An Empirical Study of the Usefulness of State-Dependent Action Costs in Planning

Sumitra Corraya[1], Florian Geißer[2], David Speck[1], and Robert Mattmüller[1]

[1] University of Freiburg, Germany
{corrayas,speckd,mattmuel}@informatik.uni-freiburg.de
[2] Australian National University, Canberra, Australia
florian.geisser@anu.edu.au

Abstract. The vast majority of work in planning to date has focused on state-independent action costs. However, if a planning task features state-dependent costs, using a cost model with state-independent costs means either introducing a modeling error, or potentially sacrificing compactness of the model. In this paper, we investigate the conflicting priorities of modeling accuracy and compactness empirically, with a particular focus on the extent of the negative impact of reduced modeling accuracy on (a) the quality of the resulting plans, and (b) the search guidance provided by heuristics that are fed with inaccurate cost models. Our empirical results show that the plan suboptimality introduced by ignoring state-dependent costs can range, depending on the domain, from inexistent to several orders of magnitude. Furthermore, our results show that the impact on heuristic guidance additionally depends strongly on the heuristic that is used, the specifics of how exactly the costs are represented, and whether one is interested in heuristic accuracy, node expansions, or overall runtime savings.

Keywords: Planning · State-dependent costs · Heuristic accuracy.

1 Introduction and Background

State-dependent action cost (SDAC) models can provide more compact representations of planning tasks than unit-cost or constant-cost models [7]. Among many other applications, they are useful to model state-dependent penalties for unsupplied power lines in the power supply restoration domain [14]. Research has been done in an attempt to plan with state-dependent action costs both using explicit-state search [7, 8, 15, 6] and symbolic search [17].

Geißer et al. [7, 6] studied compilations of tasks with state-dependent costs into tasks with state-independent costs. The exponential compilation replaces each action with a collection of actions, including one action for each possible valuation of the state variables on which the original action cost function depends. The EVMDD-based compilation, which uses edge-valued multi-valued decision diagrams [2] to represent action cost functions, also replaces each action with a collection of actions, which in this case encode a simulation of the evaluation of the original cost function, using as few new actions as possible. Both compilations preserve plan costs, and both are worst-case exponential in the number of relevant state variables. The EVMDD-based compilation is often smaller, though, whereas the exponential compilation is necessarily exponential.

Previous work mostly focused on the representational power of the EVMDD-based compilation and on the question to what extent heuristic values are preserved under this compilation [6]. The question of how bad it is to ignore state-dependent costs altogether was not in the focus of that work. In the present paper, we will address this question, or more specifically, two sub-questions: in Section 2, we empirically study the plan suboptimality caused by ignoring state-dependent costs, and in Section 3, we study the impact of various ways of dealing with state-dependent costs, including ignoring them, on heuristic quality and related measures such as node expansions and search time.

Related work includes literature on diverse action costs [4]. Note that planning tasks with diverse and with state-dependent action costs both induce edge-weighted transition systems and differ only in their compact representations – diverse costs are a special case of state-dependent costs. For the former, Fan et al. [4] proved a no-free-lunch theorem stating that, depending on the specifics of the task, diverse costs can be either beneficial or harmful for search. Our empirical results in Section 3 confirm their result empirically in the setting of state-dependent costs. Our results are also in line with those of Ivankovic et al. [14], who use a different encoding of state-dependent costs and additionally support state constraints.

2 Sub-Optimality of Ignoring SDAC During Search

For a formal definition of planning tasks with state-dependent costs, we refer the reader to the literature [7]. For this exposition, suffice it to say that every planning task comes with a cost function $c : A \times S \to \mathbb{Q}_{\geq 0}$ that maps every action $a \in A$ and every state $s \in S$ to a value $c(a, s)$, the cost of a in s. Let $u : A \times S \to \mathbb{Q}_{\geq 0}$ be the *unit-cost function* with $u(a, s) = 1$ for each $a \in A$ and $s \in S$. The cost $\mathcal{C}^c(\pi)$ of a plan π is the sum of action costs along the execution trace of π, where c is the cost function that is used to evaluate the cost of each plan step. Furthermore, let $OPT^c(\Pi)$ be the set of cost-optimal plans for task Π, where optimality is with respect to cost function c. Unit-cost optimal plans are not necessarily optimal for other cost functions, i.e. $\mathcal{C}^c(\pi') \geq \mathcal{C}^c(\pi)$ for all $\pi' \in OPT^u(\Pi)$ and $\pi \in OPT^c(\Pi)$, where c is the original cost function of Π. Clearly, for some Π the inequality can be strict. Note that $\mathcal{C}^c(\pi)$ is identical for all $\pi \in OPT^c(\Pi)$, since the same cost function c is used both in determining optimality and in the evaluation of $\mathcal{C}^c(\pi)$. Therefore, if c is the original cost function of Π, we also refer to this value as $\mathcal{C}^*(\Pi)$, or $\mathcal{C}^*$, if Π is clear from context. By contrast, $\mathcal{C}^c(\pi')$ is *not* necessarily identical for all $\pi' \in OPT^u(\Pi)$, since plan optimality is with respect to unit costs, whereas evaluation is with respect to the true original costs. An extreme case where this happens is the Travelling Salesman domain. Here, the planner has to solve a travelling salesman problem and the cost function to move between cities is based on the Manhattan distance. There can be a unique optimal plan with respect to the true cost function c, whereas all $n!$ orders of visiting the n cities are optimal under the assumption of unit costs u. For this reason, in the empirical results presented below, there may be more than one data point per planning task: there is one data point for each $\pi' \in OPT^u(\Pi)$, relating $\mathcal{C}^c(\pi')$ to $\mathcal{C}^*(\Pi)$. Treating the cost of unit-cost optimal plans as a random variable depending on Π, we also refer to $\mathcal{C}^c(\pi')$ as $\mathcal{U}^*(\Pi)$, or just $\mathcal{U}^*$, below. Notice that in the TRAVELLING SALESMAN domain, the relative error $\mathcal{C}^c(\pi')/\mathcal{C}^*(\Pi)$ for $\pi' \in OPT^u(\Pi)$ can be made arbitrarily large by choosing large enough distance between cities on suboptimal tours, as those distances determine the costs c of moving between cities.

Whereas all of the above is clear *theoretically*, the purpose of this section is to study *empirically* how large the relative error becomes in the benchmark domains used in planning with state-dependent action costs. All plans in this section were computed using the planning system SYMPLE[3] [18, 17], which performs symbolic bidirectional breadth-first search. For our evaluation, we used 206 tasks from eight domains introduced by Speck et al. [17][4]. All experiments were run on a 3.3 GHz machine with 16 GB memory limit, with a runtime limit of 30 minutes.

Before studying the relative error, let us discuss why one would want to plan under a unit-cost assumption instead of using the true cost function in the first place. An argument might be that, by planning under the assumption of unit costs, one is able to solve more tasks. The question is whether the original state-dependent action costs guide the search to the goal or not. Depending on the domain, this is indeed the case. Whereas coverage is unaffected by the unit-cost transformation in the domains ASTERIX (30 vs. 30 in 30 problems solved), PEG SOLITAIRE-08 (27 vs. 27 in 30) and PEG SOLITAIRE-11 (17 vs. 17 in 20), and even slightly decreases in OPENSTACKS-08 (15 vs. 12 in 30), OPENSTACKS-11 (20 vs. 16 in 20), and OPENSTACKS-14 (7 vs. 4 in 20), it increases in COLORED GRIPPER (10 vs. 12 in 30), and, most pronouncedly it doubles in TRAVELLING SALESMAN (13 vs. 26 in 26). This makes intuitively sense, as ignoring state-dependent costs in TRAVELLING SALESMAN essentially turns an optimization problem into a satisfaction problem, which makes it easy to solve.

The relative error introduced by ignoring state-dependent costs is depicted in Figure 1. Each dot represents a pair $(\mathcal{C}^*(\Pi), \mathcal{U}^*(\Pi))$ for one task Π solved in both configurations. The error bars in the $\mathcal{U}^*(\Pi)$ dimension are owed to the above-mentioned fact that different optimal plans under the assumption of unit costs can have different true costs under the original state-dependent cost function. The dots represent the mean values, the error bars the standard deviations. Wherever possible, to compute means and standard deviations, we took all plans $\pi' \in OPT^u(\Pi)$ into account. If, the cardinality of $OPT^u(\Pi)$ grew prohibitively, we resorted to a representative sample from $OPT^u(\Pi)$.

[3] `https://gkigit.informatik.uni-freiburg.de/dspeck/symple`

[4] `https://gkigit.informatik.uni-freiburg.de/dspeck/SDAC-Benchmarks`

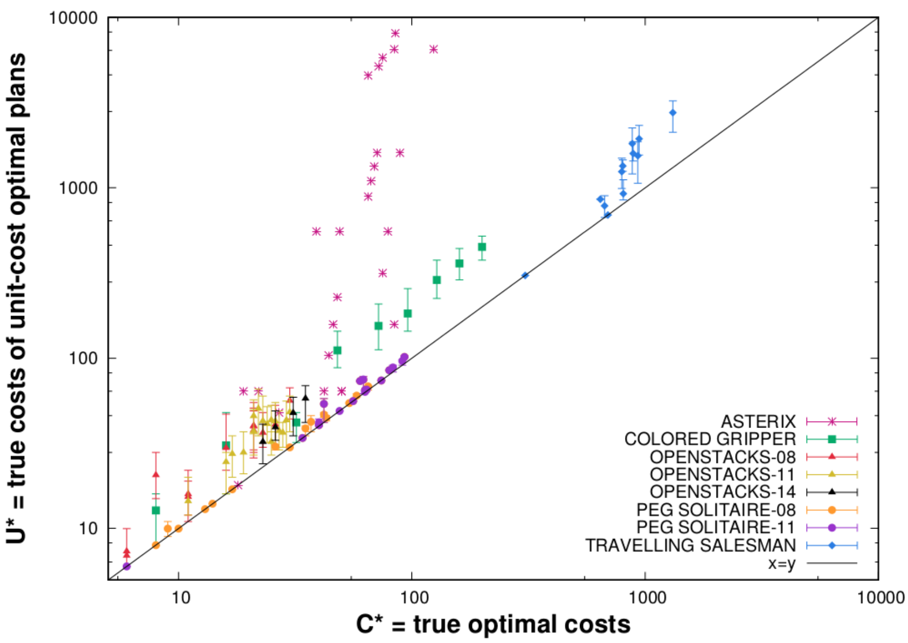

Fig. 1: Scatter plot relating $\mathcal{C}^*$ to $\mathcal{U}^*$.

Generally, the results indicate that sometimes, ignoring state-dependent costs is largely unproblematic in terms of optimality (Peg Solitaire), whereas in Openstacks, Colored Gripper, and Travelling Salesman we see errors of an order of magnitude. In the Asterix domain, the worst-case error ranges up to three orders of magnitude and is clearly not negligible.

3 Sub-Optimality of Ignoring SDAC in Heuristics

While searching for unit-cost optimal plans leads to arbitrarily suboptimal plans, computing heuristics based on unit-cost cost functions preverves plan optimality, as long as the heuristic is still admissible [5, 16]. In this section, we investigate therefore the effects of different ways of dealing with state-dependent costs on the informativeness of several goal-distance heuristics and on the guidance they provide to the search (in terms of numbers of node expansions). As different ways of dealing with state-dependent costs we consider: (a) the exponential compilation [6], (b) the EVMDD-based compilation [6], (c) replacing all costs by u, and (d) replacing all costs $c(a, s)$ by $\min_{s' \in S} c(a, s')$. Transformation (d), to which we will also refer as the *minimum transformation*, is similar in simplicity to (c), but unlike in (c), different actions can still have different constant costs, and as a consequence of taking minimal costs, admissibility of heuristics is preserved under transformation (d). This is not generally guaranteed with transformation

(c) if the task contains actions with cost less than one. Notice that (a) and (b) are "lossless", where as (c) and (d) are "lossy" transformations. This section complements Geißer's study [6], which investigated invariance of various heuristics under transformations (a)–(d) theoretically and empirically, by considering additional heuristics not studied there. In particular, we considered (a) the maximum heuristic $h^{\max}$ [1], (b) the incremental pattern database heuristic h^{iPDB} [9], (c) the merge-and-shrink heuristic $h^{\mathrm{M\&S}}$ [12], and (d) the landmark-cut heuristic $h^{\text{LM-cut}}$ [11]. We chose those four heuristics as all of them are implemented in the Fast Downward [10] planner that we use in this experiment, and all of them are admissible.

For the sake of brevity, in the following, we will focus on the results for one specific heuristic, $h^{\mathrm{M\&S}}$, exemplarily. Recall that $h^{\mathrm{M\&S}}$ is an abstraction heuristic that builds the abstract transition system that is used for goal-distance estimation incrementally, starting with atomic abstractions, one per state variable, and keeps *merging* abstract transition systems into larger ones until a critical size has been reached. Merging is done by computing synchronized products of the involved transition systems and leads to finer, more informative, abstractions. Once the critical size has been reached, abstractions get *shrunk*, i. e., abstractions are made coarser, hence less informative, by lumping abstract states together.

Since the EVMDD-based and the exponential compilation preserve plan costs, $h^{\mathrm{M\&S}}$ computed on either of those compilations is still admissible with respect to the original cost function. Similarly, since the minimum transformation only leads to underestimation of the true actions costs, $h^{\mathrm{M\&S}}$ computed on the minimum transformation also remains admissible. Hence, plans computed by A* search in all $h^{\mathrm{M\&S}}$ configurations are still optimal. The only configuration that is not guaranteed to be optimal is $h^{\mathrm{M\&S}}$ together with the unit-cost transformation, in case the original task contains actions with costs less than one in some states. This is the case with the Openstacks domain, where actions have a cost of zero in some states. Indeed, $h^{\mathrm{M\&S}}$ with the unit-cost transformation leads to slightly suboptimal plans. Note that we used "standard" merge-and-shrink, not the version with delta cost partitioning of Fan el al. [3].

To assess how informative $h^{\mathrm{M\&S}}$ is combined with the four task transformations, we first measured the initial heuristic values in all cases. The exponential compilation tends to lead to the most informative heuristic values, especially in the Travelling Salesman domain. However, informative heuristic values are only a means to another end: few node expansions, and ultimately low runtime and high coverage. Numbers of node expansions are depicted in Figure 2. As can be seen, the exponential compilation often leads to the fewest node expansions, especially in the Travelling Salesman domain. Yet, overall, the exponential compilation solves the fewest tasks, since the compiled task is often too large to be generated. The unit-cost transformation tends to lead to slightly fewer node expansions than the minimum-cost transformation and the EVMDD-based compilation, but this comes at the cost of suboptimality of some plans. The two transformations without those obvious flaws (unacceptable problem size increase, inadmissibility of heuristic), the minimum-cost transformation (simple, but lossy) and the EVMDD-based compilation (complicated, but lossless), are remarkably similar in terms of node expansions. The minimum transformation does better in a few OPENSTACKS tasks, whereas the EVMDD-based compilation does better in the smaller TRAVELLING SALESMAN tasks. There, the resulting heuristic of the initial state is almost perfect. Only in the larger TRAVELLING SALESMAN instances the performance tends to degrade, as the M&S abstractions get shrunk and shortcuts are introduced in the abstract transition systems.

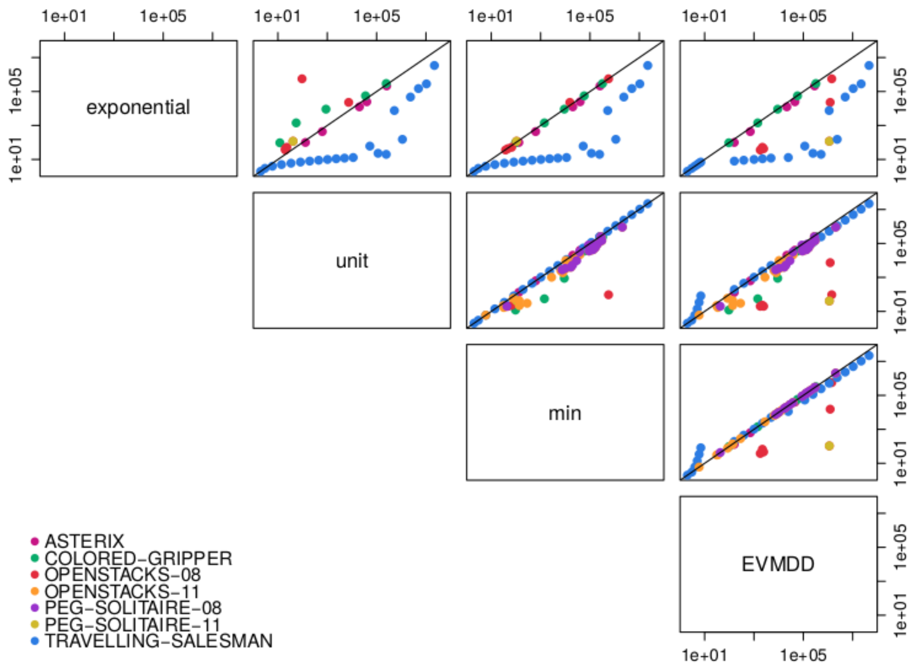

Fig. 2: Matrix of scatter plots showing comparison of node expansions with M&S heuristic under different transformations. All axes denote node expansions.

The difference in complexity of computing the various transformations (exponential is most costly, followed by EVMDD-based, minimum-cost, and unit-cost) and in informativeness of the compilations (exponential is most informative, usually followed by EVMDD-based, minimum-cost, and unit-cost) also results in different overall numbers of solved tasks: 41 (exponential), 87 (EVMDD-based), 88 (minimum-cost), and 91 (unit-cost). The picture is similar with the other heuristics we investigated. While these results give the impression that more complex transformations do not pay off, we have to note that at least in the classical setting some of the domains are not a good indicator for heuristic performance [13]. This might also be the case for their state-dependent counter part. Furthermore, without adaptation none of the presented heuristics is invariant under any of the presented transformations except exponential compilation. This motivates further research into whether it is worth to make these heuristics invariant.

4 Conclusion

In this paper, we have empirically investigated the benefits of planning when state-dependent action costs are supported or ignored. Our first experiment showed that supporting them during search is usually beneficial in that it leads to better plans. Depending on the domain, ignoring costs can positively or negatively affect search guidance. Our second experiment investigated how beneficial it is to reflect state-dependent cost within goal-distance heuristics. Again, results are mixed. The more accurately costs are represented within the heuristic, the more informative the heuristic values can become, provided that the simplification that underlies the heuristic computation is compatible with the way costs are represented in the heuristic (which is not necessarily the case—i. e., auxiliary predicates and actions introduced to represent cost functions may also "confuse" the heuristic). More informative heuristic values may or may not, in turn, lead to fewer node expansions, as discussed by, e. g., Fan et al. [4], and this may or may not translate into lower runtimes and more solved problems. Even if the heuristic is accurate, but expensive, heuristic values may be informative and node expansions low, but runtime can still be prohibitively large.

Acknowledgments. David Speck was supported by the German National Science Foundation (DFG) as part of the project EPSDAC (MA 7790/1-1). Florian Geißer was supported by ARC project DP180103446, "On-line planning for constrained autonomous agents in an uncertain world".

References

1. Bonet, B., Geffner, H.: Planning as heuristic search: New results. In: Proceedings of the 5th European Conference on Planning (ECP). pp. 360–372 (1999)
2. Ciardo, G., Siminiceanu, R.: Using edge-valued decision diagrams for symbolic generation of shortest paths. In: Proceedings of the Fourth International Conference on Formal Methods in Computer-Aided Design (FMCAD). pp. 256–273 (2002)
3. Fan, G., Müller, M., Holte, R.: Additive merge-and-shrink heuristics for diverse action costs. In: Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI). pp. 4287–4293 (2017)
4. Fan, G., Müller, M., Holte, R.: The two-edged nature of diverse action costs. In: Proceedings of the Twenty-Seventh International Conference on Automated Planning and Scheduling (ICAPS). pp. 98–106 (2017)
5. Geffner, H., Bonet, B.: A Concise Introduction to Models and Methods for Automated Planning. Morgan & Claypool Publishers (2013)
6. Geißer, F.: On Planning with State-dependent Action Costs. Ph.D. dissertation, Albert-Ludwigs-Universität Freiburg (2018)

7. Geißer, F., Keller, T., Mattmüller, R.: Delete relaxations for planning with state-dependent action costs. In: Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI). pp. 1573–1579 (2015)
8. Geißer, F., Keller, T., Mattmüller, R.: Abstractions for planning with state-dependent action costs. In: Proceedings of the 26th International Conference on Automated Planning and Scheduling (ICAPS). pp. 140–148 (2016)
9. Haslum, P., Botea, A., Helmert, M., Bonet, B., Koenig, S.: Domain-independent construction of pattern database heuristics for cost-optimal planning. In: Proceedings of the 22nd AAAI Conference on Artificial Intelligence. pp. 1007–1012 (2007)
10. Helmert, M.: The Fast Downward planning system. Journal of Artificial Intelligence Research **26**, 191–246 (2006)
11. Helmert, M., Domshlak, C.: Landmarks, critical paths and abstractions: What's the difference anyway? In: Proceedings of the 19th International Conference on Automated Planning and Scheduling (ICAPS) (2009)
12. Helmert, M., Haslum, P., Hoffmann, J.: Flexible abstraction heuristics for optimal sequential planning. In: Proceedings of the 17th International Conference on Automated Planning and Scheduling (ICAPS). pp. 176–183 (2007)
13. Helmert, M., Röger, G.: How good is almost perfect? In: Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence (AAAI 2008). pp. 944–949 (2008)
14. Ivankovic, F., Haslum, P., Gordon, D.: Planning with global state constraints and state-dependent action costs. In: Proceedings of the 29th International Conference on Automated Planning and Scheduling (ICAPS) (2019)
15. Keller, T., Pommerening, F., Seipp, J., Geißer, F., Mattmüller, R.: State-dependent cost partitionings for Cartesian abstractions in classical planning. In: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI). pp. 3161–3169 (2016)
16. Pearl, J.: Heuristics: Intelligent Search Strategies for Computer Problem Solving. Addison-Wesley Longman (1984)
17. Speck, D., Geißer, F., Mattmüller, R.: Symbolic planning with edge-valued multi-valued decision diagrams. In: Proceedings of the Twenty-Eighth International Conference on Automated Planning and Scheduling (ICAPS). pp. 250–258 (2018)
18. Speck, D., Geißer, F., Mattmüller, R.: SYMPLE: Symbolic planning based on EVMDDs. In: Ninth International Planning Competition (IPC-9): planner abstracts. pp. 91–94 (2018)