cost of increased plan length.

In considering the causal proofs of π_2 (see Figure 2), we note that there are two ways to achieve $at(1a)$. The first uses the effect of go(0, 1a) to establish $at(1a)$, and the other uses go(1ar, 1a). The first way to support $at(1a)$ is identical to that of plan π_1. Given the overlap in causal proofs (and satisfied landmark disjuncts) it appears that π_1 and π_2 are not meaningfully different.

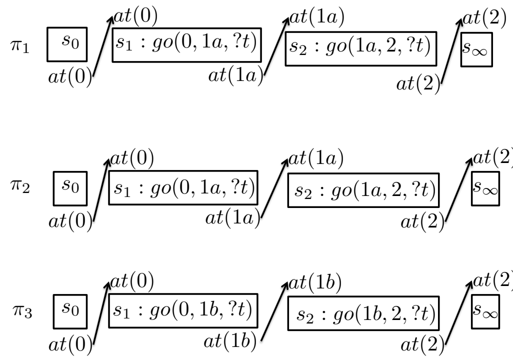

Figure 3: Causal proofs of plans π_1, π_2 and π_3 in the road network instance with redundant actions.

Redundant Actions: The second instance involves four possible actions to travel between each location. These actions are of the form $go(0, 1a, t1)$ or $go(0, 1a, t2)$ and have identical preconditions and effects, but use a different object $t1$ or $t2$ (e.g., a different truck). The objects $t1 - t4$ are not mentioned in the action preconditions or effects, and effectively rename the same action.[1]

Consider the three plans:

$$\pi_1 = \langle go(0, 1a, t1), go(1a, 2, t1)\rangle$$
$$\pi_2 = \langle go(0, 1a, t2), go(1a, 2, t2)\rangle$$
$$\pi_3 = \langle go(0, 1b, t1), go(1b, 2, t1)\rangle$$

where the set of landmarks and satisfied landmark disjuncts by the plans in this instance are identical to the instance with irrelevant actions. An action set-based distance metric will identify four actions in the symmetric difference of the actions sets in plans π_1 and π_2, namely $go(0, 1a, t1)$, $go(0, 1a, t2)$, $go(1a, 2, t1)$, and $go(1a, 2, t2)$. The causal proofs of the plans are listed in Figure 3. We see that the third parameter of each action is irrelevant, and hence plans π_1 and π_2 have identical causal proofs. Plans that use different actions to achieve the same subgoals are seen as different by action set-based distance measures (and also with causal link-based measures). In this case, the subgoals satisfied by the plans are landmarks and thus easily identified by landmark-based distance measures as not contributing to diversity. This is not always possible with the landmark-based distance measures, as highlighted by the next example.

Non-Landmark Subgoals: The last example in Figure 1 illustrates how alternative subgoals that are not landmarks can be achieved in different ways. The landmarks in the final instance are:

$$\{\phi_1 : \{at(0)\}, \phi_2 : \{at(1a), at(1b)\}, \phi_3 : \{at(2)\}\}$$

The instance is constructed so that $1a'$, $1a''$, $1b'$, and $1b''$ are not identified as part of a disjunctive landmark. These subgoals represent an opportunity to find different plans that landmark-based distance measures cannot identify. For example, the plans

$$\pi_1 = \langle go(0, 1a'), go(1a', 1a), go(1a, 2)\rangle$$
$$\pi_2 = \langle go(0, 1a''), go(1a'', 1a), go(1a, 2)\rangle$$

reach $1a$ using alternative actions, but satisfy the same landmark disjuncts. LAMA could identify $\{1a', 1a''\}$ as a disjunctive landmark if the goal is to reach location $1a$, and landmark-based distance measures would treat plans that go through either $1a'$ or $1a''$ as different. This example illustrates how landmark-based distance measures are tied to landmark identification algorithms. While LAMA cannot find it, there is a disjunctive landmark based on locations $\{1a', 1a'', 1b', 1b''\}$, which could then be used to find more diverse plans and differentiate plans π_1 and π_2 above.

In our empirical evaluation, we experiment with scalable versions of these domains. Each instance repeats the structure, between locations 0 and 2, n times. For example, when $n = 2$ in the instance with redundant actions in Figure 1b, there are locations 0, 1a-1d, 2, 3a-3d, and 4 where the paths from 0 to 2 are isomorphic to those between 2 and 4.

These intuitive examples illustrate both the benefits and challenges of using landmarks to compute diverse plans. In the following, we formally define plans and landmarks. Using these definitions, we formalize the distance measures discussed informally above.

3 SAS+ and Landmarks

A SAS+ task defines the tuple $\Pi = \langle \mathcal{V}, \mathcal{O}, s_0, s_* \rangle$, where:

- $\mathcal{V}$ is a set of state variables, where each $v \in \mathcal{V}$ has a finite domain $\mathcal{D}_v$. A partial variable assignment s is a set of variables, each with an assigned value. A state is a variable assignment over each variable in $\mathcal{V}$.
- $\mathcal{O}$ is a set of operators, where each $o \in \mathcal{O}$ defines a tuple $\langle pre(o), eff(o)\rangle$, where $pre(o)$ is a partial variable assignment and $eff(o)$ is a set of effects. Each effect is a triple $\langle cond, v, d\rangle$ where $cond$ is a partial variable assignment, v is a variable, and d is a value in the domain of v.
- s_0, the initial state, is a set of variable value pairs $\langle v, d\rangle$ such that each $v \in \mathcal{V}$ is assigned a value from its domain $\mathcal{D}_v$.
- s_*, the goal, is a set of variable value pairs $\langle v, d\rangle$.

An operator $o = \langle pre(o), eff(o)\rangle \in \mathcal{O}$ is applicable in s if $pre(o) \subseteq s$ and its effects are consistent. That is, there is a state s' such that $s'(v) = d$ for all $\langle cond, v, d\rangle \in eff(o)$ where $cond \subseteq s$ and $s'(v) = s(v)$ otherwise. The result of applying o in s is denoted $s[o] = s'$. An operator sequence $\pi = \langle o_1, ..., o_n\rangle$ applied in s results in a state $s[\pi] = s[o_1]...[o_n]$. An operator sequence π is a plan iff $s_* \subseteq s_0[\pi]$.

Landmarks are propositional formulas satisfied by each plan. A propositional formula ϕ over the variables $\mathcal{V}$ is a fact formula. A fact $\langle v, d\rangle$ is true at time i under the operator sequence $\pi = \langle o_1, ..., o_n\rangle$ iff $\langle v, d\rangle \in s_0[\langle o_1, ..., o_i\rangle]$. A fact formula ϕ is true at time i under the operator sequence

[1]The objects $t1 - t4$ could also have been mentioned in effects that are not relevant to the plan (e.g., used(t1))

$\pi = \langle o_1, ..., o_n \rangle$ iff ϕ holds given the facts in $s_0[\langle o_1, ..., o_i \rangle]$. The fact formula ϕ is a landmark iff for each plan π of Π, ϕ holds at some time.

Similar to LAMA, we restrict our focus to landmarks that are disjunctions of atomic facts represented by sets, so that each $\phi = \{\langle v, d\rangle, ..., \langle v', d'\rangle\}$ is a set of landmark disjuncts (when $|\phi| = 1$, the landmark is an atomic fact). We are agnostic to the way in which landmarks are computed, except that we expect them to be sound. We allow the set of landmarks $\mathcal{L}$ to be incomplete. Reasoning about landmark distance with an incomplete set of landmarks implies that the plan set may not be as diverse as possible; we cannot seek out plans varying the satisfaction of an unknown landmark's disjuncts. Therefore, we can only achieve diversity up to the completeness of the landmark set.

4 Landmark Distance

Each landmark $\phi \in \mathcal{L}$ will have one or many of its disjuncts satisfied by a plan. We denote the subset of satisfied disjuncts in a landmark by $\phi(\pi)$, defined as

$$\phi(\pi) = \bigcup_{1 \le i \le n} \phi \cap s_0[o_1, ..., o_i]$$

The set of all satisfied disjuncts is defined as

$$\mathcal{L}(\pi) = \{\phi(\pi) | \phi \in \mathcal{L}\}$$

and the set of all satisfied disjuncts of disjunctive landmarks is defined as

$$\mathcal{L}_{>1}(\pi) = \{\phi(\pi) | \phi \in \mathcal{L}_{>1}\}$$

where $\mathcal{L}_{>1} = \{\phi \in \mathcal{L} | \, |\phi| > 1\}$ is the set of all disjunctive landmarks.

The landmark distance $D_L(\pi, \pi')$ is defined as the size of the symmetric difference[2] of disjunctive landmark disjuncts satisfied by the plans:

$$D_L(\pi, \pi') = \frac{1}{|\mathcal{L}_{>1}|} \sum_{\phi \in \mathcal{L}_{>1}} \frac{|\phi(\pi) \triangle \phi(\pi')|}{|\phi(\pi) \cup \phi(\pi')|}$$

where $0 \le D_L(\pi, \pi') \le 1$ and $D_L(\pi, \pi') = 0$ when $|\mathcal{L}_{>1}| = 0$.

For example, using the plans from the first example in Section 2, we can compute the distances:

D_L	π_1	π_2	π_3
π_1	0	0	1
π_2	0	0	1
π_3	1	1	0

The distance $D_L(\pi_1, \pi_3)$ is computed as follows:

$$\frac{|\{at(1a)\} \triangle \{at(1b)\}|}{|\{at(1a)\} \cup \{at(1b)\}|} = \frac{|\{at(1a), at(1b)\}|}{|\{at(1a), at(1b)\}|} = 1$$

[2] $A \triangle B = (A \cup B) \backslash (A \cap B)$

5 Action Set Distance

Action set distance, as defined in prior work (Nguyen et al. 2012), is the proportion of unique actions appearing in two plans (i.e., normalized carnality of the symmetric difference between two plans' action sets):[3]

$$D_A(\pi, \pi') = \frac{|(A(\pi) \triangle A(\pi'))|}{|(A(\pi) \cup A(\pi'))|}$$

where $A(\pi)$ is the set of actions in plan π.

For example, using the plans from the first example in Section 2, we can compute the distances:

D_L	π_1	π_2	π_3
π_1	0	2/6	4/4
π_2	2/6	0	6/6
π_3	4/4	6/6	0

The distance between plans π_1 and π_3 is computed as follows:

$$\begin{aligned} &D_A(\pi_1, \pi_3) \\ &= \frac{|\{go(0, 1a), go(1a, 2)\} \triangle \{go(0, 1b), go(1b, 2)\}|}{|\{go(0, 1a), go(1a, 2)\} \cup \{go(0, 1b), go(1b, 2)\}|} \\ &= \frac{|\{go(0, 1a), go(1a, 2), go(0, 1b), go(1b, 2)\}|}{|\{go(0, 1a), go(1a, 2), go(0, 1b), go(1b, 2)\}|} \\ &= 4/4 = 1 \end{aligned}$$

Algorithm 1 DLAMA Algorithm

1: **procedure** DLAMA($\Pi, k, \overline{k}$)
2: $\quad\langle \mathcal{L}, \mathcal{O} \rangle \leftarrow$ConstructLandmarks($\Pi$)
3: $\quad L \leftarrow \{\}, Plans \leftarrow \{\}$
4: $\quad$**repeat**
5: $\quad\quad \mathcal{L}_i \leftarrow$NextLandmarkSet($\mathcal{L}$, L)
6: $\quad\quad L = L \cup \mathcal{L}_i$
7: $\quad\quad \pi_i \leftarrow$LAMA($\Pi$, $\langle \mathcal{L}_i, \mathcal{O} \rangle$)
8: $\quad\quad$**if** $\pi_i \ne FAIL$ **then**
9: $\quad\quad\quad Plans \leftarrow Plans \cup \pi_i$
10: $\quad\quad$**end if**
11: $\quad$**until** $|Plans| = k$ or $|L| = \overline{k}$
12: $\quad$**return** $Plans$
13: **end procedure**

6 DLAMA

DLAMA builds upon LAMA to find a diverse set of k plans. DLAMA calls LAMA within a loop that stops when it finds k plans or attempts $\overline{k} \ge k$ atomic landmark sets. In our experiments we selected $\overline{k} = 2k$. As outlined in Algorithm 1, DLAMA uses LAMA's landmark graph algorithm to find the landmarks $\mathcal{L}$ (line 2). Within its loop, DLAMA uses Algorithm 2 (line 5, described below) to construct each set of atomic (non-disjunctive) landmarks $\mathcal{L}_i$. For each atomic landmark set, DLAMA finds a plan using a modified version

[3] Coman and Muñoz-Avila (2011) define a similar unnormalized measure.

Algorithm 2 Greedy Landmark Set Selection Algorithm

1: **procedure** NEXTLANDMARKSET($\mathcal{L}$, L)
2: $\quad i \leftarrow |L| + 1$
3: $\quad \mathcal{L}_i \leftarrow \{\}$
4: $\quad$ **for** $\mathcal{L}_j \in L$ **do**
5: $\quad\quad d(\mathcal{L}_j) \leftarrow 0$
6: $\quad$ **end for**
7: $\quad \triangleright$ Select a disjunct from each landmark
8: $\quad$ **for** $\phi \in \mathcal{L}$ **do**
9: $\quad\quad \triangleright$ Compute distance wrt. each disjunct
10: $\quad\quad$ **for** $\langle v, d\rangle \in \phi$ **do**
11: $\quad\quad\quad$ **for** $\mathcal{L}_j \in L$ **do**
12: $\quad\quad\quad\quad \delta \leftarrow (\{\langle v, d\rangle\} \in \mathcal{L}_j \,?\, 0 : 1)$
13: $\quad\quad\quad\quad d(\mathcal{L}_j, \langle v, d\rangle) \leftarrow d(\mathcal{L}_j) + \delta$
14: $\quad\quad\quad$ **end for**
15: $\quad\quad$ **end for**
16: $\quad\quad$ **if** $|L| > 0$ **then**
17: $\quad\quad\quad \triangleright$ Select disjunct with max. avg. distance
18: $\quad\quad\quad \triangleright$ Tie break with max. min. distance
19: $\quad\quad\quad l \leftarrow \underset{\langle v,d\rangle \in \phi}{\arg\max} \frac{1}{|L|} \sum_{j=1}^{|L|} d(\mathcal{L}_j, \langle v, d\rangle)$
20: $\quad\quad$ **else**
21: $\quad\quad\quad l \leftarrow$lexical-first($\phi$)
22: $\quad\quad$ **end if**
23: $\quad\quad \mathcal{L}_i \leftarrow \mathcal{L}_i \cup \{l\}$
24: $\quad\quad$ **for** $\mathcal{L}_j \in L$ **do**
25: $\quad\quad\quad d(\mathcal{L}_j) \leftarrow d(\mathcal{L}_j, l)$
26: $\quad\quad$ **end for**
27: $\quad$ **end for**
28: $\quad$ **return** $\mathcal{L}_i$
29: **end procedure**

of LAMA (based upon the 2011 IPC version, unless stated otherwise). We modified LAMA to only return plans that are guaranteed to satisfy the given atomic landmark set $\mathcal{L}_i$ (described below).

Algorithm 2 returns a new atomic landmark set by selecting alternative disjuncts from each of the disjunctive landmarks. It does this by constructing each atomic landmark set $\mathcal{L}_i$ to greedily maximize the average Hamming distance between it and previously constructed atomic landmark sets $L = \{\mathcal{L}_1, ..., \mathcal{L}_{i-1}\}$. To construct an atomic landmark set, it considers each landmark $\phi \in \mathcal{L}$ and calculates the residual Hamming distance $d(\mathcal{L}_j, \langle v, d\rangle)$ of $\mathcal{L}_j$ to each of the previous atomic landmark sets for each possible disjunct $\langle v, d\rangle \in \phi$ (lines 10-15). It then selects the disjunct $l = \langle v, d\rangle$ that maximizes the average Hamming distance (breaking ties in favor of the disjunct with maximal minimal Hamming distance). The first atomic landmark set selects the lexical first disjunct from each disjunctive landmark. Our approach to selecting atomic landmark sets is similar to that of selecting program inputs in anti-random testing (Malaiya 1995).

LAMA Modifications: LAMA, and most other planners, must construct solutions that satisfy each landmark; by definition, a landmark must be satisfied by each possible plan. Without modification, LAMA will return the same solution for each atomic landmark set that we choose. We make three modifications to LAMA to enforce each plan satisfies the chosen atomic landmark set. First, we modify the landmark count heuristic to only treat a landmark as satisfied if the appropriate disjunct (as dictated by the atomic landmark set) is satisfied. Second, we only allow LAMA to return solutions whose landmark count heuristic is zero, meaning that it must satisfy all landmarks. Third, we allow LAMA to reopen closed search nodes if they are reached with a better landmark count heuristic value. Normally LAMA will only reopen search nodes if they can be reached with a lower g-value (path cost). Allowing LAMA to reopen nodes with lower landmark counts is essential to finding solutions that satisfy all chosen atomic landmarks. Otherwise, solution bearing search paths satisfying different landmark disjuncts cannot be extended through the closed nodes.

Discussion: As previously noted, DLAMA is unique in that it does not consider previously constructed plans to find a diverse plan set. Prior works evaluate each search node against previous plans to bias search away from these plans. We note that DLAMA does use a similar form of bias when selecting the atomic landmark sets. As such, DLAMA can reduce the per node cost of search, because it does not compute plan distance measures during search.

A potential downside to DLAMA that we did not observe in our experiments is that it may "thrash." That is, its reachability heuristic may not agree with our modification of the landmark count heuristic, resulting in more search node expansions than an unmodified LAMA. A particular atomic landmark set might require a significantly longer plan than what is estimated by the reachability heuristic. The reason that we did not observe this phenomena is due to how LAMA constructs its landmarks. LAMA subgoals on the first achievers of a landmark to identify predecessor landmarks, thus avoiding landmark disjuncts that are "costly" to satisfy.

While DLAMA is required to satisfy a different set of atomic landmarks with each solution, a solution may satisfy additional disjuncts. This can have a positive or negative effect upon the distance between plans. If the additional satisfied disjuncts are not satisfied by other plans, then distance is increased; if already satisfied, distance decreases. Ideally, the set of diverse plans should partition the disjuncts satisfied in each landmark to maximize the distance between plans. In most cases, the number of landmark disjuncts is relatively small compared to the number of diverse plans sought, making our choice of selecting a single disjunct per plan appropriate. While we do not actively bias LAMA away from plans that satisfy more than one landmark disjunct, we rely on the reachability heuristic to make shorter plans appealing and thus introduce a similar bias against satisfying extraneous landmark disjuncts.

DLAMA is not complete. As illustrated by the navigation task with non-landmark subgoals with $k > 2$, DLAMA will find two of the four distinct plans. The navigation task with irrelevant actions also demonstrates DLAMA's incompleteness because DLAMA will not generate plans with different numbers of irrelevant actions.

There is no guarantee that a plan exists for each atomic landmark set, just as there is no guarantee of a plan for the instance. By forcing LAMA to satisfy an atomic landmark set or fail, DLAMA may prove that no plan exists many times. The benchmarks we used for evaluation exhibit many satisfiable atomic landmark sets. However, proportionately higher DLAMA runtimes over LAMA can be attributed to cases where an unsatisfiable atomic landmark set causes DLAMA to exhaust the search space to show no plan exists.

Potential Improvements: A peculiarity to the landmarks identified by LAMA is that it only considers disjuncts that are atoms created from alternative groundings of the same predicate. For example, LAMA will generate disjunctive landmarks of the form $\{in(obj1, truck1), in(obj2, truck1)\}$ and not of the form $\{in(obj1, truck1), at(truck1, loc1)\}$. While DLAMA does not modify LAMA's landmark construction algorithms, it motivates the need for more types of disjunctive landmarks. Having more disjunctive landmarks equates to more dimensions along which to diversify plans.

We also note that DLAMA may be synergistic with existing diverse planners. For example, each invocation of LAMA with a different atomic landmark set can have multiple solutions that are diverse with respect to action set distance, as we previously illustrated in the navigation example with non-landmark subgoals. We may be able to explain all "interesting" forms of action set diversity by landmarks, but it remains to seen if we can discover these landmarks. One possible approach might be to consider path-dependent landmarks (Karpas and Domshlak 2009), which appear in a subset of solutions (e.g., the set of optimal solutions).

7 Empirical Results

There are two questions that we would like to answer in our empirical analysis of landmark-based distance measures for diverse planning and DLAMA:

- Q1: What is the the empirical relationship between landmark and action set distances?
- Q2: Can DLAMA find plan sets with superior landmark or action set distance, in comparison with other approaches?

Following a discussion of our experimental setup, we address these questions.

Experiment Setup

We compare DLAMA with LAMA, FF^{GrDiv} and LPG-d. Each planner is used to find plan sets with four plans. LAMA finds a single solution. Both DLAMA and FF^{GrDiv} return the first four solutions. We evaluated, but do not report on finding larger plan sets because the trends we observed were consistent across sets with both 16 and 64 plans.

LPG-d is given an average plan distance threshold that it must exceed with the set of four plans. We selected thresholds 0.01, 0.5, and 1.0 for LPG-d. Each planner on each instance is allowed 1200 seconds and 2 GB of RAM to find a plan set. Both FF^{GrDiv} and LPG-d are randomized approaches, and all results are the median of four random seeds. We collected total planning time to find all plans in a plan set, average plan length, average landmark distance, and average action set distance.

We attempted all instances of the Storage, Rovers, Driverlog, Satellite, and Depot International Planning Competition (IPC) domains (drawn from the Fast Downward distribution) and report results on instances with disjunctive landmarks. We also collected results on three artificial domains that are scalable versions of the domains presented in Section 2.

The Irrelevant Navigation (IN) domain allows irrelevant reversible actions in plans. The IN domain scales in the length of optimal (i.e., no irrelevant actions) solutions. The initial location leads to four locations. Each of these four locations leads to the same next location, or a unique irrelevant location. Repeating this fan-out and fan-in structure leads to instances with longer plans and additional disjunctive landmarks. The index of each instance from one to twenty indicates the number of disjunctive landmarks.

The Redundant Navigation (RN) domain is similar to the IN domain in that it has a fan-in and fan-out structure to increase disjunctive landmarks, but omits the unique irrelevant locations. RN includes four different actions for each state transition and scales in the number of disjunctive landmarks by repeating the problem structure.

The Non-Landmark Subgoals Navigation (NLSN) domain is a scalable version of the same domain discussed in Section 2. It involves a fan out to four locations, and then a fan in to two locations, followed by one location. In this manner, there are still disjunctive landmarks, but there are subgoals not recognized as landmarks.

Q1: Landmark vs. Action Set Distance

We show that landmark-based distance is superior to action set-based distance. Plan sets with high landmark distance will have high action set distance, but not vice versa. Through two artificial domains, we illustrate two ways that action set distance can be increased without increasing landmark distance. We observe the same trend in results from several IPC domains. The third artificial domain illustrates the weakness of landmark-based distance measures.

Irrelevant Navigation: In the IN domain, it is possible to plan reversible actions (and increase plan length) while achieving the same landmark disjuncts. Figure 4 illustrates the average landmark and action set distance between plans (left), the average plan length (center), and total time in seconds (right) of several planners. The results in this domain do not include FF^{GrDiv} because it found a set of identical plans for each instance (i.e., the average distance was zero).

We see that in this domain, it is possible to have high action set distance and relatively low landmark distance. There are no cases where landmark distance is high and action set distance is low. We see that DLAMA finds maximal (i.e., 1.0) landmark and action distance in all instances and that it finds the shortest plans. DLAMA is also relatively efficient, finding four plans in total time only marginally greater than LAMA (which finds a single solution).

LPG-d is mislead into finding solutions that use unnecessary reversible operators to increase action set distance. LPG-d solutions also exhibit increased average plan lengths.

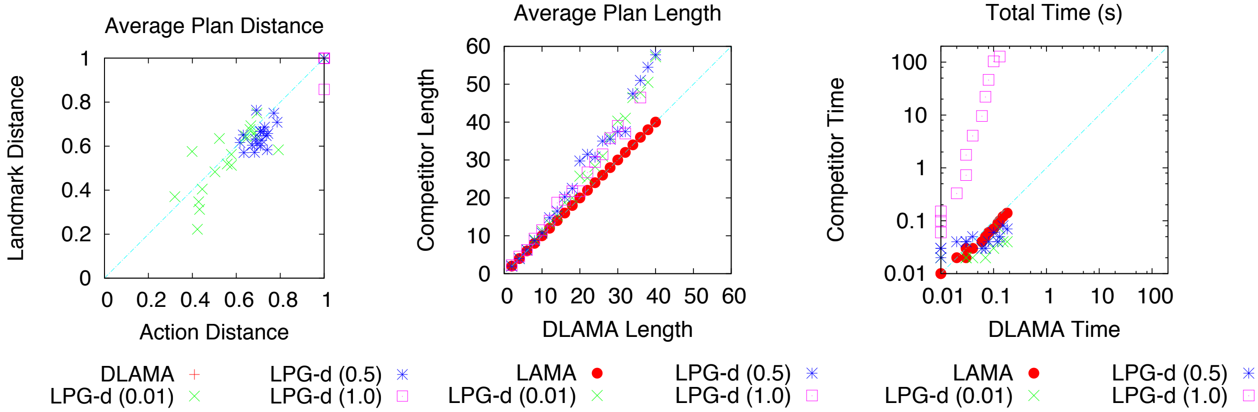

Figure 4: Average landmark-based and action set-based plan distance (left), average plan length (center), and total time in seconds (right) results on IN instances. The instances scale number of disjunctive landmarks from one to twenty.

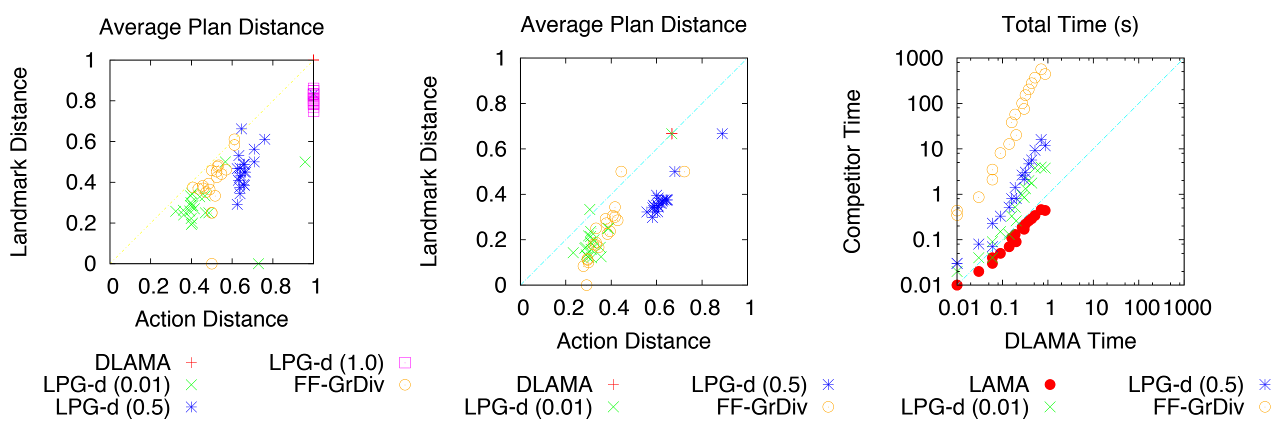

Figure 5: Average landmark-based and action set-based plan distance in RN (left) and NLSN (center), and total time in seconds (right) results on NLSN. The instances scale number of disjunctive landmarks from one to twenty.

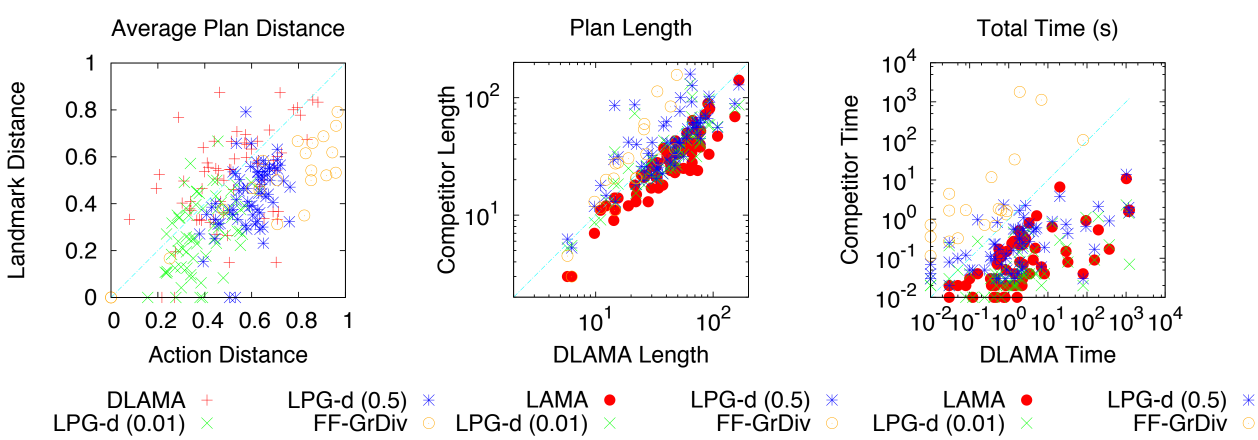

Figure 6: Average plan set distance (left), plan length (center), and total time (right) on IPC instances.

It is unsurprising that LPG selects the reversible operators over alternative paths because the plans with reversible operators are nearby the existing plans in LPG-d's local search neighborhood, but distant in terms of action sets. This mismatch between distance in the decision and objective spaces leads to poor performance in some of the larger instances, as evidenced by LPG-d's increased planning time and inability to find plan sets within its self-imposed limit on the number of local search moves. While DLAMA and LPG-d have very different search spaces, DLAMA can better shape its search space by modifying the landmark graph heuristic to seek out plans with high landmark distance.

Redundant Navigation: In the Navigation domain with redundant actions, renamed copies of identical actions can be interchanged to increase action distance while achieving the same landmark disjuncts. It is only possible to increase landmark distance by varying the locations entered, and not by varying the action used to enter the same location. Figure 5 (left plot) illustrates the average plan distance of each plan set.

DLAMA finds maximally distant plans in terms of both landmark and action set distance because satisfying different landmark disjuncts requires selecting different actions. LPG-d and FF^{GrDiv} find plan sets with lower overall landmark and action set distance because they choose to vary the actions used, but make many of the same state transitions. The total time required by each planner to solve instances is relatively small, but exhibits trends (on a smaller scale) that are similar to IN domain.

Non-Landmark Subgoals Navigation: Figure 5 illustrates results in the NLSN domain, including average plan distance (center) and total time (right). This domain allows only two plans per instance that satisfy different landmark disjuncts. This means that the average landmark-based plan distance will be less than one for any set of plans with more than two elements. It offers relatively more opportunity to increase the action set distance between plans by satisfying alternative non-landmark subgoals. Somewhat surprisingly, LPG-d and FF^{GrDiv} fail to find more diverse solutions than DLAMA (in all but two cases), and also require considerably more planning time. We would expect that LPG-d and FF^{GrDiv}, with additional opportunities for increasing action set distance, should outperform DLAMA in both the plan distance and planning time. While DLAMA diversity is degraded in this domain by design, LPG-d and FF^{GrDiv} suffer for other reasons. While LPG-d is only required to exceed a minimum threshold on plan set diversity, it exceeds this threshold by a smaller factor in NSLN than in the RN and IN domains. We believe that the results in NSLN are influenced by the relatively larger search space further challenging LPG-d and FF^{GrDiv} to find plans with distant action sets.

IPC Domains: The left-most plot in Figure 6 demonstrates the same trend in the IPC domains that is apparent in the navigation domains: it is common to have high action set distance and low landmark distance, especially with action set distance-based planners. The planners with typically better action set distance LPG-d (with 0.5 threshold) and FF^{GrDiv} also have relatively lower landmark distance. This suggests that the increase in action set distance may be due to redundant or irrelevant actions. However, we note that many domains include alternative action sequences that are not redundant or irrelevant, but pertain to a subset of the possible solutions (and are hence not captured by landmarks).

Q2: DLAMA vs. Competition

While DLAMA is not meant to optimize action set distance and LPG-d and FF^{GrDiv} are not meant to optimize landmark distance, we should expect that each planner excels at optimizing its respective distance measure. The left-most and center plots in Figure 6 illustrate that on IPC instances DLAMA tends to find the plan sets with the best average landmark distance and that their action set distance is competitive with those of LPG-d and FF^{GrDiv}. FF^{GrDiv} tends to find some of the best plan sets with high action set distance, but also scales fairly poorly. FF^{GrDiv} cannot solve many of the larger instances and exhibits a significantly higher average plan length and total planning time (center and right-most plots in Figure 6). LPG-d performs comparably to DLAMA in terms of planning time and has a sometimes higher average solution length. LAMA, which only finds a single plan, has lower plan lengths and solution time than DLAMA, but does not outscale DLAMA considerably.

8 Related Work

Diverse planning was first explored in HTN planning, where multiple domain-dependent criteria (e.g., safety, resource utilization, and speed) are associated with methods (Myers and Lee 1999). Diverse plans in this setting use alternative methods to vary these criteria, and can be seen as a similar form of systematic variation as explored in this work – where instead of the criteria, landmark disjuncts are varied. Constructing diverse plans by varying plan criteria in a domain-independent fashion has been explored as a form of multi-criteria optimization where the diverse set corresponds to a set of non-dominated plans (Bryce, Cushing, and Kambhampati 2007; Nguyen et al. 2009). Approaches that vary plan quality criteria have been typically studied in isolation from techniques that seek diversity in plan structure.

Plan structure diversity was first explored by Nguyen et al. (2012), including distance measures based on action sets, causal links, and states. More recent work studies how plan structure diversity can be expressed through both quantitative and qualitative distance measures (Coman and Muñoz-Avila 2011).

We compared landmark-based distance with action set-based distance measures. We would like to note that other syntactic distance measures share the same problems in comparison to landmark distance. The primary issue is that of *relevance*. Landmarks are relevant to plan correctness and hence capture plan semantics. States, causal links, and actions may be present in plans but have no bearing on goal achievement. Without relevance to goal achievement, syntactic distance measures can result in plan sets that vary non-essential aspects of plans.

9 Conclusion

We propose a new method for constructing diverse plans that is based upon satisfying different landmark disjuncts. We have demonstrated that using this distance measure to construct a diverse set of plans has several advantages over prior syntactic plan distance measures. Landmark-based distance is less susceptible to biasing planners to plan redundant or irrelevant actions. Plan sets with high average landmark distance also typically have high action set distance, but the reverse does not hold.

Our diverse planner DLAMA is conceptually simple and can be easily parallelized. DLAMA is also built upon the state of the art LAMA planner, and demonstrates reasonable scalability as a result. The main drawback of DLAMA is that it can only find diverse plan sets if given disjunctive landmarks. While DLAMA can be hybridized with other diverse planners to overcome this limitation, we are hopeful that our work will motivate new approaches to discovering landmarks.

References

Bryce, D.; Cushing, W.; and Kambhampati, S. 2007. Model-lite planning: Diverse mult-option plans and dynamic objective functions. In *Proceedings of the 3rd Workshop on Planning and Plan Execution for Real-World Systems*.

Coman, A., and Muñoz-Avila, H. 2011. Generating diverse plans using quantitative and qualitative plan distance metrics. In Burgard, W., and Roth, D., eds., *National Conference on Artificial Intelligence (AAAI-11).* AAAI Press.

Gerevini, A.; Saetti, A.; and Serina, I. 2003. Planning through stochastic local search and temporal action graphs in lpg. *J. Artif. Intell. Res. (JAIR)* 20:239–290.

Hoffmann, J., and Nebel, B. 2001. The ff planning system: Fast plan generation through heuristic search. *J. Artif. Intell. Res. (JAIR)* 14:253–302.

Hoffmann, J.; Porteous, J.; and Sebastia, L. 2004. Ordered landmarks in planning. *J. Artif. Intell. Res. (JAIR)* 22:215–278.

Kambhampati, S., and Kedar, S. 1994. A unified framework for explanation-based generalization of partially ordered and partially instantiated plans. *Artificial Intelligence* 67(1):29–70.

Karpas, E., and Domshlak, C. 2009. Cost-optimal planning with landmarks. In *IJCAI*, 1728–1733.

Malaiya, Y. 1995. Antirandom testing: getting the most out of black-box testing. In *Software Reliability Engineering, 1995. Proceedings., Sixth International Symposium on*, 86–95.

Myers, K., and Lee, T. 1999. Generating qualitatively different plans through metatheoretic biases. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence*. AAAI Press.

Nguyen, T. A.; Do, M. B.; Kambhampati, S.; and Srivastava, B. 2009. Planning with partial preference models. In Boutilier, C., ed., *IJCAI*, 1772–1777.

Nguyen, T. A.; Do, M. B.; Gerevini, A.; Serina, I.; Srivastava, B.; and Kambhampati, S. 2012. Generating diverse plans to handle unknown and partially known user preferences. *Artif. Intell.* 190:1–31.