Statement of Research Interests

Troy McMahon

April 2018

Kinodynamic Planning for Terrain Environments:

Kinodynamic planning is a branch of motion planning where differential constraints are applied to the motion of the robot. Constraints could include constraints on the direction of motion as well as upper an lower limits on velocity and acceleration. Kinodynamic planning has application in mobile robotics[2], autonomious driving[3], aerial vehicles [4], grasping, manipulation, space exploration, search and rescue and agricultural robots.

Kinodynamic planning is characterized by the lack of a solution to the 2-point boundary problem and consequently the lack of a steering function. This makes many common motion planning algorithms, such as the PRM [5] and RRT* [6] unsuitable for kinodynamic planning. We must therefore rely on algorithms such as the RRT[7] which only expand through forward propagation. Recent work has focused on the development of asymptotically optimal planners for systems without steering functions, which has lead to the development of methods such as Stable Sparse RRT [8], AO-X [9] and AO-RRT2 [10], and DIRT [11].

The lack of a steering function also makes it infesiable to make use of heurestics such as cost-maps during node expansion which means that most existing methods can only expand nodes by propagating random controls.

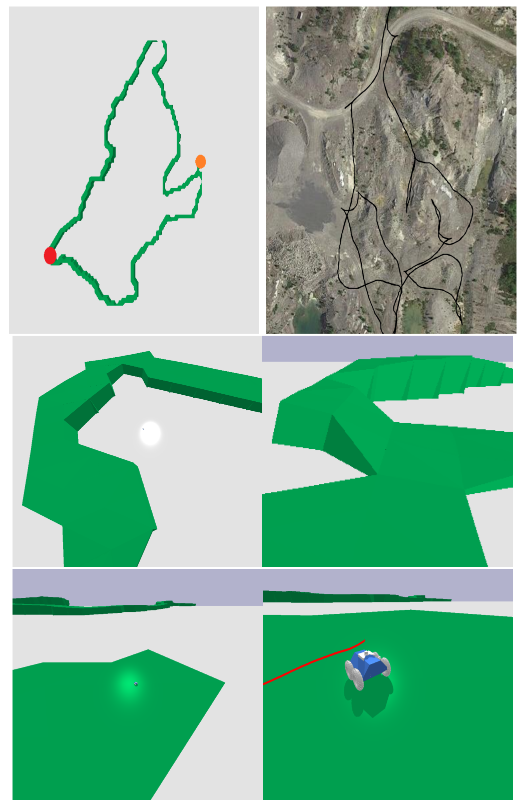


Figure 1: (top-left) Experimental environment constructed using the *Gravel Pit Lidar-Intensity Imagery Dataset*[1], which is generated from LIDAR scans of a 1.1 km of a circuit. (top left) Satellite image of the terrain based on which the dataset is generated [1]. (middle and bottom) Zoomed in images taken at various points along the path.

Physics and Terrain Environments Planning in real-world physical environments is a difficult open problem, due primarily to all of the factors which can influence motion. In addition to dynamics, planning must incorporate factors such as friction, gravity and the complex interactions between physical components of the robot. Terrain environments may also include regions that are impassable and areas with rough or steep terrain that is difficult to traverse.

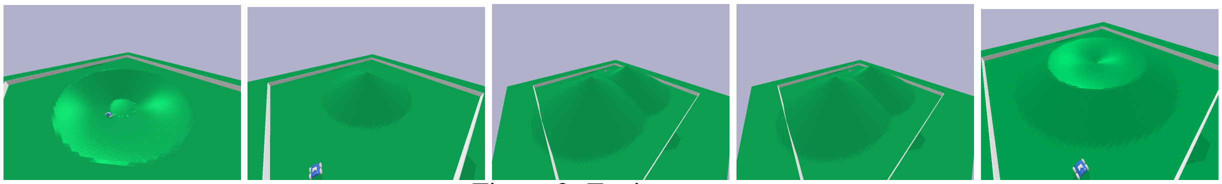


Figure 2: Environments.

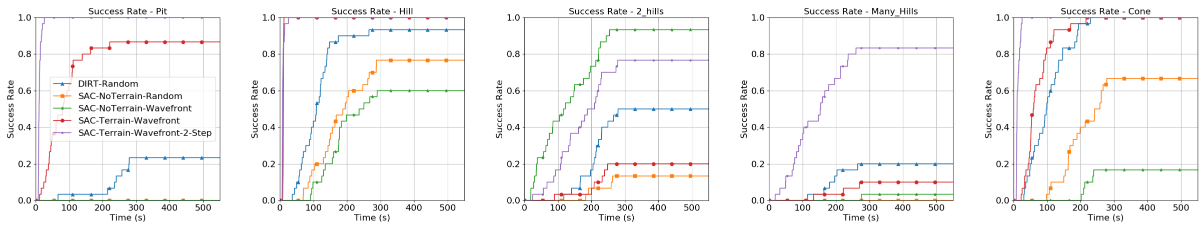


Figure 3: Success rate for system-order system.

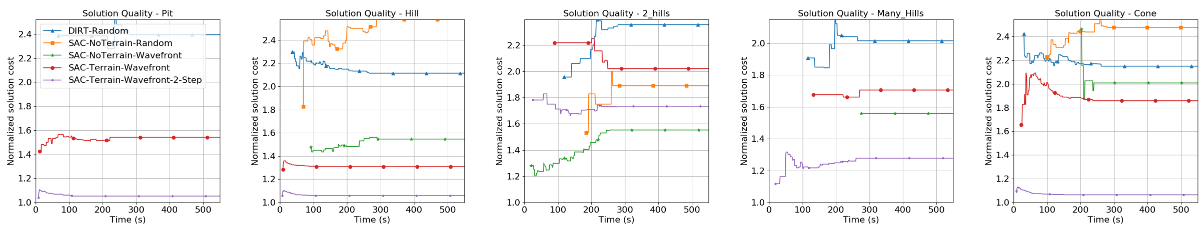


Figure 4: Solution quality for second-order system.

They may even include areas such as ravines or pits where the robot can become trapped. Furthermore, terrain environments may be extremely large (the Gravel Pit environment shown in Figure 1 consists of a 1.1 km circuit). Real world environments may also be unmapped, or not mapped in detail, which means that a motion planner needs the ability to plan and re-plan based on the robot's local observations.

Phyiscs simulators such as Bullet [12] and MuJoCo[13] have been developed to model physical environments and can be used to predict forward propagation during roadmap construction, however these simulators do not provide a steering function or a solution to the two point boundary problem.

Application of Learning Kinodynamic Planning in Physical Terrains My current research focuses on using learning to intelligently select controls to apply during forward propagation. In particular, I develop a learned model which can be used as a surrogate for a steering function in order to obtain controls which approximate reach a specified goal. This controller is parameterized on the robot's state and the topology of the terrain at the robot's position. I train this controller in a set of sloped environments using the Soft Actor-Critic (SAC) [14] architecture with Hindsight Experience Replay (HER) [15]. A variation of this learned model can also be used as a cost-to-go.

One major advantage of this controller is that it only requires local knowledge of the environment in the form of a tangent vector at the robot's current position. In real-world robots this information can be obtained from on-board sensors such as an inclinometer or from performing PCA on pointcloud representation of the local terrain. This locality makes the terrain-aware controller well suited to real-world applications where the robot may only know the local topology of the environment due to limited sensor range. This representation is also advantageous because it reduces the terrain to three parameters which limits the complexity of the controller.

I incorporate this model into the DIRT motion planning framework by using a wavefront function to select local goals using the learned controller to do forward propagation. The wavefront can either be generated from a cost-map of the environment or directly from the environment using the learned cost-to-go distance metric.

I apply this planner to a second order systems consisting of a RMP 440 LE robot in a variety of terrain environments. These experiments (Figures 1–4) demonstrate that the terrain aware controller significantly improves planner performance, with variations of the terrain-aware controller (SAC-Terrain-Wavefront and SAC-Terrain-2-Step-Wavefront) solving four of the five environments more effeciently than other methods. SAC-Terrain-2-Step-Wavefront also finds better quality (lower cost) paths in four environments. These results also show that the proposed wavefront function improves performance with methods using the wavefront giving the highest success rate and lowest cost path in all five environments.

Motion Planning for High Degree of Freedom Problems and Constrained Systems

Motion planning with constraints has applications in parallel robotics [16], grasping and manipulation [17], computational biology and molecular simulations [18], and animation [19]. Constrained motion planning problems place constraints on the motion of an object (robot). These constraints might require that the robot remain in contact with a surface or that it maintain a specific clearance. They could also require that certain joints of the robot remain in contact with each other (e.g., closed chains). Such constraints could be used to constrain the graspers of a manipulator to a set of grasping positions, or handles. They-https://www.sharelatex.com/project/5ae2d9ec8d13d5043f1f878d could also be used in industrial automation to constrain a tool mounted on a robot to a surface or a seam (for example, we could constrain a welder mounted on a robot to a seam which needs to be welded). Constraints could also be used to simulate contacts or bindings in protein folding simulations.

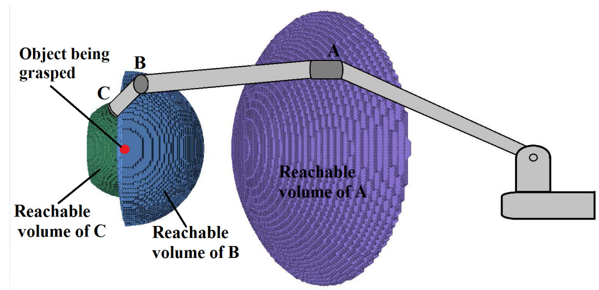


Figure 5: **The reachable volume of a robotic arm grasping a spherical object. An example configuration is shown in gray.**

Sampling-based motion planning methods such as the graph-based PRM [5] and the tree-based RRT [7] are state of the art solutions to traditional motion planning problems. Unfortunately, these methods are poorly suited for many constrained problems where the constraints form a manifold in C-space and planning must be restricted to this manifold [20]. Previous methods have developed specialized samplers that generate samples that satisfy constraints [21, 22, 23] that can be used in combination with existing PRM-based methods to solve problems with constraints. However, these methods are either unable to handle high degree of freedom (dof) systems or unsuited for systems with spherical or prismatic joints or systems that combine different types of joints.

Motion Planning with Reachable Volumes: The primary focus of my thesis work is motion planning for high degree of freedom robot and motion planning for highly constrained systems. I introduce the concept of *Reachable Volumes* [24], which are a geometric representation of the regions the joints and end effectors of a robot can reach, and use it to define a new planning space called RV-space where all points automatically satisfy a problems constraints. Samples and paths generated in RV-space naturally conform to constraints, making planning for constrained systems no more difficult than planning for unconstrained systems. Consequently, constrained motion planning problems that were previously difficult or unsolvable become manageable and in many cases trivial.

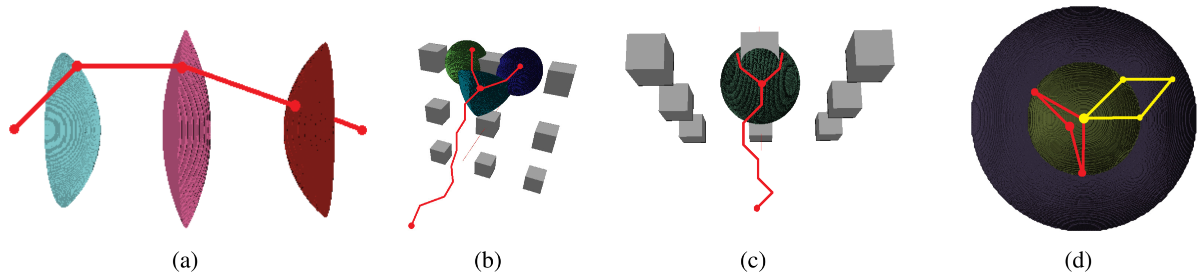


Figure 6: **The reachable volumes (a) of a 4 link chain with spherical joints with its end effector constrained to a point. The reachable volumes of a 16 dof fixed-base grasper with spherical joints is affected by constraints placed either (b) on the end effectors to be within spherical regions of the object or (c) on the base to be at a point. Note that in (c) the end effector reachable volumes are identical so only one is shown. The reachable volume (d) of a closed chain with 4 spherical joints and 4 links of equal length. The first and third joints can reach any point along the inner sphere (green) while the second joint can reach any point inside the outermost sphere (blue). Example configurations shown in red and yellow (d).**

The reachable volume of a joint/end effector is the volume of RV-space it can reach while satisfying a problem's constraints. Reachable volumes generalize the concept of reachable distances [25] so that it can be applied to linkages, closed chains and tree-like robots with prismatic and spherical as well as planar joints. Furthermore, reachable volumes allow constraints to be placed on any combination of joints and end effectors, and for multiple constraints to be applied at the same time. Most previous work focuses solely on end effector constraints.

Visualizations of reachable volumes have applications in robot control and operation where they can assist an operator by showing them what regions their robot can reach from a given location. This will help them to decide where and how to position the robot to accomplish their goal. For example, they could show the operator of a grasper where they need to position their robot in order to grasp and manipulate an object. They can also assist in robot design by indicating whether the robot can reach parts of the environment that it needs to in order to accomplish it's task. Figures 4 and 6 show examples of reachable volumes for a variety of robots.

My work also introduces tools and techniques to extend the state of the art sampling based motion planning algorithms to RV-space. We proposed a reachable volume sampler, a reachable volume local planner and a reachable volume distance metric. Reachable volume sampling generates samples by iteratively sampling the joints of a robot in their reachable volumes, resulting in samples which are guaranteed to satisfy a problems constraints. It can solve problems with constraints applied to any combination of joints/end effectors, while most other methods (e.g. [26, 23, 27]) assume a single constraint, usually on one of the end effectors. The reachable volume local planner and distance metric can be be used to generate constraint satisfying local paths, even in problems such as closed chains where the constraints form a manifold. As part of the reachable volume local planner we present a novel method for stepping reachable volume samples to generate samples that are close to the original while ensuring they satisfy the problem's constraints.

Further adaptations of reachable volumes [28] allow them to be used to construction RRTs. The resulting Reachable Volume RRT (RVRRT) constructs an RRT in RV-space, resulting in paths that are guaranteed to satisfy the constraints. RVRRTs are capable of solving many high degree of freedom and highly constrained problems that RRTs have previously been unable to solve.

The geometric complexity of reachable volumes is $\mathcal{O}(1)$ in unconstrained problems as well as for many constrained problems. This allows us to generate samples in linear time with respect to the number of bodies in the robot, which is the best possible complexity for a sampler. In problems with more complex constraints, samples can be generated in $\mathcal{O}(|L|^2\ |S|\ \text{Complexity}(S))$ time (where S is the set of constraints and $|L|$ is the number of bodies in the robot). The reachable volumes of all of the joints/end effectors in a robot can be computed in $\mathcal{O}(|J|{*}\text{diameter}(R))$ time, where $|J|$ is the number of joints in the robot and diameter(R) is the diameter of the robot. This is superior to $\mathcal{O}(|J|^2)$ time that would be required to compute these reachable volumes separately. Furthermore, roadmaps generated using reachable volume sampling are probabilistically complete.

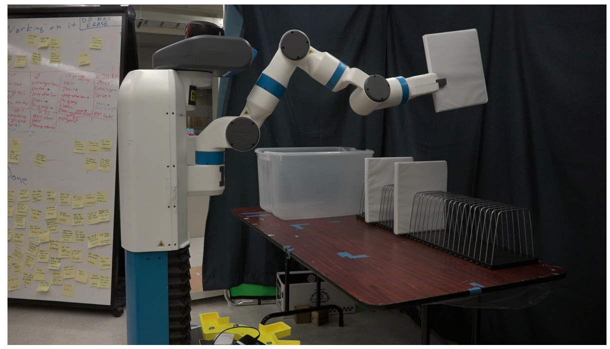


Figure 7: **Fetch robot manipulating notebooks**

In [29], we extended the concept of reachable volumes to accommodate rotational joints and directional constraints. To facilitate this we introduced the concept of directional reachable volumes which expands the concept of reachable volumes to include direction and orientation. We proposed a directed reachable volume planning space and presented methods for computing directed reachable volumes and using them to generate constraint satisfying samples. We showed that directional reachable volumes could be applied to real world grasping and manipulation problems such as those shown in Figure 7.

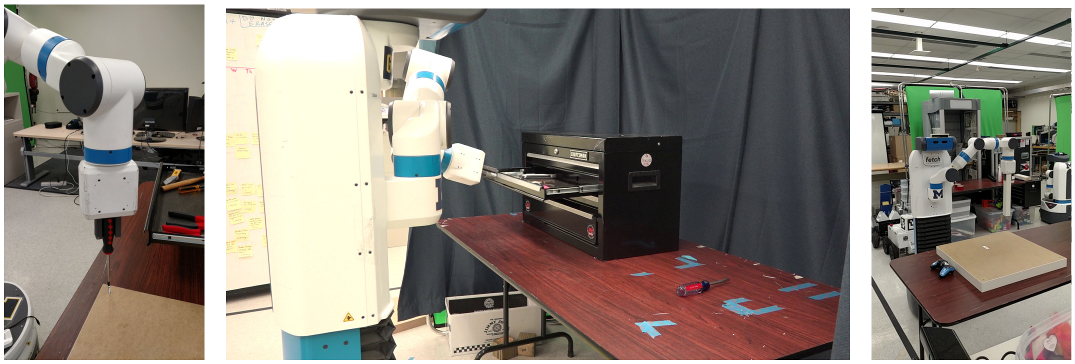
Motion Planning for Manipulation Affordances:

Affordances provide a natural means for a robot to reason about its environment and the universe of actions available to it. Planning for affordances has applications in areas such as eldercare, manufacturing, and interplanetary exploration.

In order to accommodate affordances, the robot must be able to produce motions that interact with the objects associated with the affordance. These motions must be robust to obstacles in the environment, and they must be able to accommodate any constraints associated with the affordance. To address this problem, we present the concept of affordance wayfields which represent affordances as a cost field over workspace or c-space. We then apply gradient descent planning in wayfields in order to generate motion that enact affordances. Affordance wayfields are advantageous in that they can adapt to obstacles and they can accommodate problem specific constraint. We showed that affordance wayfields were able to solve complex tasks such as those shown in Figure 8.

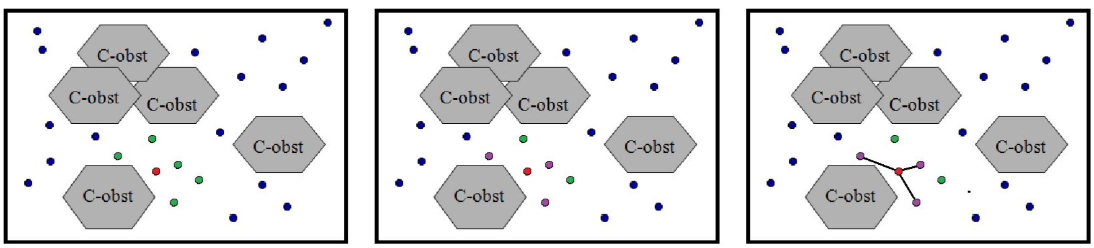
Addressing motion planning fundamentals:

In another project, I studied how different neighborhood selection methods effected PRM constructed. I first evaluated how neighborhood selection methods such as kd-trees and metric trees effected PRT construction time. I then studied how approximate methods such as Spill Trees and DPES effected roadmap quality. This work showed that using approximate neighborhood finders could reduce roadmap construction time significantly for roadmaps with a large number of nodes, and that roadmaps constructed using approximate methods are similar in quality to those constructed using exact methods.



(a) driving a screw (b) opening a drawer (c) screwing in a table leg

Figure 8: **Affordnaces**



(a) (b) (c)

Figure 9: **LocalRand Neighbor Selection: The LocalRand method first locates the k' closest nodes to a sample then selects k of those nodes at random. In this example, LocalRand selects the 5 (green) closest nodes to the sample (red). It then selects 3 (purple) of those nodes at random and attempts to connect them to the sample (green and red edges).**

I next studied how introducing randomness to neighbor selection effects roadmap quality [30]. The results of this work indicate that introducing some randomness into neighbor selection resulted in roadmaps that were better connected. As part of this work, I introduced a method called LocalRand (Figure 9) which finds a set of nodes that are close to a sample then selects a subset of these nodes at random. I showed experimentally that LocalRand is able to produce better connected roadmaps than the existing methods such as K-Closest, while maintaining a similar cost.

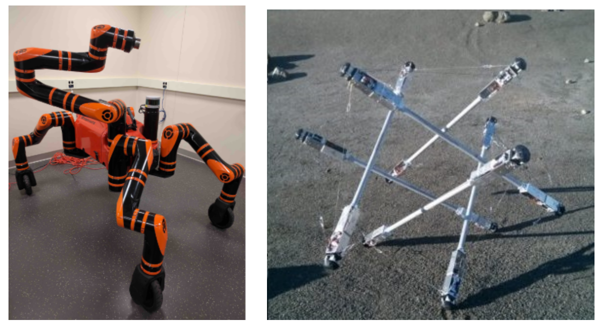
Future Directions:

I intend to continue working on general motion planning and kinodynamic planning with the objective of further developing tools to better solve motion planning problems and exploring novel applications for motion planning methods. I also plan to continue working on applications of machine learning to motion planning and kinodynamic planning. The next steps in my research will be to further develop the work I am doing on kinodynamic planning for terrains by applying it to more complicated terrains and to different types of robots. I also plan to further develop reachable volumes and to apply them to real world problems.

I will also continue developing methods for applying the concept of affordances to task and motion planning. I also see myself potentially branching out into areas such as belief space planning, parallel robotics, computational geometry and computational biology.

Expansion of Work in Kinodynamic Planning for Terrains My next immediate step will be to look at different methods for generating local goals. In particular, I plan to develop learning based methods for identifying local goals in terrain environment. Such a method can be used in combination with the terrain-aware controller that I have developed.

I am also in the process of developing a version of the terrain-aware planner which only does planning over local terrain. This planner will maintain a local window of terrain around the robot in which it does planning, while applying a high-level representation of the environment (such as a cost-map or low-resolution heightmap) to generate local goals. The main bottleneck with the terrain-aware planner is the cost of simulating large, complex environments, and only needing to simulate a window of the environment will greatly increase the size and resolution of environments the planner can handle. This method will also allow us to apply the terrain-aware planner to real-world applications where the robot only has high-resolution information about the portion of the terrain it can observe with its sensors.



(a) (b)

Figure 10: a) Robomantis legged robot b) NASA tensegrity robot

I also plan to develop a variation of the terrain-aware controller that takes into account types of terrain (e.g. pavement, grass, sand, ect.). A controller which takes into account terrain composition will be better able to select appropriate controls and should give better performance in environments with different types of terrain.

Finally, I plan to apply my methods to different types of robots. In particular, I would like to develop a terrain-aware controller that can be used by legged robots such as the robomantis (Figure 10(a)). I would also like to try applying my method to unique robots such as NASA's tensegrity robot (Figure 10(b)).

Further Development of Reachable Volumes: My next objective will be to develop a probabilistic reachable volume where the reachable volumes of joints are represented by probability distributions. Probabilistic reachable volumes will give us the ability to track the position of joints as probability distributions while enforcing constraints. We foresee them being applied to medical robotics such as steering needles where there is a great deal of uncertainty in the environment and the motions of the robot. We also see them being applied to industrial and home robotics where lower cost, mass produced robots may not be as precise as those encountered in the lab, requiring methods that adapt to noise in robot motion. Probabilistic reachable volumes allow is to do planning under uncertainty for manipulators and articulate robots. They also have applications in belief space planning and could be used to apply SLAM-based methods to articulated robots.

Applications of Reachable Volumes: Another area of future research is to explore applying reachable volumes to other problems. I am particularly interested in applying it to computational biology problems such as protein folding. These problems have a large number of degrees of freedom, and are well suited for reachable volumes. To facilitate this I intend to collaborate with bio-medical researchers and with researchers in the drug industry. I also expect that the bio-medical applications of my work will will help me to obtain funding from pharmaceutical companies and from organizations such as the National Institute of

Health.

I would also like to explore applying reachable volumes to folding robots. Motion planning for folding robots has applications in areas such as deformable materials (eg. shape-memory alloys).

I also plan to explore how reachable volumes can improve control and interaction with high degree of freedom robots. This will involve further development of the reachable volume visualization tool so it can be used to provide feedback that will assist in robot control. It will also involve applying reachable volumes and reachable volume sampling to user guided motion planning where reachable volumes can be used to guide planning and provide feedback.

Task Planning with Manipulation Affordances: Affordances provide an atomic representation of actions that could be used for high level task planning. I plan to explore combining the motion planning tools I developed for affordance with high level task planning methods in order to produce complex sequences of motions needed to solve complicated, multi-step tasks.

References

[1] S. Anderson, C. Mcmanus, H. Dong, E. Beerepoot, and T. Barfoot, "The gravel pit lidar-intensity imagery dataset," 01 2013.

[2] J. Barraquand and J.-C. Latombe, "Nonholonomic multibody mobile robots: controllability and motion planning in the presence of obstacles," in *Proceedings. 1991 IEEE International Conference on Robotics and Automation*, 1991, pp. 2328–2335 vol.3.

[3] J. Choi, "Kinodynamic motion planning for autonomous vehicles," *International Journal of Advanced Robotic Systems*, vol. 11, no. 6, p. 90, 2014. [Online]. Available: https://doi.org/10.5772/58683

[4] E. Frazzoli, M. A. Dahleh, and E. Feron, "Real-time motion planning for agile autonomous vehicles," *Journal of guidance, control, and dynamics*, vol. 25, no. 1, pp. 116–129, 2002.

[5] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE transactions on Robotics and Automation*, vol. 12, no. 4, pp. 566–580, 1996.

[6] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *The international journal of robotics research*, vol. 30, no. 7, pp. 846–894, 2011.

[7] S. M. Lavalle, "Rapidly-exploring random trees: A new tool for path planning," Tech. Rep., 1998.

[8] Y. Li, Z. Littlefield, and K. E. Bekris, "Asymptotically optimal sampling-based kinodynamic planning," *The International Journal of Robotics Research*, vol. 35, no. 5, pp. 528–564, 2016.

[9] K. Hauser and Y. Zhou, "Asymptotically optimal planning by feasible kinodynamic planning in a statecost space," *T-RO*, vol. 32, no. 6, p. 14311443, 2016.

[10] M. Kleinbort, E. Granados, K. Solovey, R. Bonalli, K. E. Bekris, and D. Halperin, "Refined analysis of asymptotically-optimal kinodynamic planning in the state-cost space," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 6344–6350.

[11] Z. Littlefield and K. E. Bekris, "Efficient and asymptotically optimal kinodynamic motion planning via dominance-informed regions," in *IROS*, 2018.

[12] E. Coumans and Y. Bai, “PyBullet, a python module for physics simulation for games, robotics and machine learning,” http://pybullet.org, 2016–2019.

[13] E. Todorov, T. Erez, and Y. Tassa, “Mujoco: A physics engine for model-based control,” in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 5026–5033.

[14] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, “Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor,” in *ICML*, 2018.

[15] M. Andrychowicz, D. Crow, A. Ray, J. Schneider, R. H. Fong, P. Welinder, B. McGrew, J. Tobin, P. Abbeel, and W. Zaremba, “Hindsight experience replay,” in *NeurIPS*, 2017.

[16] J.-P. Merlet, “Still a long way to go on the road for parallel mechanisms,” 10 2002.

[17] G. Oriolo and C. Mongillo, “Motion planning for mobile manipulators along given end-effector paths,” Barcelona, Spain, 2005, pp. 2154–2160.

[18] J. Corts and T. Simon, *Sampling-Based Motion Planning under Kinematic Loop-Closure Constraints*, 12 2005, pp. 75–90.

[19] M. Kallmann, A. Aubel, T. Abaci, and D. Thalmann, “Planning collision-free reaching motions for interactive object manipulation and grasping,” *Computer Graphics Forum*, vol. 22, no. 3, pp. 313–322, 2003. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/1467-8659.00678

[20] S. LaValle, J. Yakey, and L. Kavraki, “A probabilistic roadmap approach for systems with closed kinematic chains,” in *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No.99CH36288C)*, vol. 3, 1999, pp. 1671–1676 vol.3.

[21] L. Han and N. Amato, “A kinematics-based probabilistic roadmap method for closed chain systems,” *Proc. 2000 International Workshop on Algorithmic Foundations of Robotics (WAFR)*, 01 2000.

[22] L. Han and L. Rudolph, “A unified geometric approach for inverse kinematics of a spatial chain with spherical joints,” in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, 2007, pp. 4420–4427.

[23] X. Tang, S. L. Thomas, P. Coleman, and N. M. Amato, “Reachable distance space: Efficient sampling-based planning for spatially constrained systems,” *I. J. Robotic Res.*, vol. 29, no. 7, pp. 916–934, 2010. [Online]. Available: http://dx.doi.org/10.1177/0278364909357643

[24] T. McMahon, S. Thomas, and N. M. Amato, “Sampling-based motion planning with reachable volumes for high-degree-of-freedom manipulators,” *The International Journal of Robotics Research*, vol. 37, no. 7, pp. 779–817, 2018. [Online]. Available: https://doi.org/10.1177/0278364918779555

[25] X. Tang, S. Thomas, P. Coleman, and N. M. Amato, “Reachable distance space: Efficient sampling-based planning for spatially constrained systems,” *The International Journal of Robotics Research*, vol. 29, no. 7, pp. 916–934, 2010. [Online]. Available: https://doi.org/10.1177/0278364909357643

[26] L.-C. T. Wang and C. C. Chen, “A combined optimization method for solving the inverse kinematics problem of mechanical manipulators,” vol. 7, no. 4, pp. 489–499, 1991.

[27] L. Han and N. M. Amato, “A kinematics-based probabilistic roadmap method for closed chain systems,” Department of Computer Science, Texas A&M University, Tech. Rep. TR 00-003, 2000.

[28] T. McMahon, S. Thomas, and N. M. Amato, “Reachable volume RRT,” Seattle, Wa., May 2015, pp. 2977–2984.

[29] T. McMahon, R. Sandstrom, S. L. Thomas, and N. M. Amato, “Manipulation planning with directed reachable volumes,” Vancouver, Canada, Sept 2017.

[30] T. McMahon, S. A. Jacobs, B. Boyd, L. Tapia, and N. M. Amato, “Local randomization in neighbor selection improves PRM roadmap quality,” Algarve, Portugal, October 2012, pp. 4441–4448.