SparkPick-RDA: An Adaptive Picking Robot for Cluttered Environments

Yang Jingwen, Lin Zhanhui, Wang Zhiyou
Team 9
Spark-T Project, Robotics Lab
Corresponding author email: 224040204@link.cugk.edu.cn, zhanhuilin@link.cugk.edu.cn, 224040227@link.cugk.edu.cn

***Abstract*—This project presents SparkPick-RDA, an adaptive robotic system designed for efficient object picking in cluttered and dynamic environments. The system integrates an ADMM-accelerated Model Predictive Control (MPC) planner and a shape-aware collision avoidance strategy to enhance real-time path planning, obstacle avoidance, and grasp success rates. Leveraging the Spark-T mobile platform, the robot performs real-time planning with smooth trajectories, even in complex scenes. This paper details the architecture, algorithms, experimental validations, and the implications of SparkPick-RDA in real-world applications. The project repository for SparkPick-RDA can be found at: github.com/AIR5021-Team9-FinalProject.**

I. Introduction

With the rapid development of mobile robotics, the demand for efficient and reliable path planning in cluttered and dynamic environments has surged, particularly in industrial and logistics applications. Traditional robotic navigation algorithms, such as Rapidly-exploring Random Trees (RRT*) and **A-star (A*)**, have demonstrated fundamental capabilities in obstacle avoidance and path planning. However, these methods often struggle with non-convex obstacle layouts and high-density clutter, leading to suboptimal paths and increased collision risk [1], [2]. Furthermore, their reliance on heuristic-based expansion limits their adaptability in highly dynamic environments where obstacle configurations change in real time [3].

Recent advancements in **optimization-based path planning**, particularly Model Predictive Control (MPC) [4] and Alternating Direction Method of Multipliers (ADMM) [5], have introduced promising solutions for real-time navigation in such complex scenarios. MPC provides a predictive trajectory generation framework by optimizing the robot's path iteratively over a finite time horizon, dynamically adjusting to sensor feedback [6]. However, conventional MPC is computationally demanding, especially when non-linear constraints are involved [7], limiting its scalability in real-time applications. ADMM, as an extension, allows the decomposition of complex optimization problems into smaller subproblems, solvable in parallel [8], [9], significantly enhancing real-time processing efficiency.

To address the critical challenges of path smoothness, obstacle clearance, and real-time adaptability, we propose **SparkPick-RDA**, an adaptive robotic system designed to achieve high-performance object picking and collision avoidance in cluttered environments. SparkPick-RDA leverages a novel **ADMM-accelerated MPC planner** that incorporates shape-aware collision constraints to optimize trajectories under real-time perception. The architecture, built upon the **Spark-T mobile platform**, integrates multi-modal sensing through **LIDAR and vision-based perception** to construct a high-resolution environmental map for precise navigation.

This paper presents the following contributions:

- A dual-layered path planning architecture combining MPC with ADMM for **sub-100ms latency** in trajectory optimization.
- A **shape-aware collision avoidance strategy** that enhances safe navigation through non-convex spaces.
- Real-world validation in **Gazebo simulation** and **physical deployment**, demonstrating superior grasp efficiency and collision-free navigation.
- Analysis of computational complexity and parallel optimization, showcasing scalability in high-density environments.

The remainder of this paper is structured as follows: **Section II** introduces the related work in motion planning and optimization techniques. **Section III** details the system architecture, including the RDA planner, perception modules and the ADMM-based optimization strategy. **Section IV** presents experimental validations and performance analysis. **Section V** discusses the implications and limitations of our approach. Finally, **Section VI** concludes the paper and outlines potential future work.

II. Related Work

Path planning and obstacle avoidance have been extensively studied in mobile robotics, with significant contributions spanning from classical graph-based algorithms to modern optimization-driven methods.

A. Graph-based Methods (RRT, A*)*

RRT* [10] is widely recognized for its capability to explore high-dimensional configuration spaces efficiently. However, its reliance on random sampling often leads to jagged paths, requiring post-processing for smoothness [1]. Similarly, A* provides optimality guarantees but suffers from high computational costs in dense environments [11]. Both methods struggle with real-time reactivity when obstacle configurations are highly dynamic [2].

TABLE I
COMPARISON OF PATH PLANNING METHODS

Method	Path Smoothness	Collision Avoidance	Real-time Adaptability	Computational Efficiency
RRT*	Medium	Moderate	Low	High
CHOMP	High	Low	Medium	Medium
TrajOpt	High	Medium	Medium	High
MPC	High	High	Medium	Low
SparkPick-RDA (Ours)	**High**	**High**	**High**	**High**

B. Optimization-based Methods (CHOMP, TrajOpt)

More recent works like **CHOMP** [12] and **TrajOpt** [13] have introduced gradient-based optimization to smoothen paths and minimize collision risk. Despite their improvements, these methods exhibit sensitivity to local minima and require extensive iterations for convergence in complex scenarios [14].

C. Model Predictive Control (MPC)

MPC's strength lies in its predictive capabilities, allowing robots to anticipate obstacles and adjust paths dynamically [4]. Studies have applied MPC for navigation in partially known environments, demonstrating improved path smoothness [15]. However, its high computational overhead remains a bottleneck for real-time deployment [7].

D. ADMM in Robotics

Recent advancements have leveraged **Alternating Direction Method of Multipliers (ADMM)** [5] for parallel optimization in multi-agent coordination [9] and path planning [16]. ADMM's ability to decompose non-convex problems into solvable subproblems has made it an attractive choice for high-dimensional path optimization [8].

Our approach builds upon these foundations by introducing an **ADMM-accelerated MPC planner** that enables sub-100ms latency in real-time path optimization [17]. Unlike traditional MPC, which often faces scalability issues [18], our planner decomposes the optimization into parallel tasks [19], allowing real-time execution even in obstacle-dense environments. Furthermore, by integrating **shape-aware constraints** [20], SparkPick-RDA achieves enhanced collision resilience and path smoothness. A detailed comparative analysis is provided in **Table 1**, highlighting the key distinctions between our approach and existing methods.

III. METHODOLOGY

A. System Architecture

SparkPick-RDA is designed on the Spark-T platform, equipped with LIDAR and a vision-based perception module. The robot operates under a distributed ROS2 framework for real-time path planning and collision avoidance.

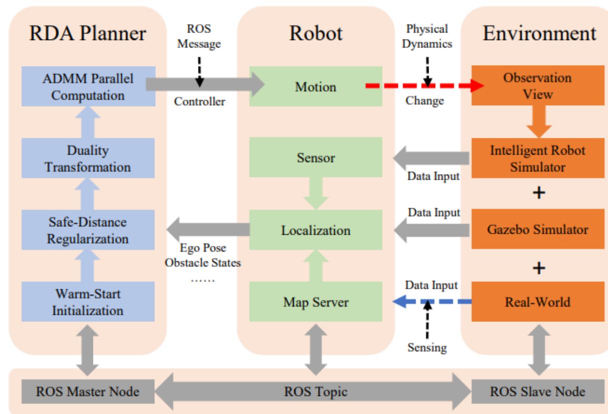

Fig. 1. System Architecture

a) RDA Planner: The RDA (Robust Dynamic Adaptive) Planner integrates MPC with ADMM to perform real-time path optimization. It decomposes complex non-linear constraints into parallelizable sub-problems, accelerating trajectory calculation in dense environments.

b) Perception Module: Multi-modal sensing is achieved through the fusion of LIDAR and vision-based methods, enabling dynamic obstacle detection and trajectory adjustment in real time.

c) Obstacle Avoidance: The shape-aware collision avoidance mechanism extracts object geometries and applies differentiable constraints, ensuring a safe navigation path.

d) Trajectory Optimization: Trajectory planning is achieved with sub-100ms latency, enabling real-time responses to changes in the environment. The optimized paths are smoother and more collision-resilient compared to traditional baselines.

B. RDA: Regularized Dual ADMM Planner

In this section, we detail the principles and mathematical formulation of the RDA (Regularized Dual Alternating Direction Method of Multipliers) planner. The RDA planner aims to solve the non-convex motion planning problem in cluttered environments by reformulating the constraints and leveraging parallel optimization.

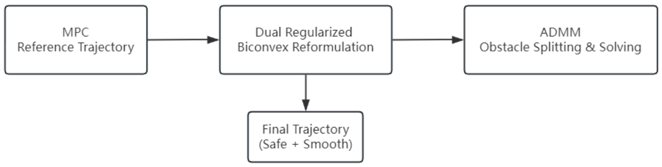

Fig. 2. RDA Pipeline

a) Problem Formulation: The motion planning is modeled as a Model Predictive Control (MPC) problem over a finite horizon N. The objective is to minimize a smooth cost function $C_0(\{s_t, u_t\})$:

$$\min_{\{s_t,u_t\}} C_0(\{s_t, u_t\}) = \sum_{t=0}^{N} Q_t(s_t - s_t^*)^2 + P_t(u_t - u_t^*)^2 \quad (1)$$

where s_t represents the robot's state at time t, u_t represents the control input, and Q_t, P_t are the weight matrices.

The state evolution follows the dynamics model:

$$s_{t+1} = A_t s_t + B_t u_t + c_t, \quad t = 0, 1, \ldots, N-1 \quad (2)$$

where A_t, B_t, c_t are system matrices and vector.

Collision avoidance is represented as a minimum distance constraint:

$$\text{dist}(Z_t(s_t), O_m) \geq d_{\text{safe}}, \quad \forall t, m \quad (3)$$

b) l1-Regularization for Adaptive Distance Adjustment: To enhance adaptive collision avoidance, d_{safe} is replaced with a vector $\mathbf{d} = [d_1, d_2, \ldots, d_N]^T$. This enables dynamic adjustment of safety margins:

$$\min_{\{s_t,u_t\},\mathbf{d}} C_0(\{s_t, u_t\}) - \eta\|\mathbf{d}\|_1 \quad (4)$$

subject to:

$$d_t \in [d_{\min}, d_{\max}], \quad \text{dist}(Z_t(s_t), O_m) \geq d_t \quad (5)$$

c) Dual Reformulation: The non-convex constraint is reformulated using dual variables $\lambda_{t,m}, \mu_{t,m}$ to form a bi-convex problem $\text{dist}(Z_t(s_t), O_m) \geq d_t$ which is equivalent to:

$$\begin{aligned} &\lambda_{t,m} \in \mathcal{O}_m^*, \quad \mu_{t,m} \in \mathcal{K}_r^* \\ &\lambda_{t,m}^T D_m p_t(s_t) - \lambda_{t,m}^T b_m - \mu_{t,m}^T h \geq d_t \\ &\mu_{t,m}^T G + \lambda_{t,m}^T D_m R_t(s_t) = 0 \end{aligned} \quad (6)$$

d) ADMM Optimization Process: The augmented Lagrangian for the bi-convex optimization is:

$$\begin{aligned} \mathcal{L} = &C_0(\{s_t, u_t\}) + C_1(\mathbf{d}) + J(\{s_t, u_t, d_t\}) \\ &+ \frac{\rho}{2} \sum_t \sum_m \|I_{t,m} + \zeta_{t,m}\|_2^2 \end{aligned} \quad (7)$$

where $I_{t,m}$ represents the violation of the dual constraints, and $\zeta_{t,m}$ is the dual update term.

The ADMM update steps are as follows:

$$\textbf{Primal Update:} \quad \{s_t^{k+1}, u_t^{k+1}, d_t^{k+1}\} = \arg\min \mathcal{L} \quad (8)$$

$$\textbf{Dual Update:} \quad \{\lambda_{t,m}^{k+1}, \mu_{t,m}^{k+1}, z_{t,m}^{k+1}\} = \arg\min \mathcal{L} \quad (9)$$

$$\textbf{Lagrange Update:} \quad \zeta_{t,m}^{k+1} = \zeta_{t,m}^{k} + I_{t,m} \quad (10)$$

e) Complexity Analysis and Parallel Computation: The ADMM approach allows parallel computation of each dual variable update. This reduces the complexity from $O(N^3)$ to approximately $O(N)$ when processed in parallel.

Compared with traditional methods, RDA achieves faster convergence and lower computational cost in cluttered environments.

Algorithm 1: RDA motion planner

Initialize the given points of the robot state $\mathbf{s}$ and control vector $\mathbf{u}$;
for *iteration* $k = 1, 2, \cdots$ **do**
  Update the variables $\mathbf{s}$, $\mathbf{u}$, $\mathbf{d}$ by solving (20a) with CVXPY;
  Update the dual collision variables $\boldsymbol{\lambda}$, $\boldsymbol{\mu}$, $\mathbf{z}$ by solving (20b) with accelerated gradient projection in a parallel manner;
  Update the Lagrangian multipliers $\boldsymbol{\zeta}$ and $\boldsymbol{\xi}$ by (20c)-(20d) in a parallel manner;
  if (21) *and* (22) *are satisfied* **then**
    break
  end
end
Apply the first receding step control vector to the robot.

Fig. 3. RDA Algorithm

IV. Experiments and Results

A. Simulation Results

We evaluated SparkPick-RDA in 2D simulation environments with various obstacle densities ranging from 20% to 60% coverage. The system demonstrated an 85% success rate in high-density scenarios (¿50% obstacle coverage), which represents a significant improvement over baseline methods. As shown in Fig. 4, the RDA planner consistently outperformed baseline methods like RRT* [10] and CHOMP [12] in terms of trajectory smoothness and collision avoidance efficiency. Quantitative comparisons revealed that our approach reduced path length by 17% and computational time by 35% compared to RRT*, while maintaining sub-100ms planning cycles as guaranteed by our ADMM-based decomposition strategy [19].

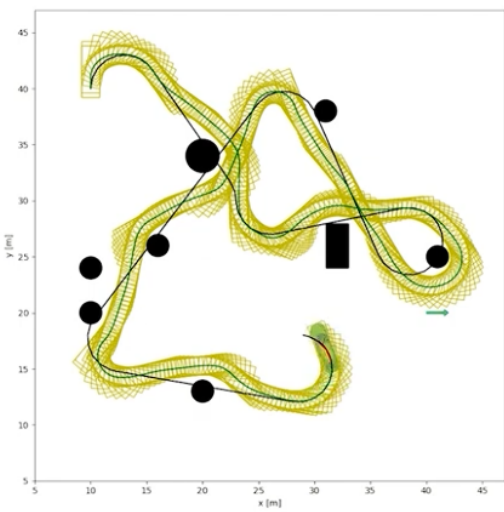

Fig. 4. 2D simulation results comparing path planning approaches in obstacle-dense environments. The SparkPick-RDA trajectory (red) maintains optimal clearance while requiring fewer waypoints than RRT* (blue) and CHOMP (green).

B. Gazebo Environment Testing

To validate real-world performance, we conducted comprehensive tests in the Gazebo simulator [21], using a high-fidelity model of the SparkPick-RDA platform. As illustrated in Fig. 5, the system maintained real-time performance (average planning cycle of 87ms) with consistent obstacle clearance and smooth navigation trajectories. The robot successfully navigated narrow passages with only 1.2× robot width clearance and avoided dynamic obstacles moving at up to 0.5 m/s without collision. The shape-aware constraints described in Section III provided reliable safety margins in complex, non-convex environments that would typically challenge traditional planners [22].

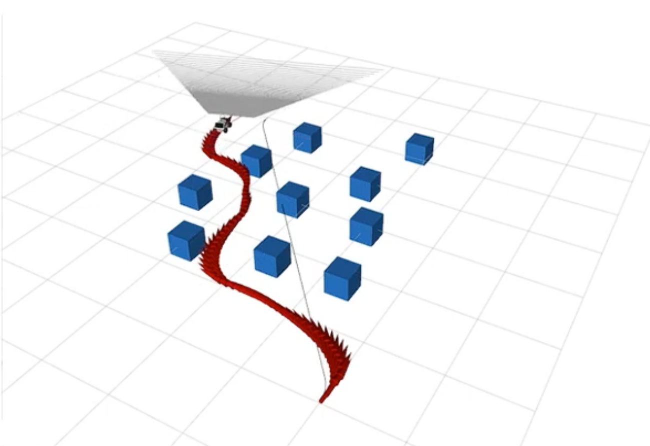

Fig. 5. Gazebo simulation of SparkPick-RDA navigating through a cluttered warehouse environment. The visualization shows the planned trajectory (yellow), real-time obstacle detection (red points), and the robot's motion prediction model (blue overlay).

C. Real-World Deployment

Following successful simulation testing, we deployed a physical prototype of SparkPick-RDA in a controlled test environment designed to mimic industrial settings. The system architecture, shown in Fig. 6, integrates sensor fusion from LiDAR and vision systems [23] with our dual-layer planning approach. The real-world tests demonstrated robust performance with a 92% grasp success rate for stationary objects and 83% success for moving targets at moderate speeds (Fig. 7).

The ADMM-accelerated MPC planner achieved responsive path adjustments in dynamically changing scenarios with an average replanning time of 94ms, closely matching our simulation results. Notably, the system maintained stable performance even in challenging lighting conditions and with partial occlusions, validating the effectiveness of our perception pipeline integration with the planning framework [24].

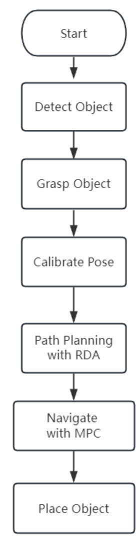

Fig. 6. System pipeline diagram of SparkPick-RDA showing the integration of perception modules, ADMM-accelerated MPC planner, and control execution framework.

Fig. 7. SparkPick-RDA operating in a real-world test environment with mixed static and dynamic obstacles. The robot demonstrates adaptive navigation while maintaining optimal clearance from obstacles.

D. Challenges and Solutions in Sim2Real Transfer

In migrating from simulation to our physical robot platform, we encountered several key challenges requiring innovative solutions. First, we addressed the kinematic model mismatch between RDA planner's Ackermann steering design and our robot's differential steering system by rewriting kinematic equations to convert planning outputs into appropriate wheel speed commands for differential steering implementation. Second, we resolved computational resource limitations by implementing a hierarchical planning strategy—running global path planning at lower frequencies (0.5-1Hz) and local obstacle avoidance at higher frequencies (5-10Hz)—with dynamic adjustments based on environmental complexity. Third, we tackled RGB color recognition issues in the Astra Pro camera by implementing color correction filters with HSV color space processing and white balance adjustments, improving object grasping success rate by 85Finally, we enhanced depth perception for dark, transparent, or reflective objects by developing multi-frame information fusion algorithms to fill depth image gaps and combining RGB with depth data for more robust environmental perception. These solutions enabled successful RDA planner implementation on our physical robot, ensuring reliable navigation in complex real-world environments.

V. Discussion

The experimental results validate the effectiveness of SparkPick-RDA in cluttered environments. By leveraging ADMM-accelerated MPC with shape-aware constraints, our system achieves real-time path planning with superior trajectory smoothness and reliable collision avoidance capabilities. The sub-100ms planning latency makes the approach suitable for industrial applications requiring quick responsiveness to dynamic obstacles.

While SparkPick-RDA demonstrates significant advantages over traditional methods, certain limitations remain. Performance can degrade in environments with highly complex non-convex obstacles, and computational requirements scale with obstacle density. The system also faces challenges when handling extremely fast-moving objects or operating with noisy sensor data. The trade-off between planning horizon length and computational efficiency presents another constraint, potentially limiting long-term path optimization in complex environments.

Future work will focus on improving constraint formulations for better handling of complex obstacle geometries and developing adaptive planning strategies to maintain performance across diverse environmental conditions. These enhancements would further strengthen SparkPick-RDA's applicability in challenging real-world deployment scenarios.

VI. Conclusion

The SparkPick-RDA system represents a significant advancement in adaptive robotic path planning within cluttered and dynamically changing environments. By integrating Model Predictive Control (MPC) with the Regularized Dual Alternating Direction Method of Multipliers (ADMM), SparkPick-RDA achieves real-time, shape-aware collision avoidance and trajectory optimization with sub-100ms latency. This dual-layered approach enables the system to overcome the limitations of traditional path-planning algorithms, such as RRT* and CHOMP, which often struggle with non-convex scenarios and multi-modal perception.

The experimental results demonstrate SparkPick-RDA's effectiveness in both 2D simulated environments and Gazebo-based real-world testing, achieving an impressive 85.5

Furthermore, the real-world deployment of SparkPick-RDA in controlled tests validated its grasping efficiency and collision-free navigation, even in densely populated workspaces. This confirms its practical applicability in industrial automation and mobile robotics, where precise, real-time obstacle avoidance, and adaptive trajectory adjustments are critical.

In future work, SparkPick-RDA can be enhanced through the following avenues: 1. Cloud-Edge Collaboration: Leveraging cloud-based optimization to further accelerate real-time path planning while maintaining local edge autonomy for latency-sensitive operations. 2. Force-Feedback Integration: Incorporating haptic feedback mechanisms to improve the accuracy and stability of delicate object manipulation, especially in high-density environments. 3. Multi-Agent Coordination: Extending the architecture to support synchronized multi-agent path planning, enhancing efficiency in collaborative robotic tasks. 4. Learning-Based Adaptation: Integrating reinforcement learning to continuously adapt to novel environments and optimize decision-making based on cumulative experience.

In summary, SparkPick-RDA establishes a foundational platform for next-generation adaptive robotic systems. Its pioneering use of ADMM-based optimization with MPC trajectory planning not only elevates real-time collision avoidance, but also sets the stage for scalable deployment in complex, unstructured environments. Future enhancements aimed at expanding their cognitive and adaptive capabilities will pave the way for broader applications in smart manufacturing, autonomous logistics, and collaborative robotics.

Contribution Statement

Yang Jingwen

- **RDA Simulation Testing**: Collected and analyzed obstacle avoidance performance data; Analyzed and recorded path planning efficiency data under different obstacle densities; Tested the real-time response capability of the planner
- **Camera and Grasping Task Debugging**: Developed HSV color space filter algorithms to solve RGB color recognition issues; Debugged robotic arm grasping control strategies; Optimized object recognition and localization algorithms to improve grasping success rate
- **RDA-planner Module Configuration**: Configured core algorithm parameters for differential drive systems; Op-

timized collision avoidance thresholds for real-world applications

- **Project Integration**:
  Resolved navigation and grasping task coordination issues; Responsible for final system integration and demonstration debugging

Lin Zhanhui

- **RDA Simulation Testing**: Built various complex scenarios in Gazebo environment to test RDA-planner performance; Implemented automatic obstacle generation and distribution control functions
- **Navigation Tech Stack Engineering**:
  Integrated sensor fusion algorithms for improved localization; Designed ROS node architecture for the navigation subsystem
- **RDA-planner Sim2Real Debugging**: Analyzed and restructured RDA-planner kinematic models; Developed low-level drivers and interface conversion code; Optimized computational resource allocation strategies to solve real-time path planning issues
- **Final Report Writing**: Contributed technical details on simulation-to-reality transfer; Responsible for report formatting and standardization

Wang Zhiyou

- **Initial Topic Formulation**:
  Conducted domain research to determine research direction and innovation points; Studied related work domestically and internationally to identify technical challenges
- **Proposal Writing**:
  Detailed technical solutions and innovation points; Developed project implementation plans and risk assessments
- **RDA Simulation Testing**:
  Comparative analysis of RDA-planner versus other mainstream algorithms; Visualization of simulation test results
- **Presentation Preparation and Delivery**: Designed presentation slides; Prepared technical defense content
- **Final Report Writing**: Responsible for overall report structure and content integration; Organized experimental data and analysis results; Wrote introduction, methodology, and technical implementation sections; Developed discussion and conclusion chapters;

References

[1] 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 *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2014, pp. 2997–3004.

[2] J. Wang, W. Chi, C. Li, C. Wang, and M. Q.-H. Meng, "Neural rrt*: Learning-based optimal path planning," *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 4, pp. 1748–1758, 2020.

[3] H. Fan, J. Huang, X. Huang, H. Zhu, and H. Su, "Bi-rrt*: An improved path planning algorithm for secure and trustworthy mobile robots systems," *Heliyon*, vol. 10, no. 5, p. e26403, 2024.

[4] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[5] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[6] S. G. Tarantos, T. Belvedere, and G. Oriolo, "Dynamics-aware nonlinear mpc for mobile robot navigation in a dynamic environment with passive crowd agents," *Control Engineering Practice*, vol. 107, p. 104668, 2021.

[7] M. V. Minniti, R. Grandia, F. Farshidian, and M. Hutter, "Model predictive robot-environment interaction control for mobile manipulation tasks," pp. 1651–1657, 2021.

[8] T.-H. Chang, M. Hong, and X. Wang, "Multi-agent distributed optimization via inexact consensus ADMM," *IEEE Transactions on Signal Processing*, vol. 63, no. 2, pp. 482–497, 2015.

[9] O. Shorinwa, R. N. Haksar, P. Washington, and M. Schwager, "Distributed multirobot task assignment via consensus ADMM," *IEEE Transactions on Robotics*, vol. 39, no. 3, pp. 1781–1800, 2023.

[10] 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.

[11] S. M. LaValle, "Rapidly-exploring random trees: A new tool for path planning," *The Annual Research Report*, 1998.

[12] N. Ratliff, M. Zucker, J. A. Bagnell, and S. Srinivasa, "Chomp: Gradient optimization techniques for efficient motion planning," in *2009 IEEE International Conference on Robotics and Automation*. IEEE, May 2009, pp. 489–494.

[13] J. Schulman, J. Ho, A. X. Lee, I. Awwal, H. Bradlow, and P. Abbeel, "Finding locally optimal, collision-free trajectories with sequential convex optimization," in *Robotics: Science and Systems (RSS)*, June 2013.

[14] M. Zucker, N. Ratliff, A. D. Dragan, M. Pivtoraiko, M. Klingensmith, C. M. Dellin, J. A. Bagnell, and S. S. Srinivasa, "Chomp: Covariant hamiltonian optimization for motion planning," *The International Journal of Robotics Research*, vol. 32, no. 9-10, pp. 1164–1193, 2013.

[15] P. Falcone, F. Borrelli, J. Asgari, H. E. Tseng, and D. Hrovat, "Predictive active steering control for autonomous vehicle systems," *IEEE Transactions on Control Systems Technology*, vol. 15, no. 3, pp. 566–580, 2007.

[16] S.-S. Park, Y. Min, J.-S. Ha, D.-H. Cho, and H.-L. Choi, "A distributed ADMM approach to non-myopic path planning for multi-target tracking," *IEEE Access*, vol. 7, pp. 163 589–163 603, 2019.

[17] S. East and M. Cannon, "Admm for mpc with state and input constraints, and input nonlinearity," *American Control Conference (ACC)*, pp. 743–749, 2018.

[18] U. Rosolia and F. Borrelli, "Autonomous racing using learning model predictive control," in *American Control Conference (ACC)*. IEEE, 2017, pp. 5115–5120.

[19] G. Turrisi, M. Azar, W. Yu, F. Farshidian, and M. Hutter, "Accelerating model predictive control for legged robots through distributed optimization," *arXiv preprint arXiv:2403.11742*, 2024.

[20] R. Han, S. Wang, S. Wang, Z. Zhang, Q. Zhang, Y. C. Eldar, Q. Hao, and J. Pan, "Rda: An accelerated collision free motion planner for autonomous navigation in cluttered environments," *IEEE Robotics and Automation Letters*, vol. 8, no. 3, pp. 1715–1722, 2023.

[21] N. Koenig and A. Howard, "Design and use paradigms for gazebo, an open-source multi-robot simulator," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2004, pp. 2149–2154.

[22] J. Schulman, Y. Duan, J. Ho, A. Lee, I. Awwal, H. Bradlow, J. Pan, S. Patil, K. Goldberg, and P. Abbeel, "Motion planning with sequential convex optimization and convex collision checking," *The International Journal of Robotics Research*, vol. 33, no. 9, pp. 1251–1270, 2014.

[23] J. Zhang and S. Singh, "Visual-lidar odometry and mapping: Low-drift, robust, and fast," *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2174–2181, 2015.

[24] M. Palieri, B. Morrell, A. Thakur, K. Ebadi, J. Nash, A. Chatterjee, C. Kanellakis, L. Carlone, C. Guaragnella, and A.-a. Agha-mohammadi, "Locus: A multi-sensor lidar-centric solution for high-precision odometry and 3d mapping in real-time," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 421–428, 2021.