Learning Bench Transition Systems

Previous paper [Heusner et al., 2017] shows that the search space of greedy best-first search (GBFS) with a given heuristic h induces a *bench transition system* (BTS) where *benches* are connected via *progress stats* (PS) and *bench entry states*. In those papers, the bench transition system is not learned but deduced after hand, when the solution (the plan) already has been found. Our main goal is to use those benches and break down the search into intermediate sub-searches to finally achieve sub-plans such that if combined sequentially produce one plan. In [Drexler et al., 2022] there is a similar approach however using a different technique that is limited to some specific domains.

In this study, I intend to:

1. learn a generalized representation of the bench transition system for a given domain and heuristic based on data from small instances

2. exploit the learned bench transition system by performing a sequence of searches from a bench entry state to a progress state

The main idea is a decomposition of planning problems into sub-tasks. The technique is by learning the BTS of a domain from small examples (small instances) and then using the learned BTS on large instances of the same domain.

A high-level overview of our approach can be sketched as follows: Start with generating all benches for each task and then generate description logic features. For each bench, learn a formula that describes its exit states, and merge bench A into bench B if the goal formula of B is also perfect for A. Lastly, for each bench, learn a formula that describes a state in the bench. To use this framework perform a bench walking: perform an intermediate search between the learned benches.

This line of research is an extension of our successful previous work presented at IJCAI 22 [Ferber et al., 2022] in which the main goal was to introduce a novel approach that learns a description logic formula characterizing all progress states in a classical planning domain. Using the learned formulas in a GBFS to break ties in favor of progress states often significantly reduces the search effort. Our previous work showed that learning progress states is feasible and efficient. The next step is taking the power of learning PS to learn the complete BTS and improve the search itself and not only resolve cases of tie-breaking. The immediate implications of the results attained and the knowledge gained is the ability to decompose a planning problem into sub-problems and solved it sequentially, other work that is based on Sketches showed that the decomposition technique is efficient and revolutionary. More secondary implications are related to implementation, we aim to represent PDDL goals as description logic and vice versa, this itself is a contribution. The fields that are likely to benefit from our results are automated planning. As mentioned before, previous work introduced the concept of *progress states*, *BTS*, etc. Before our work, other methods could identify progress states only for a single task, and only *after* a solution for the task has been found. We passed this step and are now able to learn progress stats, learning the entire BTS is a huge step in decomposing planning tasks.

References

[Drexler et al., 2022] Drexler, D., Seipp, J., and Geffner, H. (2022). Learning sketches for decomposing planning problems into subproblems of bounded width. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 32, pages 62–70.

[Ferber et al., 2022] Ferber, P., Cohen, L., Seipp, J., and Keller, T. (2022). Learning and exploiting progress states in greedy best-first search. In *Proceedings of the 31th International Joint Conference on Artificial Intelligence (IJCAI 2022)*, pages 4740–4746.

[Heusner et al., 2017] Heusner, M., Keller, T., and Helmert, M. (2017). Understanding the search behaviour of greedy best-first search. In *Proceedings of the International Symposium on Combinatorial Search*, volume 8, pages 47–55.