Preparation of Papers for IEEE Sponsored Conferences & Symposia*

Albert Author[1] and Bernard D. Researcher[2]

***Abstract*— This electronic document is a Olive template. The various components of your paper [title, text, heads, etc.] are already defined on the style sheet, as illustrated by the portions given in this document.**

I. INTRODUCTION

A. Motion Planning

Kinematic redundancy [41], [42] enables a manipulator to follow a predefined task space trajectory using the endeffector (EE), while simultaneously, optimising for an additional task with the remaining movement capacity without impacting the trajectory adherence. This is possible because the robot's degrees of freedom (DOF) go beyond what is required to perform the primary task. Consequently, the robot can adopt different joint configurations optimised according to the secondary task while performing the primary task. Common secondary tasks [43] include avoiding singularities, optimising the manipulability measure, minimising joint torques and avoiding obstacles in the operating space.

The task of finding the joint trajectories of a manipulator is called motion planning [1]. It consists of finding a sequence of joint configurations for a robot so that the robot can move along this path from its initial configuration to the goal configuration without colliding with itself, static obstacles or other agents in the environment. In addition to collision avoidance, motion planning for manipulators can optionally take into account various constraints, such as position, velocity, acceleration or jerk constraints for the joint angle or end effector, precision of the end effector with respect to position and orientation, stability of the manipulator, avoidance of singularities, or any number of other criteria.

There are numerous methods for planning manipulator movements [33], [34], they can rougly be seperated into global and local approaches. Global, sampling-based, methods such as PRM [35], RRT* [36], [38], RRT-Connect [40], Informerd RRT* [39], BIT* [37] and others offer a globally optimal solution based on a global search in configuration space. However, the generated trajectories are not always smooth or optimal, and the performance of the methods may be insufficient for real-time operation.

~~Recently, a number of learning-based methods using data-driven techniques have been proposed to improve or accelerate the functionality of sampling-based methods.~~

~~Trajectory optimisation methods such as CHOMP, STOMP and TrajOpt, on the other hand, use optimization to improve an initial seed trajectory. Consequently, the optimality of the solution is highly dependent on this initial trajectory. Nevertheless, these methods are capable of generating smooth trajectories, and although they can be too computationally intensive for high DOF dynamic real-time environments, they are generally effective in finding constrained motion plans.~~

Local motion planning approaches employ optimization techniques, two common ones are inverse kinematics [11], [20], that is based on finding a least squares solution of the manipulator joint velocities, and quadratic programming (QP) [2], [3], [4], [5]. Both methods are fast, suitable for real-time applications in dynamic environments and provide smooth solutions. However, since they do not plan further than one step ahead, they tend to get stuck in local minima. Therefore, they are often combined with a higher-level planner, for global static environment based ~~motion~~ path planning, while local optimisation takes dynamic environment changes into account. In the following text we will focus on inverse kinematics based approaches.

B. Kinematic Obstacle Avoidance

~~Different control schemes have been proposed. Acceleration-based control excels in precise handling of motion changes, velocity-based control offers consistent and smooth movement, while force and torque-based control provides direct control over joint forces for robust physical interactions. (CITATIONS in COMMENT BELLOW)~~

Various researchers have adopted different approaches to kinematic avoidance, as detailed in [12], [21].

Colbaugh and Glass (1989) [13], [14] tackled this problem in a two-step process. Initially, they calculated the robot's end-effector trajectory. Subsequently, they used optimization to enhance the robot's dynamic response during trajectory execution and for tasks like obstacle avoidance.

Sciavicco and Siciliano [16], [17], as well as Egeland [18], independently introduced the concept of task-space augmentation, later revisited by Searji [19]. This method involves extending the primary task Jacobian with the secondary task Jacobian into a square matrix, yielding a singular solution.

Maciejewski and Klein (1985) [11], and Nakamura and Hanafusa (1987) [20], presented the concept of task prioritization and a null space for avoidance. In this framework, the secondary task is restricted to velocities that do not directly affect the primary task's movement. Their method calculates avoidance joint velocities based on the minimum distance between a point on the manipulator and an avoidance point,

*This work was not supported by any organization

[1]Albert Author is with Faculty of Electrical Engineering, Mathematics and Computer Science, University of Twente, 7500 AE Enschede, The Netherlands albert.author@papercept.net

[2]Bernard D. Researcheris with the Department of Electrical Engineering, Wright State University, Dayton, OH 45435, USA b.d.researcher@ieee.org

object, or obstacle. Žlajpah [23] improved the concept by proposing a reduced operational space formulation, that reshapes the Jacobain of the avoidance task from three cartesian axis to only the direction of obstacle avoidance. Petrič [24] suggested a way to smoothly transition between avoidance and trajectory following tasks.

C. Artificial Potential Field

These approaches typically do not focus on the environmental context, often representing it as geometric primitives and then calculating the distance between these primitives and the lines describing the manipulator.

Khatib [15] proposes the concept of an Artificial Potential Field. This field consists of a repulsive component that deflects the end effector away from obstacles, depicted as geometric primitives, and an attractive component that draws the manipulator towards its target. In the following years many different modifications and improvement of the original APF idea have been proposed, often focused on removal of local minimas in the potential field. Khim and Khosla [22] suggested the use of harmonic functions to solve the problem of local minima. Pinto et al. [25] proposes to vary the field based on the distance of robot from obstacles to fill the local minima. Many researchers tried using APF as a heuristic to better guide sampling based approaches [27], [28], [29].

Many of the recent works focus on use of a variation of artificial potential field to plan motion of the manipulator. Xia et al. [31] uses a variation of APF for manipulator motion [31]. Park et al. [61] used a numerical Jacobian in combination with APF for motion planning. Zhang et al. [60] proposes dynamic repulsive field based on direction and speed between point on robot and obstacle, it also suggests decision making force that moves the robot away from certain local minima. Chen et al. proposes~~ an application ~~using APF in joint space and a variable kinematic optimization step [32]. Long [26] suggests creating motion plan of the manipulator using APF, he extends it using RRT to calculate virtual attractive point for the robot to move towards in case of local minima. Zhu et al. [30] proposes use of APF in combination with MPC, to plan in environment with dynamic obstacles.

D. Environment representation

To be able to generate collision free trajectory we need to have a representation of the static environment. One common approach involves enclosing the obstacles around the manipulator within a collection of simple geometric primitives and then constructing a tree-like structure to facilitate efficient navigation and path planning [44]. Two of the commonly used methods for this purpose include AABB (axis-aligned bounding boxes) [46], [47], [59] and OBB (oriented bounding boxes) [45], [48]~~and FDH (fixed directions hulls).~~. The problem is generation of such hulls based on partial sensor measurements of the environment. Han et al. [62] used a complicated pipeline to convert point-cloud sensor measurements to octree, than to voxel grid and finally into convex hulls, used for collision detections. Another significant method worth mentioning is the use of Octomaps [50]. Octomaps employ an octree data structure to represent 3D environments efficiently, making them particularly suitable for areas with large open spaces.

In real environment we usually collect depth data with LiDAR, radar of RGBD camera sensors, that usually return clouds of points, that tell us the distance of objects in robots environment. While approaches have been proposed to use the point-cloud data directly to find obstacle free areas for robot operation [51], the data is often further converted into voxel grids [54], [49], [53].

Voxel grid [54], [52] representation divides the space into a regular grid of volumetric elements, or voxels, which can be used to create a more manageable approximation of the environment. While this approach offers a balance between detail and computational efficiency, it can introduce discretization errors, particularly when modeling objects with smooth surfaces or intricate details. The fidelity of the representation is dependent on the size of the voxels: smaller voxels can capture more detail but require more memory and computation, while larger voxels result in coarser approximations but are more memory and computation efficient. Adaptive voxel grids have been explored [54], where the voxel size can vary throughout the space to provide higher resolution in regions of interest while conserving resources in less critical areas. Nießner et al. [55] proposed Voxel hashing, for more efficient memory management in sparse environments. Dryanovski et al. [68] proposed variable voxel height and saving information about occupied and about free measurements, as well as the information about number of measurements for a specific cell.

Voxel grids can incorporate probabilistic information [57], [68], such as in occupancy grid maps, where each voxel holds a probability indicating the chance of an obstacle's presence. The occupancy probability of a voxel can be updated dynamically using sensor measurements and Bayesian updating methods. As new sensor data is collected, the probabilities are revised to reflect the increased or decreased likelihood of the presence of an obstacle in the voxel space.

Another type of data that voxel ~~grids~~ fields can hold are ESDF (Euclidian Signed Distance Field) or TSDF (Truncated Signed Distance Field) [49], in which case each of the voxels contains information how far nearest obstacle is in its vicinity.

E. ESDF creation

ESDF grids can be generated directly using sensor measurements. Oleynikova et al. [49] proposed a method for calculating the ESDF from TSDF. Han et al. [53] proposed a way to integrate point-cloud data into ESDF using ray-casting.

Another way is to generate ESDF from occupancy grids, that can be previously generated using sensors or based on predefined maps. When generating ESDF from occupancy grid, common approach is the Brushfire method [58], that spreads from obstacles until it calculates the distance for every field on a grid. Jump Flooding Algorithm (JFA) [63] is a similar method, that can be implemented on a GPU for faster parallelized distance calculation.

Methods that allow for the dynamical distance field generation are rare. Zhou et al. [67] propose use of pairs of points on trajectory and obstacles in their quadcopter trajectory optimization algorithm.

-Wavefront
-D* Klančar
-Euclidian distnace algorithms
-distance transforms
-voronoi diagrams
-ray casting, bullet casting ...

10-ish citatov iz seznama ni klicanih v tekstu !

-fix citation styles (sometimes et al, sometimes not, sometimes names, sometimes years ... not same format)

References

[1] IDEAS Lab, "Motion and Path Planning," presented at Purdue University, 2023. [Online]. Available: `https://ideas.cs.purdue.edu/research/robotics/planning/`. Accessed on: Jan. 9, 2024.

[2] E. A. Basso and K. Y. Pettersen, "Task-Priority Control of Redundant Robotic Systems using Control Lyapunov and Control Barrier Function based Quadratic Programs," in IFAC-PapersOnLine, vol. 53, no. 2, pp. 9037–9044, 2020. doi: 10.1016/j.ifacol.2020.12.2024.

[3] H. Toshani and M. Farrokhi, "Real-time inverse kinematics of redundant manipulators using neural networks and quadratic programming: A Lyapunov-based approach," Robotics and Autonomous Systems, vol. 62, no. 6, pp. 766–781, Jun. 2014. doi: 10.1016/j.robot.2014.02.005.

[4] J. Haviland and P. Corke, "NEO: A Novel Expeditious Optimisation Algorithm for Reactive Motion Control of Manipulators," IEEE Robot. Autom. Lett., vol. 6, no. 2, Art. no. 2, Apr. 2021, doi: 10.1109/LRA.2021.3056060.

[5] Y. Zhang, S. S. Ge, and T. H. Lee, "A Unified Quadratic-Programming-Based Dynamical System Approach to Joint Torque Optimization of Physically Constrained Redundant Manipulators," IEEE Trans. Syst., Man, Cybern. B, vol. 34, no. 5, pp. 2126–2132, Oct. 2004. doi: 10.1109/TSMCB.2004.830347.

[6] J. Nakanishi, R. Cory, M. Mistry, J. Peters, and S. Schaal, "Comparative experiments on task space control with redundancy resolution," in Proc. 2005 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, Edmonton, Alta., Canada, 2005, pp. 3901–3908. doi: 10.1109/IROS.2005.1545203.

[7] M. H. Raibert and J. J. Craig, "Hybrid Position/Force Control of Manipulators," Journal of Dynamic Systems, Measurement, and Control, vol. 103, no. 2, pp. 126–133, Jun. 1981. doi: 10.1115/1.3139652.

[8] T. Yoshikawa, "Dynamic hybrid position/force control of robot manipulators–Description of hand constraints and calculation of joint driving force," IEEE J. Robot. Automat., vol. 3, no. 5, pp. 386–392, Oct. 1987. doi: 10.1109/JRA.1987.1087120.

[9] O. Khatib, "A unified approach for motion and force control of robot manipulators: The operational space formulation," IEEE J. Robot. Automat., vol. 3, no. 1, pp. 43–53, Feb. 1987. doi: 10.1109/JRA.1987.1087068.

[10] N. Hogan, "Impedance Control: An Approach to Manipulation," in Proc. 1984 American Control Conf., San Diego, CA, USA, Jul. 1984, pp. 304–313. doi: 10.23919/ACC.1984.4788393.

[11] A. A. Maciejewski and C. A. Klein, "Obstacle Avoidance for Kinematically Redundant Manipulators in Dynamically Varying Environments," The International Journal of Robotics Research, vol. 4, no. 3, pp. 109–117, Sep. 1985. doi: 10.1177/027836498500400308.

[12] L. Lajpah and T. Petri, "Obstacle Avoidance for Redundant Manipulators as Control Problem," in Serial and Parallel Robot Manipulators - Kinematics, Dynamics, Control and Optimization, S. Kucuk, Ed. InTech, 2012. doi: 10.5772/32651.

[13] R. Colbaugh and K. Glass, "Cartesian control of redundant robots," J. Robotic Syst., vol. 6, no. 4, pp. 427–459, Aug. 1989. doi: 10.1002/rob.4620060409.

[14] K. Glass, R. Colbaugh, D. Lim, and H. Seraji, "Real-time collision avoidance for redundant manipulators," IEEE Trans. Robot. Automat., vol. 11, no. 3, pp. 448–457, Jun. 1995. doi: 10.1109/70.388789.

[15] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in 1985 IEEE International Conference on Robotics and Automation Proceedings, Mar. 1985, pp. 500–505. doi: 10.1109/ROBOT.1985.1087247.

[16] L. Sciavicco and B. Siciliano, "A solution algorithm to the inverse kinematic problem for redundant manipulators," IEEE J. Robot. Automat., vol. 4, no. 4, pp. 403–410, Aug. 1988. doi: 10.1109/56.804.

[17] L. Sciavicco and B. Siciliano, "Solving the Inverse Kinematic Problem for Robotic Manipulators," in RoManSy 6, A. Morecki, G. Bianchi, and K. Kedzior, Eds., Boston, MA: Springer US, 1987, pp. 107–114. doi: 10.1007/978-1-4684-6915-8_9.

[18] O. Egeland, "Task-space tracking with redundant manipulators," IEEE J. Robot. Automat., vol. 3, no. 5, pp. 471–475, Oct. 1987. doi: 10.1109/JRA.1987.1087118.

[19] H. Seraji, "Configuration control of redundant manipulators: theory and implementation," IEEE Trans. Robot. Automat., vol. 5, no. 4, pp. 472–490, Aug. 1989. doi: 10.1109/70.88062.

[20] Y. Nakamura, H. Hanafusa, and T. Yoshikawa, "Task-Priority Based Redundancy Control of Robot Manipulators," The International Journal of Robotics Research, vol. 6, no. 2, pp. 3–15, Jun. 1987. doi: 10.1177/027836498700600201.

[21] B. Siciliano and O. Khatib, Eds., Springer Handbook of Robotics. in Springer Handbooks. Cham: Springer International Publishing, 2016. doi: 10.1007/978-3-319-32552-1.

[22] J.-O. Kim and P. Khosla, "Real-Time Obstacle Avoidance Using Harmonic Potential Functions," 1992. doi: 10.1109/70.143352.

[23] L. Zlajpah and B. Nemec, "Kinematic control algorithms for on-line obstacle avoidance for redundant manipulators," in Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems, Lausanne, Switzerland, 2002, pp. 1898–1903. doi: 10.1109/IRDS.2002.1044033.

[24] T. Petrič and L. Žlajpah, "Smooth continuous transition between tasks on a kinematic control level: Obstacle avoidance as a control problem," Robotics and Autonomous Systems, vol. 61, no. 9, Art. no. 9, Sep. 2013. doi: 10.1016/j.robot.2013.04.019.

[25] M. F. Pinto, T. R. F. Mendonça, L. R. Olivi, E. B. Costa, and A. L. M. Marcato, "Modified approach using variable charges to solve inherent limitations of potential fields method," in Proc. 2014 11th IEEE/IAS International Conference on Industry Applications, Dec. 2014, pp. 1–6. doi: 10.1109/INDUSCON.2014.7059414.

[26] Z. Long, "Virtual target point-based obstacle-avoidance method for manipulator systems in a cluttered environment," Engineering Optimization, vol. 52, no. 11, Art. no. 11, Nov. 2020. doi: 10.1080/0305215X.2019.1681986.

[27] A. H. Qureshi and Y. Ayaz, "Potential Functions based Sampling Heuristic For Optimal Path Planning," Auton Robot, vol. 40, no. 6, Art. no. 6, Aug. 2016. doi: 10.1007/s10514-015-9518-0.

[28] A. H. Qureshi et al., "Adaptive Potential guided directional-RRT*," in Proc. 2013 IEEE International Conference on Robotics and Biomimetics (ROBIO), Shenzhen, China, Dec. 2013, pp. 1887–1892. doi: 10.1109/ROBIO.2013.6739744.

[29] J. Yi, Q. Yuan, R. Sun, and H. Bai, "Path planning of a manipulator based on an improved P_RRT* algorithm," Complex Intell. Syst., vol. 8, no. 3, pp. 2227–2245, Jun. 2022. doi: 10.1007/s40747-021-00628-y.

[30] T. Zhu, J. Mao, L. Han, C. Zhang, and J. Yang, "Real-Time Dynamic Obstacle Avoidance for Robot Manipulators Based on Cascaded Nonlinear MPC With Artificial Potential Field," IEEE Trans. Ind. Electron., pp. 1–11, 2023. doi: 10.1109/TIE.2023.3306405.

[31] X. Xia et al., "Path Planning for Obstacle Avoidance of Robot Arm Based on Improved Potential Field Method," Sensors, vol. 23, no. 7, Art. no. 7, Apr. 2023. doi: 10.3390/s23073754.

[32] Y. Chen, L. Chen, J. Ding, and Y. Liu, "Research on Real-Time Obstacle Avoidance Motion Planning of Industrial Robotic Arm Based on Artificial Potential Field Method in Joint Space," Applied Sciences, vol. 13, no. 12, p. 6973, Jan. 2023. doi: 10.3390/app13126973.

[33] S. M. LaValle, Planning Algorithms. Cambridge: Cambridge University Press, 2006.

[34] M. G. Tamizi, M. Yaghoubi, and H. Najjaran, "A review of recent trend in motion planning of industrial robots," Int J Intell Robot Appl, vol. 7, no. 2, Art. no. 2, Jun. 2023. doi: 10.1007/s41315-023-00274-2.

[35] P. Švestka and M. H. Overmars, "Motion planning for carlike robots using a probabilistic learning approach," The International Journal of Robotics Research, vol. 16, no. 2, pp. 119–143, 1997.

[36] S. LaValle, "Rapidly-exploring random trees: A new tool for path planning," Research Report 9811, 1998.

[37] J. D. Gammell, S. S. Srinivasa, and T. D. Barfoot, "Batch Informed Trees (BIT*): Sampling-based Optimal Planning via the Heuristically Guided Search of Implicit Random Geometric Graphs," in Proc. of the 2015 IEEE International Conference on Robotics and Automation (ICRA), May 2015, pp. 3067–3074. doi: 10.1109/ICRA.2015.7139620.

[38] S. Karaman and E. Frazzoli, "Incremental sampling-based algorithms for optimal motion planning," in Proc. Robotics: Science and Systems (RSS), 2010.

[39] J. D. Gammell, S. S. Srinivasa, and T. D. Barfoot, "Informed RRT*: Optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic," in Proc. of the 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, Chicago, IL, USA, Sep. 2014, pp. 2997–3004. doi: 10.1109/IROS.2014.6942976.

[40] J. J. Kuffner and S. M. LaValle, "RRT-connect: An efficient approach to single-query path planning," in Proceedings of the 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics

and Automation. Symposia Proceedings (Cat. No.00CH37065), Apr. 2000, pp. 995–1001 vol.2. doi: 10.1109/ROBOT.2000.844730.

[41] B. Siciliano, "Kinematic control of redundant robot manipulators: A tutorial," J. Intell. Robot. Syst., vol. 3, no. 3, Art. no. 3, 1990, doi: 10.1007/BF00126069.

[42] B. Siciliano and O. Khatib, Eds., Springer Handbook of Robotics, in Springer Handbooks. Cham: Springer International Publishing, 2016. doi: 10.1007/978-3-319-32552-1.

[43] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, Robot. Model. Plan. Control, 2010, pp. 161–189. [Online]. Available: http://link.springer.com/10.1007/978-1-84628-642-1_4

[44] Y. Dai, C. Xiang, Y. Zhang, Y. Jiang, W. Qu, and Q. Zhang, "A Review of Spatial Robotic Arm Trajectory Planning," Aerospace, vol. 9, p. 361, Jul. 2022, doi: 10.3390/aerospace9070361.

[45] S. Gottschalk, M. C. Lin, and D. Manocha, "OBBTree: a hierarchical structure for rapid interference detection," in Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques, ACM, Aug. 1996, pp. 171–180. doi: 10.1145/237170.237244.

[46] G. van den Bergen, "Efficient Collision Detection of Complex Deformable Models using AABB Trees," Journal of Graphics Tools, vol. 2, no. 4, pp. 1–13, 1997. doi: 10.1080/10867651.1997.10487480.

[47] G. Chen, D. Liu, Y. Wang, Q. Jia, and X. Zhang, "Path planning method with obstacle avoidance for manipulators in dynamic environment," International Journal of Advanced Robotic Systems, vol. 15, no. 6, Art. no. 1729881418820223, Nov. 2018, doi: 10.1177/1729881418820223.

[48] D. Puiu and F. Moldoveanu, "Real-time collision avoidance for redundant manipulators," in Proc. of the 2011 6th IEEE International Symposium on Applied Computational Intelligence and Informatics (SACI), Timisoara, Romania, 2011, pp. 403–408, doi: 10.1109/SACI.2011.5873037.

[49] H. Oleynikova, Z. Taylor, M. Fehr, R. Siegwart, and J. Nieto, "Voxblox: Incremental 3D Euclidean Signed Distance Fields for onboard MAV planning," in Proc. of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Sep. 2017, pp. 1366–1373. doi: 10.1109/IROS.2017.8202315.

[50] K. M. Wurm, A. Hornung, M. Bennewitz, C. Stachniss, and W. Burgard, "OctoMap: A Probabilistic, Flexible, and Compact 3D Map Representation for Robotic Systems," [Details of publication, e.g., in Proc. of the Conference/Journal Name, Year, pp. Page numbers]. [DOI or URL if available].

[51] F. Gao, W. Wu, W. Gao, and S. Shen, "Flying on point clouds: Online trajectory generation and autonomous navigation for quadrotors in cluttered environments," Journal of Field Robotics, vol. 36, no. 4, pp. 710–733, 2019, doi: 10.1002/rob.21842.

[52] A. Elfes, "Using occupancy grids for mobile robot perception and navigation," Computer, vol. 22, no. 6, pp. 46–57, Jun. 1989, doi: 10.1109/2.30720.

[53] L. Han, F. Gao, B. Zhou, and S. Shen, "FIESTA: Fast Incremental Euclidean Distance Fields for Online Motion Planning of Aerial Robots," arXiv, Jul. 26, 2019. Accessed: Jan. 11, 2024. [Online]. Available: http://arxiv.org/abs/1903.02144

[54] Y. Xu, X. Tong, and U. Stilla, "Voxel-based representation of 3D point clouds: Methods, applications, and its potential use in the construction industry," Automation in Construction, vol. 126, p. 103675, Jun. 2021, doi: 10.1016/j.autcon.2021.103675.

[55] M. Nießner, M. Zollhöfer, S. Izadi, and M. Stamminger, "Real-time 3D reconstruction at scale using voxel hashing," ACM Trans. Graph., vol. 32, no. 6, pp. 1–11, Nov. 2013, doi: 10.1145/2508363.2508374.

[56] I. Dryanovski, W. Morris, and J. Xiao, "Multi-volume occupancy grids: An efficient probabilistic 3D mapping model for micro aerial vehicles," in Proc. of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, Oct. 2010, pp. 1553–1559. doi: 10.1109/IROS.2010.5652494.

[57] S. Thrun, Probabilistic robotics, vol. 45, 2002. Accessed: Jun. 14, 2023. [Online]. Available: https://dl.acm.org/doi/10.1145/504729.504754

[58] B. Lau, C. Sprunk, and W. Burgard, "Improved updating of Euclidean distance maps and Voronoi diagrams," in Proc. of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, Oct. 2010, pp. 281–286. doi: 10.1109/IROS.2010.5650794.

[59] L. Luo et al., "Collision-Free Path-Planning for Six-DOF Serial Harvesting Robot Based on Energy Optimal and Artificial Po-

tential Field," Complexity, vol. 2018, pp. 1–12, Nov. 2018, doi: 10.1155/2018/3563846.

[60] W. Zhang, H. Cheng, L. Hao, X. Li, M. Liu, and X. Gao, "An obstacle avoidance algorithm for robot manipulators based on decision-making force," Robotics and Computer-Integrated Manufacturing, vol. 71, p. 102114, Oct. 2021, doi: 10.1016/j.rcim.2020.102114.

[61] S.-O. Park, M. C. Lee, and J. Kim, "Trajectory Planning with Collision Avoidance for Redundant Robots Using Jacobian and Artificial Potential Field-based Real-time Inverse Kinematics," Int. J. Control Autom. Syst., vol. 18, no. 8, Art. no. 8, Aug. 2020, doi: 10.1007/s12555-019-0076-7.

[62] D. Han, H. Nie, J. Chen, and M. Chen, "Dynamic obstacle avoidance for manipulators using distance calculation and discrete detection," Robotics and Computer-Integrated Manufacturing, vol. 49, pp. 98–104, Feb. 2018, doi: 10.1016/j.rcim.2017.05.013.

[63] G. Rong and T.-S. Tan, "Jump flooding in GPU with applications to Voronoi diagram and distance transform," in Proceedings of the 2006 Symposium on Interactive 3D Graphics and Games - SI3D '06, Redwood City, California, 2006, p. 109. doi: 10.1145/1111411.1111431.

[64] R. Szeliski, "Computer Vision: Algorithms and Applications, 2nd Edition," Springer, 2021.

[65] A. Rosenfeld and J. L. Pfaltz, "Distance functions on digital pictures," Pattern Recognition, vol. 1, no. 1, pp. 33–61, Jul. 1968, doi: 10.1016/0031-3203(68)90013-7.

[66] P. F. Felzenszwalb and D. P. Huttenlocher, "Distance Transforms of Sampled Functions," Theory of Comput., vol. 8, no. 1, pp. 415–428, 2012, doi: 10.4086/toc.2012.v008a019.

[67] X. Zhou, Z. Wang, H. Ye, C. Xu, and F. Gao, "EGO-Planner: An ESDF-Free Gradient-Based Local Planner for Quadrotors," IEEE Robot. Autom. Lett., vol. 6, no. 2, pp. 478–485, Apr. 2021, doi: 10.1109/LRA.2020.3047728.

[68] I. Dryanovski, W. Morris, and J. Xiao, "Multi-volume occupancy grids: An efficient probabilistic 3D mapping model for micro aerial vehicles," in 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei: IEEE, Oct. 2010, pp. 1553–1559, doi: 10.1109/IROS.2010.5652494.