

DALAI – Disjunctive Action Landmarks All In

Clemens Büchner¹, Remo Christen¹, Salomé Eriksson¹, Thomas Keller^{1,2}

¹University of Basel, Switzerland

²University of Zürich, Switzerland

{clemens.buechner, remo.christen, salome.eriksson, tho.keller}@unibas.ch

This is the planner abstract for DALAI, a reincarnation of LAMA (Richter and Westphal 2010). DALAI uses related concepts, in particular landmarks, but adds recent findings to the mix and implements most landmark-specific parts from scratch. Moreover, it goes all in betting on landmarks and rejects complementary ideas. Our implementation is based on version 22.12 of Fast Downward (Helmert 2006).

Mythology

The first written evidence of the term *landmark* in the planning context can be found in an ancient manuscript that dates back to the very early days of plannerkind (Porteous, Sebastia, and Hoffmann 2001). These early plan-hunters and gatherers observe the existence of (*fact*) *landmarks* (also called *atoms*) that are encountered in any successful plan-chase, and they use *orderings* among the landmarks to decompose the cumbersome hunt into subproblems that are easier to tackle and have the landmarks as subgoals. However, as this hunting technique is incomplete, they often returned without success despite the presence of plans in their surroundings.

Another scroll (Zhu and Givan 2003), which was written hundreds of days later, indicates that the next step of plannerkind was to raise the concept from its initial, fact-based representation to a methodology that describes which *activities* (or *actions*) are necessary in any successful plan-gathering. The same scroll also describes that those early planners no longer used landmarks for decomposition purposes, but to come up with an estimate of the effort that is required until the plan can be reaped successfully. Further artifacts of these ancient technologies provide evidence that the ideas were expanded to disjunctive sets of atoms (Richter, Helmert, and Westphal 2008), conjunctive sets of atoms (Keyder, Richter, and Helmert 2010), arbitrary propositional formulas over atoms (Karpas and Domshlak 2009), or disjunctive sets of actions (Bonet and Helmert 2010).

The rise of landmarks on the planning horizon climaxed with the creation of the LAMA planner (Richter and Westphal 2010), which emerged victorious from the international planning competition (IPC) 2008 and certified its dominance in the year 2011 with another victory. Nowadays, over a decade after the reign of LAMA, its spirit lives on while its physical remains (i.e., the source code) could not keep up with the advancement of time. During these years, many other, much younger spirits started to show their powers,

leaving LAMA, as legendary as the stories are, a mere memory of today’s planning society.

Nevertheless, researchers from all over the world continued to study the topic of landmarks; recent findings suggest that LAMA did—despite its success—not live up to its full potential. Here, we describe our aim at taking a step on this unsatisfactory and painful path to true enlightenment, which manifests in the LAMA reincarnation DALAI.

Ideology

*The goal is not to be better than the other ~~man~~ [planner],
but your previous self.*

—Dalai Lama

With DALAI, we revive the spirit of LAMA in a new shape. Our planner is inspired by the many concepts responsible for LAMA’s success, first and foremost the use of landmarks. However, we revise crucial parts of LAMA’s implementation such as the representation of landmarks; in LAMA, a landmark is a set of atoms such that for every plan it holds that one atom from every landmark is true in some state along this plan’s trajectory. In contrast, DALAI considers sets of actions, such that every plan contains at least one action from every landmark. The usage of these so-called *disjunctive action landmarks* provides the first three letters of the name of our planner.

The last two letters of the planner name stem from our commitment on creating a *pure landmark-based planner* rather than a portfolio of successful techniques: DALAI goes *all-in* and renounces all components of LAMA that are not related to landmarks. Specifically, LAMA’s best-first search alternates between two open lists whose order is based on two different heuristics, namely the landmark count heuristic and the delete-relaxation based FF heuristic (Hoffmann and Nebel 2001). While techniques that are based on the idea of a delete-relaxed task are among the most important methods to generate landmarks (Zhu and Givan 2003; Richter, Helmert, and Westphal 2008), the FF heuristic itself is not related to landmarks. Therefore, all versions of DALAI are guided by a single heuristic that is based on landmarks.

Technology

Optimism doesn't mean that you are blind to the reality of the situation. It means that you remain motivated to seek a solution to whatever problems arise.

—Dalai Lama

Büchner et al. (2023) suggest the LM-BFS framework and describe how the landmark component of LAMA is an instantiation of it. DALAI also fits that framework which describes a best-first search based on landmarks. The LM-BFS framework offers several degrees of freedom to achieve different behaviour of the search: (a) landmark generation method, (b) landmark progression function, (c) landmark heuristic, and (d) search algorithm. We continue by briefly discussing these components and related topics, focusing on the context of DALAI. A discussion of the explicit choices we made for these components for the submission of DALAI to the different tracks of IPC 2023 follows in the next section. Note that Büchner et al. (2023) restrict their discussion of LM-BFS to landmarks as propositional formulas over state features (i.e., atoms). Since DALAI operates on disjunctive action landmarks, though, we need to transfer the theory to this view.

Landmark Generation

In order to become prosperous, a person [planner] must initially work very hard, so he or she [it] has to sacrifice a lot of leisure time.

—Dalai Lama

Hoffmann, Porteous, and Sebastia (2004) show that deciding whether a given atom is a landmark is PSPACE-complete. Nevertheless, there are efficient algorithms that allow to come up with some specific landmarks. In DALAI we use the technique proposed by Richter, Helmert, and Westphal (RHW, 2008) which computes landmarks in the delete relaxation and additionally determines *natural* and *greedy-necessary orderings* between these landmarks. In a subsequent step, we add *reasonable orderings* the same way as Hoffmann, Porteous, and Sebastia (2004) do.

Fast Downward implements both of these techniques in a way that generates disjunctive fact landmarks. It is possible to transform a disjunctive fact landmark ℓ into a disjunctive action landmark by considering the set of *achieving actions* of each atom in ℓ . Indeed, this is what we do in some configurations. Alternatively, we implemented a direct computation of both disjunctive action landmarks and orderings between them.

In both cases, the reasoning for greedy-necessary orderings needs special care. By definition, a greedy-necessary ordering between landmarks l_1 and l_2 denotes that l_2 can only be achieved from states where l_1 holds. Since it is not properly defined for a disjunctive action landmark what it means for l_1 to hold, we need to work around this somehow. Our solution is to store for all such orderings the set of atoms l_1 achieves.

Our variant computing disjunctive action landmarks directly adapts several design aspects to better fit the action landmark world, as well as to increase the amount of landmarks and orderings found in the hope of providing better guidance during the search.

Allow overlapping landmarks. RHW enforce that $l_i \cap l_j = \emptyset$ for all considered fact landmarks l_i and l_j . We lift this restriction for two reasons: first, the translation to disjunctive action landmarks already leads to overlap (for example if we have two fact landmarks $\{v_1\}$ and $\{v_2\}$ and an action a achieving both, then the action landmarks derived from $\{v_1\}$ and $\{v_2\}$ both contain a); and second, we consider different ways of computing a heuristic value from landmarks, some of which do not rely on mere *counting* of unachieved landmarks (for which we assume that overlap-free landmarks are particularly well-suited).

Allow larger shared preconditions. RHW limit the size of shared preconditions to 4, while we turned the limit into an adjustable parameter.

Detect more cases of interfering landmarks. The procedure by Hoffmann, Porteous, and Sebastia (2004) deduces reasonable orders when landmarks *interfere* with each other. The implementation in Fast Downward requires one of the disjunctive fact landmarks to consist of only one fact, but we allow arbitrary disjunctive action landmarks.

Allow cycles in the landmark graph. LAMA enforces that no cyclic dependencies between landmarks occur, while DALAI does not require this restriction.

Progression

Reason well from the beginning and then there will never be any need to look back with confusion and doubt.

—Dalai Lama

Computing good landmarks and orderings is prohibitively expensive. Therefore, the LM-BFS framework by Büchner et al. (2023) advocates computing landmarks once for the initial state and *progressing* them within the search whenever expanding new transitions. This idea was first suggested in the context of the path-dependent landmark heuristic (Richter, Helmert, and Westphal 2008) and later strengthened by combining information from multiple paths (Karpas and Domshlak 2009).

Since Büchner et al. (2023) use landmarks based on propositional formulas over state features, their landmark progressions reason about whether a certain landmark holds in a given state or not. We consider disjunctive action landmarks, where things are not as expensive. Instead, we just need to know which landmarks a given action satisfies. This is easy to test, as these are simply the landmarks that contain that action. In the terminology of Büchner et al., we mark a landmark *past* when applying one of its operators. Landmark orderings of a certain type mark landmarks *future* according to the same rules used by Büchner et al..

Heuristics

Although you may not always be able to avoid difficult situations, you can modify the extent to which you can suffer by how you choose to respond to the situation.

—Dalai Lama

Zhu and Givan (2003) suggest to count the landmarks that need to be achieved in a given state as heuristic for the goal distance. This heuristic became known as the *landmark*

count heuristic and is also used in LAMA. Its implementation in Fast Downward was recently renamed to *landmark sum* (h^{sum}), as it was changed to take action costs of the achievers into account and is therefore not purely counting landmarks anymore. For this heuristic, it does not matter what kinds of landmarks are considered, so its idea carries over easily to the disjunctive action landmark view.

However, the more the (action) landmarks overlap, the less accurate h^{sum} becomes since it does not account for achieving several landmarks with the same action. We thus implemented the *greedy hitting set heuristic* (h^{ghs}), which greedily computes a hitting set, i.e., a set of actions A such that for all landmarks l we have $l \cap A \neq \emptyset$ (the landmark is “hit” by A). The heuristic is greedy in that it iteratively adds an action to A that maximizes the amount of landmarks that newly “are hit”. Since the hitting set is not *minimal*, this still leads to inadmissible estimates and is thus only used in satisficing settings.

Computing *minimum* hitting sets would denote an admissible heuristic. However, Büchner, Keller, and Helmert (2021) show that we can do even better using the *cyclic landmark heuristic* (h^{cycle}). Besides the landmarks, h^{cycle} also considers the orderings between them, in particular the cyclic dependencies induced by the orderings. Büchner, Keller, and Helmert prove that one of the landmarks in every cycle needs to be achieved at least twice, and that their heuristic dominates the minimum hitting set approach.

Preferred Operators

In order to carry out a positive action we must develop a positive vision.

—Dalai Lama

Analogous to LAMA, we use alternating open lists where one is populated by preferred operators. In our disjunctive action landmark framework, the definition of preferred operators is straightforward: an operator is preferred if it is both applicable in the current state, and is contained in a landmark that must still be satisfied.

To choose between the open lists, we use the boost mechanism of LAMA. Instead of using the default boost value of 1000, we individually adapt the value for the two non-optimal tracks based on empirical results.

Justified Action Applications

The important thing is that ~~men~~ [actions] should have a purpose in life. It should be something useful, something good.

—Dalai Lama

The notion of *unjustified* action applications stems from Simon and Röger (2015), who in turn based the idea on intended effects introduced by Karpas and Domshlak (2012).

Intuitively, an action application is only justified if at least one of its add effects satisfies the following two conditions: (i) it is part of the goal or enables a future action application, and (ii) it is neither deleted nor made true by a future action application until the goal is reached or the enabled action application is performed. Any non-zero-cost action application that is not justified according to these criteria can never be

part of an optimal plan, thus we can safely disregard paths that contain them.

We integrate a relaxed version of this idea into DALAI by casting Condition (i) into a disjunctive action landmark. For every action a whose effects do not contain a goal atom, we compute the set of actions S_a such that every action in S_a has at least one of a ’s add effects as a precondition. Each set S_a forms a disjunctive action landmark that is active on every path that contains action a and thus ensures that the application of a satisfies Condition (i). In other words, as soon as an action is applied we enforce that its application must later be justified, according to our relaxed notion, by a future applied action.

As the resulting landmarks can contain thousands of actions and thus become both large and uninformative, we implement an ad hoc optimization that discards all landmarks that contain more than 1000 actions.

Methodology

Action is more important than meditation.

—Dalai Lama

We participate with DALAI in all three tracks of IPC 2023. However, we use different versions tailored to the different objectives for each track. The following objectives are used in the competition:

Optimal Find optimal plans for as many problems as possible.

Satisficing Find plans for as many problems as possible, aiming for cheap plans.

Agile Find plans for as many problems as possible as fast as possible.

We experimentally tested different combinations of concrete implementations and parameter choices on the benchmarks from previous iterations of the IPC. The following paragraphs describe the configurations as submitted to the competition and some words of wisdom on why we think they might be suitable for the corresponding objectives.

Optimal We use A* search (Hart, Nilsson, and Raphael 1968) with h^{cycle} as an admissible heuristic (Büchner, Keller, and Helmert 2021). Landmarks and orderings are generated using the LM-cut procedure¹ (Helmert and Domshlak 2009; Bonet and Helmert 2010) as well as the RHW landmark generation method (Richter, Helmert, and Westphal 2008). For the latter, we use the original implementation based on fact landmarks and translate the result to disjunctive action landmarks. We use Johnson’s cycle detection algorithm (Johnson 1975). Note that this configuration does not support conditional effects and aborts without finding a solution when parsing a problem file with conditional effects.

Satisficing We follow LAMA which uses an iterative *any-time search* procedure (Richter and Westphal 2010). It sequentially starts one search after the other on the same input, using the solution cost found by the first search as an upper bound on the solution cost for the second one and so on. The

¹only landmarks, no orderings

procedure stops when either time or memory runs out, an iteration cannot find an improving solution compared to the previous one, or the last iteration found a plan. In any case, the last plan found (if any) is reported as the final solution.

As a heuristic, this configuration uses h^{ghs} with preferred operators and a boost value of 1. Conditional effects are supported. Landmarks are generated using our disjunctive action landmark variant of RHW. Furthermore, some iterations of the search consider justified action applications. We start with a lazy greedy best-first search without justified action application to find *some* plan quickly. In tasks with non-uniform action costs, we continue with a lazy greedy best-first search with action costs considered, but shifted by +1 to prefer short plans over long ones with 0-cost actions. Next, it follows a sequence of weighted A^* searches with declining weights (5, 3, 2, 1). Action costs are again transformed in the same manner in case of non-uniform action costs. In case of uniform action costs, the search stops at this point. Otherwise, we append a last iteration of A^* with the original action costs.

Agile Since this track is all about speed, we use h^{sum} as the heuristic which is much faster to evaluate than h^{ghs} or h^{cycle} . Conditional effects are supported. Furthermore, we transform costs to unit-costs to find short paths rather than cheap ones. The search is lazy greedy best-first search and considers preferred operators but without boosting them upon progress in the heuristic value.

Apology

Please excuse inaccuracies due to entertainment purposes.

Acknowledgy

We thank all Fast Downward contributors, and in particular we thank the developers of LAMA for inspiration.

References

Bonet, B.; and Helmert, M. 2010. Strengthening Landmark Heuristics via Hitting Sets. In Coelho, H.; Studer, R.; and Wooldridge, M., eds., *Proceedings of the 19th European Conference on Artificial Intelligence (ECAI 2010)*, 329–334. IOS Press.

Büchner, C.; Eriksson, S.; Keller, T.; and Helmert, M. 2023. Landmark Progression in Heuristic Search. In Koenig, S.; Stern, R.; and Vallati, M., eds., *Proceedings of the Thirty-Third International Conference on Automated Planning and Scheduling (ICAPS 2023)*. AAAI Press.

Büchner, C.; Keller, T.; and Helmert, M. 2021. Exploiting Cyclic Dependencies in Landmark Heuristics. In Goldman, R. P.; Biundo, S.; and Katz, M., eds., *Proceedings of the Thirty-First International Conference on Automated Planning and Scheduling (ICAPS 2021)*, 65–73. AAAI Press.

Hart, P. E.; Nilsson, N. J.; and Raphael, B. 1968. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2): 100–107.

Helmert, M. 2006. The Fast Downward Planning System. *Journal of Artificial Intelligence Research*, 26: 191–246.

Helmert, M.; and Domshlak, C. 2009. Landmarks, Critical Paths and Abstractions: What’s the Difference Anyway? In Gerevini, A.; Howe, A.; Cesta, A.; and Refanidis, I., eds., *Proceedings of the Nineteenth International Conference on Automated Planning and Scheduling (ICAPS 2009)*, 162–169. AAAI Press.

Hoffmann, J.; and Nebel, B. 2001. The FF Planning System: Fast Plan Generation Through Heuristic Search. *Journal of Artificial Intelligence Research*, 14: 253–302.

Hoffmann, J.; Porteous, J.; and Sebastia, L. 2004. Ordered Landmarks in Planning. *Journal of Artificial Intelligence Research*, 22: 215–278.

Johnson, D. 1975. Finding all the Elementary Circuits of a Directed Graph. *SIAM Journal on Computing*, 4(1): 77–84.

Karpas, E.; and Domshlak, C. 2009. Cost-Optimal Planning with Landmarks. In Boutilier, C., ed., *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI 2009)*, 1728–1733. AAAI Press.

Karpas, E.; and Domshlak, C. 2012. Optimal Search with Inadmissible Heuristics. In McCluskey, L.; Williams, B.; Silva, J. R.; and Bonet, B., eds., *Proceedings of the Twenty-Second International Conference on Automated Planning and Scheduling (ICAPS 2012)*, 92–100. AAAI Press.

Keyder, E.; Richter, S.; and Helmert, M. 2010. Sound and Complete Landmarks for And/Or Graphs. In Coelho, H.; Studer, R.; and Wooldridge, M., eds., *Proceedings of the 19th European Conference on Artificial Intelligence (ECAI 2010)*, 335–340. IOS Press.

Porteous, J.; Sebastia, L.; and Hoffmann, J. 2001. On the Extraction, Ordering, and Usage of Landmarks in Planning. In Cesta, A.; and Borrajo, D., eds., *Proceedings of the Sixth European Conference on Planning (ECP 2001)*, 174–182. AAAI Press.

Richter, S.; Helmert, M.; and Westphal, M. 2008. Landmarks Revisited. In *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence (AAAI 2008)*, 975–982. AAAI Press.

Richter, S.; and Westphal, M. 2010. The LAMA Planner: Guiding Cost-Based Anytime Planning with Landmarks. *Journal of Artificial Intelligence Research*, 39: 127–177.

Simon, S.; and Röger, G. 2015. Finding and Exploiting LTL Trajectory Constraints in Heuristic Search. In Lelis, L.; and Stern, R., eds., *Proceedings of the Eighth Annual Symposium on Combinatorial Search (SoCS 2015)*, 113–121. AAAI Press.

Zhu, L.; and Givan, R. 2003. Landmark Extraction via Planning Graph Propagation. In *ICAPS 2003 Doctoral Consortium*, 156–160.