Proceedings of the Seventh Annual Symposium on Combinatorial Search (SoCS 2014)

Improved Heuristic Search for Sparse Motion Planning Data Structures

Andrew Dobson, and Kostas E. Bekris
Department of Computer Science, Rutgers University
New Brunswick, New Jersey 08901

Abstract

Sampling-based methods provide efficient, flexible solutions for motion planning, even for complex, high-dimensional systems. Asymptotically optimal planners ensure convergence to the optimal solution, but produce dense structures. This work shows how to extend sparse methods achieving asymptotic near-optimality using multiple-goal heuristic search during graph constuction. The resulting method produces identical output to the existing Incremental Roadmap Spanner approach but in an order of magnitude less time.

Introduction

Heuristic search can be directly applied to address motion planning (Cohen, Chitta, and Likhachev 2013). Alternatively, it is possible to precompute a roadmap of collision-free robot configurations before calling a search method (Kavraki et al.1996). This approach provides efficient solutions even for relatively complex challenges. Recent work formulated when such roadmaps asymptotically converge to optimal solutions, which relates to the graph's density (Karaman and Frazzoli 2011, Karaman and Frazzoli 2010).

By relaxing optimality requirements and reducing the roadmap's density, asymptotic near-optimality can be achieved (Marble and Bekris 2011, Salzman and Halperin 2013, Fu and Balkcom 2013, Marble and Bekris 2013). Such methods, such as the Incremental Roadmap Spanner (IRS) (Marble and Bekris 2013), provide faster query resolution, reduced storage and communication requirements. They employ informed search to detect which edges are not needed in a sparse near-optimal roadmap. Nevertheless, this method exhibits large computational costs when tens of thousands of nodes are required to solve a problem, and the search dominates run time. There are methods that also remove nodes from the graph (Dobson and Bekris 2014) but they further relax optimality guarantees

The objective is to use advancements in discrete search to improve the efficiency of IRS, while providing the same sparsity level (Davidov and Markovitch 2006). It is known that when multiple searches are performed over similar graphs, lifelong planning approaches improve search efficiency dramatically (Koenig, Likhachev, and Furcy 2004).

Problem Setup

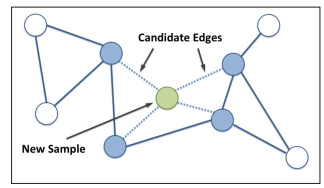

Figure 1: The algorithm finds a feasible set of candidate edges to include to connect the new node to the graph.

Sampling-based methods examine local neighborhoods for each added configuration, and must decide which neighbors to create an edge to. The asymptotically optimal PRM* attempts to connect the n-th sample to $logn$ neighbors. This work examines the Incremental Roadmap Spanner (IRS) technique (Marble and Bekris 2013), a planner that provides asymptotic near-optimality and a relatively sparse graph. IRS processes candidate edges to a configuration's $logn$ neighbors and determines if omitting the edge violates a spanner property. A graph spanner is a subgraph which maintains that the path between any pair of nodes has length no longer than t-times longer than in the original graph, where t is called the stretch of the spanner.

To maintain this invariant, IRS performs A* search for each candidate edge, and compares the returned path length to the weight of the edge, only adding it if the spanner property is violated. IRS solves the problem illustrated in Figure 1, that is, what is a small set of edges to add while maintaining a graph spanner. This work proposes a heuristic search process to reduce the computational cost of finding such edges. Multiple goal A* search is used to identify sets of edges which can be safely omitted. Instead of performing search for each neighbor, the faster IRS alternative searches only when an edge is added, and consecutive calls reuse prior information.

Methodology

The fast IRS approach iteratively adds a single candidate edge to the graph. Then, it performs multiple-goal A* search to determine which neighbors can be reached with a g-value that does not violate the spanner property, removing the edges to those nodes from the candidate edge set. It continues doing this until there are no more candidate edges. Like the original IRS, the method processes all candidate edges in order of distance from the start. The improved method will accept exactly the same set of edges as the original IRS, but determines this set at a much faster rate. Asymptotic analysis of this new method is omitted here, as the running time depends on the number of edges retained. Characterizing how many edges the IRS method retains is an open problem but does not correspond to the optimal.

Algorithm 1: multi-A$^*(G, start, N, t)$

1 $\forall u : Color[u] == BLACK,\ Color[u] := GRAY;$
2 $O \leftarrow O \cup \{start\};\ reached \leftarrow \emptyset;$
3 **while** $O \neq \emptyset$ *and* $N \neq \emptyset$ **do**
4     $u \leftarrow remove\_min(O);$
5   **if** $u \in N$ **then**
6   **if** $g(u) < t \cdot |start, u|$ **then**
7             $reached \leftarrow reached \cup \{u\};$
8         $N \leftarrow N \setminus \{u\};$
9   Update $h(\cdot)$ in O;
10     $O \leftarrow \texttt{expand}(u);$
11     $Color[u] := BLACK;$
12 **return** $reached$;

The multi-goal A* approach is given in Algorithm 1. Line 1 ensures that nodes expanded in the previous iteration are marked gray, so that the search is able to explore through these nodes. Then, the start state is put into the open set, and reached goals are reset (Line 2). Then, the A* search continues until either the open set is empty, or all of the goals have been reached (Line 3). Then, if the minimum element of the open set O is a goal (Line 5), it is checked for the spanner property (Line 6), and if it satisfies the property, the element is added to the reached set of goals (Line 7). Then, it is removed from the goal set (Line 8), and the heuristics for all elements in the open set are updated (Line 9). Whether the node was a goal or not, it is then expanded in the same fashion as a standard A* search (Line 10) and marked black, indicating its expansion (Line 11). The algorithm returns all goals which were reached while satisfying the spanner property (Line 12).

Experimental Validation

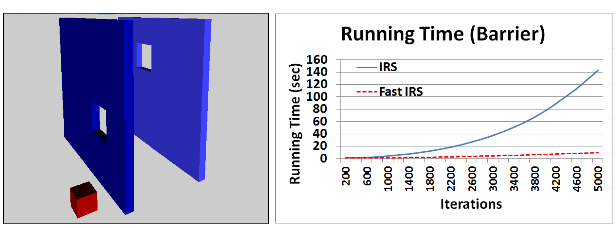

Figure 2: The simple Barriers benchmark for testing the improvements to IRS, showing much lower running time.

The benefits of the approach are validated against the original IRS implementation. First, as a baseline, the IRS methods are tested in a simple rigid body setup (Barriers), illustrated in Figure 2 (Left), which is a rather standard challenge in motion planning. The main performance measure tested is the running time for the algorithms. As shown in Figure 2 (Right), the optimized version of IRS runs an order of magnitude quicker than the original, greatly improving its practical applicability. To exemplify this, the method is also tested in a robot manipulation framework, to solve pick-and-place tasks on a shelf, shown in Figure 3 (Left). Note that the two approaches produce the same graph and return the same solutions, but with vastly different runtimes.

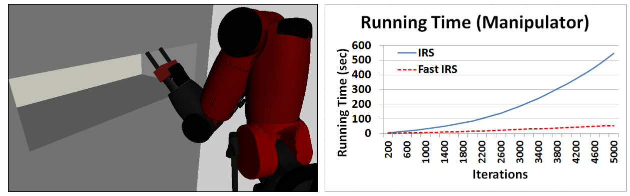

Figure 3: The manipulation benchmark, which also shows great improvement in running time.

Conclusion

Using a more intelligent search, computing sparse planning structures can be made much more efficient. This work provides a multi-target, incremental method to speed up IRS by an order of magnitude. Future work will cast the problem in an optimization framework, attempting to find the minimal set of edges to add to the graph.

References

Cohen, B.; Chitta, S.; and Likhachev, M. 2013. Heuristic Search-based Planning for Manipulation. *International Journal of Robotics Research (IJRR)*.

Davidov, D., and Markovitch, S. 2006. Multiple-Goal Heuristic Search. *Journal of Artificial Intelligence Research* 26:417–451.

Dobson, A., and Bekris, K. E. 2014. Sparse Roadmap Spanners for Asymptotically Near-Optimal Motion Planning. *IJRR* 33.

Fu, W., and Balkcom, D. 2013. A fast streaming spanner algorithm for incrementally constructing sparse roadmaps. In *IEEE/RSJ Intern. Conf. on Intelligent Robots and Systems (IROS)*.

Karaman, S., and Frazzoli, E. 2010. Incremental Sampling-based Algorithms for Optimal Motion Planning. In *RSS*.

Karaman, S., and Frazzoli, E. 2011. Sampling-based Algorithms for Optimal Motion Planning. *IJRR* 30(7):846–894.

Kavraki, L. E.; Svestka, P.; Latombe, J.-C.; and Overmars, M. 1996. Probabilistic Roadmaps for Path Planning in High-Dimensional Configuration Spaces. *IEEE TRA* 12(4):566–580.

Koenig, S.; Likhachev, M.; and Furcy, D. 2004. Lifelong Planning A*. *Artificial Intelligence Journal* 155(1-2):93–146.

Marble, J. D., and Bekris, K. E. 2011. Asymptotically Near-Optimal is Good Enough for Motion Planning. In *ISRR*.

Marble, J., and Bekris, K. E. 2013. Asymptotically Near-Optimal Planning with Probabilistic Roadmap Spanners. *IEEE Transactions on Robotics* 29:432–444.

Salzman, O., and Halperin, D. 2013. Asymptotically Near-Optimal RRT for Fast, High-quality Motion Planning. Technical report, arXiv.