Research Statement

MAXIM LIKHACHEV

My long-term research goal is to develop a methodology for robust real-time decision-making in autonomous systems. To achieve this goal, my students and I research novel decision-making algorithms and use these algorithms to build planning modules that enable complex robotic systems to operate autonomously. Our approach is currently based on pushing the limits of graph search-based planning. Conventional wisdom in the robotics community is that graph search approaches cannot provide real-time performance guarantees, do not scale to higher-dimensional problems, and cannot deal with problems that involve uncertainty. My group develops graph search algorithms that *are* capable of solving challenging problems in robotics in real-time while still maintaining all the positive properties of graph search algorithms such as generality, cost minimization and rigorous guarantees on completeness and quality of solutions. We then use these algorithms to build real-time planners and demonstrate them on physical robots performing such tasks as autonomous navigation, autonomous flight and landing, autonomous mobile manipulation and others.

Through our work on building actual planners for physical robots, we have found that it is important to link tightly research on graph search algorithms with the work on deriving the "right" representations of planning problems. The representation needs to encode the planning problem in a way that facilitates both efficient search for a solution and robust execution of the solution by a robot. For example, motion planning for a car-like robot cannot use graphs derived from simple grids. Planning on such graphs assumes the ability to turn in-place and renders the generated plans infeasible to execute for a car-like robot. Our experience has shown that careful thinking about the representation leads to a better understanding of what are the *real* challenges that need to be addressed in building an effective planner. This knowledge drives our research into developing novel graph search techniques that overcome these challenges. Also, finding the "right" graph representation can often be combined with the problem of searching the graph. Studying the combined problem can lead to a highly effective solution to the overall planning problem.

All in all, I strive to maintain the research environment in my lab that is unique in that we both, do highly algorithmic work on developing novel graph search algorithms and work closely with physical robots and draw inspiration from this work. For example, I have co-developed well-known incremental and anytime graph search algorithms such as D* Lite [16], ARA* [22] and Anytime D* [21], my group pioneered the Experience Graph [27], a framework that enables heuristic search algorithms to improve their runtimes by solving similar problems, and we have just developed Multi-

(a) dual-arm   (b) full-body   (c) industrial   (d) assembly
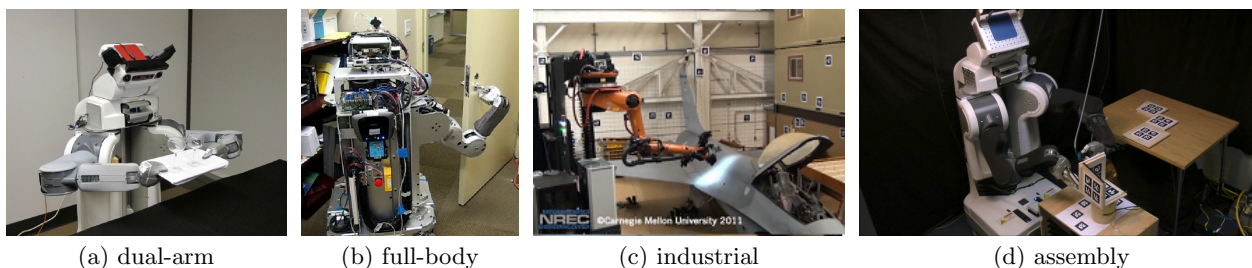
Figure 1: Our work on developing novel graph search algorithms and compact graph representations and applying them to high-dimensional planning for a wide range of mobile manipulation tasks.

Heuristic A* [3], the first heuristic search to handle multiple (possibly many) inadmissible heuristics without losing its theoretical guarantees. At the same time, my group built planners for such impressive systems as an industrial mobile manipulator used for paint stripping different aircrafts (Figure 1(c)) - the project that won the national Gold Edison award in 2013 - a full-size K-MAX helicopter (Figure 2(b)) and a full-size SUV (Figure 2(c)) that won the DARPA Urban Challenge race in 2007 [20]. By building such planners, we push forward the frontier of robotics and change how many of the planning problems in robotics are approached.

In the following, I give a few examples from my research. I first briefly mention several research themes that cut through most of the work my group and I have done in the past. Afterwards, I give several more examples that describe some of the latest research directions we have been pursuing.

- **Graph search-based planning for solutions with bounded sub-optimality.** While finding a provably optimal path in a high-dimensional search-space is computationally intractable, for many planning problems in robotics, allowing even a small amount of sub-optimality in the solution allows the search to quickly find high-quality solutions. We have exploited this observation to develop a number of graph search algorithms that allow the trade-off of solution quality for fast planning time including an anytime version of A*, ARA* [22], Anytime SIPP [26, 29] for planning in dynamic environments and Planning with Adaptive Dimensionality [11, 13]. Together with my students and colleagues, we have used these searches to build highly effective planners for high-dimensional robotic systems ranging from single-arm and dual-arm manipulation [8] to full-body manipulation on PR2 [6] (Figure 1(a,b)) and on a large mobile manipulator built to strip paint off airplanes autonomously (Figure 1(c)).

- **Decomposition of hard planning problems into a series of easy-to-solve graph searches.** Another observation my research exploits is that many seemingly difficult planning problems in robotics can often be decomposed into a series of easy-to-solve graph searches. The solutions found by these searches can be combined to obtain solutions to the original problems with rigorous theoretical guarantees. Based on this observation, we have developed a number of algorithms including R* [23] for high-dimensional planning, Probabilistic Planning with Clear Preferences (PPCP) [24] for planning under uncertainty in the environment and Distributed Path Consensus (DPC) [5] for multi-robot planning. We have then used these algorithms to build planners that can run on-board robotic systems. For example, PPCP was run on-board a small autonomous quadrotor built by my students to compute an optimal landing site selection policy under uncertainty in landing sites [19] (Figure 2(a)).

- **Incremental graph search algorithms.** Many problems in robotics require constant re-planning in response to the discovery of the environment, corrections in the localization of the robot, imperfect actuation and changes in the environment. Jointly with my students and collaborators, we have developed and continue to develop new incremental graph search

| (a) | (b) | (c) | (d) |

Figure 2: Our work on developing novel graph search algorithms and applying them to planning for autonomous flight and landing, autonomous navigation and control of small teams of robots.

algorithms that speed up repeated planning in such domains by re-using search efforts. Some of these algorithms include D* Lite [18, 16], Real-time Adaptive A* [17], Truncated Incremental Search [1], Anytime Tree-Restoring Weighted A* [14] and Anytime D* [21]. To the best of my knowledge, the algorithm Anytime D* we developed was the first heuristic search to be both anytime and incremental. We have used it to build motion planning for a variety of ground and aerial vehicles including a fully autonomous micro-aerial vehicle (Figure 2(a)) [25], a full-size K-MAX helicopter performing autonomous flight and landing (Figure 2(b)) and a full-size SUV (Figure 2(c)) that won the DARPA Urban Challenge race in 2007 [20].

- **Graph search-based planning for mobile manipulation.** Much of my inspiration for developing heuristic search algorithms for high-dimensional planning comes from the field of autonomous mobile manipulation. In planning for robotic manipulation, heuristic search-based planning was commonly thought of as impractical due to the high-dimensionality of the planning problem. In the last five years, my group has been developing novel graph search algorithms and compact graph representations that, by exploiting some of the properties of mobile manipulation tasks, *do* achieve real-time performance without sacrificing rigorous guarantees that heuristic search algorithms usually provide [8, 11]. Unlike most other solutions to motion planning for high-DOF mechanisms, these approaches provide deterministic guarantees on completeness and bounds on the sub-optimality of the generated solution with respect to the graph that models the problem. As a result, they typically generate motions that are consistent from run to run, are close to what users anticipate from the robot and minimize cost function well. These approaches have been used for single-arm [10], dual-arm [7], N-arm [9] and full-body manipulation tasks [12] (Figure 1) and run on both academic as well as multiple industrial robotic systems built at CMU (Figure 1(c)) and elsewhere.

Recently, my group began to explore several new directions of research. These directions were motivated by several key observations we made while building search-based planners for physical robots and getting them to run effectively in real-world scenarios. I believe that one of the key benefits we get from transitioning our algorithms onto real robots is making such observations about the characteristics of robotic systems and the tasks they are required to accomplish. These observations enable us to build new classes of algorithms that become capable of solving problems that were previously unsolvable with heuristic search-based planning algorithms. Below I outline several of our recent findings.

- **Graph search algorithms that learn to improve their performance based on experience and demonstrations.** Robots are often used to perform similar tasks over and over again. It is therefore important for us to study how planning algorithms can improve their speed and robustness based on past planning experiences as well as demonstrations provided by humans and/or other robots. This approach is in contrast to incremental graph search algorithms that speed up re-planning within a *single* execution of a task. To this end, my group has started to research a new class of heuristic searches that are capable of improving their performance based on their previous experiences and demonstrated solutions [27, 28, 2]. For example, we have recently developed a new approach to graph search-based planning that we call *Experience Graphs* [27]. Planning with Experience Graphs builds a faster-to-solve graph representation of the planning problem based on the solutions it has found previously or demonstrations provided by a person and utilizes this representation to focus the search for a solution in a way that preserves rigorous guarantees on completeness and bounded sub-optimality with respect to the original graph representation of the problem. Planning with Experience Graphs

turned out to be highly beneficial in the variety of complex mobile manipulation tasks ranging from assembly (Figure 1(d)) to paint stripping (Figure 1(c)). To my knowledge, Experience Graphs is the first heuristic search method that "learns from its experience" a more compact graph representation that speeds up its future planning times and does it in a way that preserves rigorous guarantees on the solution quality.

- **Graph search with multiple heuristics.** One of the most recent family of graph search algorithms that my group has developed is Multi-Heuristic A* [4, 30, 15, 3]. These algorithms build on the observation that while in many robotics planning problems it is common to have multiple heuristic functions (i.e., estimates on cost-to-goal) available for guiding the search, it can often be highly ineffective to combine these functions into a single heuristic function that can be utilized by a heuristic search. Furthermore, it is hard to ensure that all of these heuristic functions are admissible and consistent, the properties that are typically required to provide guarantees on completeness and solution quality. Motivated by these observations, we have developed a novel heuristic search, called Multi-Heuristic A* (MHA*), that takes in multiple, arbitrarily inadmissible heuristic functions in addition to a single consistent heuristic, and uses all of them simultaneously to search for a solution in a way that guarantees completeness and bounded sub-optimality. This methodology turned out to be highly effective for high-dimensional planning problems such as full-body mobile manipulation that often have several lower-dimensional subspaces that can be used to compute multiple heuristic functions, some of which may be inadmissible. This effectiveness combined with the simplicitly and rigorous theoretical properties of MHA* are typically what I strive to have the most in the algorithms my group and I develop.

In addition to publishing papers on our research, I am eager to see the impact of our results in the real world. To this end, my group has built and actively maintains an open-source library - Search-based Planning Library (SBPL) - that includes many of the graph search algorithms and search-based planning modules that we have developed. This library comes as part of ROS (Robotic Operating System). The SBPL library has been used by a number of universities across the world, various companies and numerous DoD service labs as either a stand-alone library or as part of ROS for such tasks as autonomous navigation, autonomous flight and mobile manipulation. In addition, my group actively participates in transitioning our technology onto fielded systems. Some of the recent examples include developing a motion planner for a full-scale K-MAX helicopter flying at a speed of up to 100 knots (115 mph) and avoiding no-fly zones detected in-flight (Figure 2(b)) and developing manipulation and navigation planners for an industrial mobile manipulator used for paint stripping different aircrafts (Figure 1(c)), the project that won the national Gold Edison award in 2013.

To summarize, I love developing algorithms that are simple, provide strong theoretical guarantees and are effective in solving real-world problems in robotics. In all of my work, I am driven by challenging decision-making problems in robotics. I believe the current state of autonomous robotics is far from mature and the lack of adequate decision-making methods contributes to this. This motivates the work of my group on developing effective decision-making algorithms and showing them in action on physical robots.

References

[1] S. Aine and M. Likhachev. Truncated incremental search: Faster replanning by exploiting suboptimality. In *Proceedings of the Eighteenth National Conference on Artificial Intelligence (AAAI)*, 2013.

[2] S. Aine, C. Sharma, and M. Likhachev. Learning to search more efficiently from experience: A multi-heuristic approach. In *Proceedings of the International Symposium on Combinatorial Search (SoCS)*, 2015.

[3] S. Aine, S. Swaminathan, V. Narayanan, V. Hwang, and M. Likhachev. Multi-heuristic A*. *International Journal of Robotics Research (IJRR)*. Accepted for publication.

[4] S. Aine, S. Swaminathan, V. Narayanan, V. Hwang, and M. Likhachev. Multi-heuristic A*. In *Proceedings of the Robotics: Science and Systems Conference (RSS)*, 2014.

[5] S. Bhattacharya, V. Kumar, and M. Likhachev. Distributed optimization with pairwise constraints and its application to multi-robot path planning. In *Proceedings of the Robotics: Science and Systems Conference (RSS)*, 2010.

[6] S. Chitta, B. Cohen, and M. Likhachev. Planning for autonomous door opening with a mobile manipulator. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, 2010.

[7] B. Cohen, S. Chitta, and M. Likhachev. Search-based planning for dual-arm manipulation with upright orientation constraints. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, 2012.

[8] B. Cohen, S. Chitta, and M. Likhachev. Heuristic search-based planning for manipulation. *International Journal of Robotics Research (IJRR)*, 2013.

[9] B. Cohen, M. Phillips, and M. Likhachev. Planning single-arm manipulations with n-arm robots. In *Proceedings of Robotics: Science and Systems (RSS)*, 2014.

[10] B. Cohen, G. Subramanian, S. Chitta, and M. Likhachev. Planning for manipulation with adaptive motion primitives. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, 2011.

[11] K. Gochev, B. Cohen, J. Butzke, A. Safonova, and M. Likhachev. Path planning with adaptive dimensionality. In *Proceedings of the International Symposium on Combinatorial Search (SoCS)*, 2011.

[12] K. Gochev, A. Safonova, and M. Likhachev. Planning with adaptive dimensionality for mobile manipulation. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, 2012.

[13] K. Gochev, A. Safonova, and M. Likhachev. Incremental planning with adaptive dimensionality. In *Proceedings of the International Conference on Automated Planning and Scheduling (ICAPS)*, 2013.

[14] K. Gochev, A. Safonova, and M. Likhachev. Anytime tree-restoring weighted A* graph search. In *Proceedings of the International Symposium on Combinatorial Search (SoCS)*, 2014.

[15] F. Islam, V. Narayanan, and M. Likhachev. Dynamic multi-heuristic A*. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2015.

[16] S. Koenig and M. Likhachev. D* Lite. In *Proceedings of the Eighteenth National Conference on Artificial Intelligence (AAAI)*, 2002.

[17] S. Koenig and M. Likhachev. Real-time adaptive A*. In *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2006.

[18] S. Koenig, M. Likhachev, and D. Furcy. Lifelong planning A*. *Artificial Intelligence Journal (AIJ)*, 2004.

[19] A. Kushleyev, B. MacAllister, and M. Likhachev. Planning for landing site selection in the aerial supply delivery. In *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, 2011.

[20] M. Likhachev and D. Ferguson. Planning long dynamically-feasible maneuvers for autonomous vehicles. *International Journal of Robotics Research (IJRR)*, 2009.

[21] M. Likhachev, D. Ferguson, G. Gordon, A. Stentz, and S. Thrun. Anytime search in dynamic graphs. *Artificial Intelligence Journal (AIJ)*, 172(14), 2008.

[22] M. Likhachev, G. Gordon, and S. Thrun. ARA*: Anytime A* with provable bounds on sub-optimality. In *Advances in Neural Information Processing Systems (NIPS) 16*. Cambridge, MA: MIT Press, 2003.

[23] M. Likhachev and A. Stentz. R* search. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, 2008.

[24] M. Likhachev and A. Stentz. Probabilistic planning with clear preferences on missing information. *Artificial Intelligence Journal (AIJ)*, 173(5-6):696–721, 2009.

[25] B. MacAllister, A. Kushleyev, J. Butzke, and M. Likhachev. Path planning for non-circular micro aerial vehicles in constrained environments. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, 2013.

[26] V. Narayanan, M. Phillips, and M. Likhachev. Anytime safe interval path planning for dynamic environments. In *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, 2012.

[27] M. Phillips, B. Cohen, S. Chitta, and M. Likhachev. E-graphs: Bootstrapping planning with experience graphs. In *Proceedings of the Robotics: Science and Systems Conference (RSS)*, 2012.

[28] M. Phillips, V. Hwang, S. Chitta, and M. Likhachev. Learning to plan for constrained manipulation from demonstrations. *Autonomous Robots (AURO)*. Accepted for publication.

[29] M. Phillips and M. Likhachev. Planning in domains with cost function dependent actions. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, 2011.

[30] M. Phillips, V. Narayanan, S. Aine, and M. Likhachev. Efficient search with an ensemble of heuristics. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2015.