Experimental Validation of Modified BP-RRT* for Dual-Arm Collaborative Robots in ROS: A Step Towards Real-Time Deployment

Josin Hippolitus A*, Senthilnathan R,

Department of Mechatronics Engineering,

SRM Institute of Science and Technology, Kattankulathur, Chennai, India

*Corresponding author: josinhippolitus@gmail.com

Abstract:

Path planning for dual-arm collaborative robots poses significant challenges due to high-dimensional configuration spaces and real-time execution constraints. In our previous research, we proposed the Modified BP-RRT* algorithm, demonstrating its superiority in generating quicker and more efficient collision-free trajectories. This study focuses on the experimental validation of the algorithm within the Robot Operating System (ROS) framework to assess its practical feasibility in real-world robotic applications. The algorithm is integrated with a dual-arm robotic system in a simulated ROS environment, and its performance is evaluated against conventional sampling-based planners. Key metrics such as path smoothness, computational efficiency, execution time, and success rate are analysed. The results confirm that the Modified BP-RRT* significantly reduces planning time while maintaining optimality and collision avoidance, making it well-suited for dynamic and industrial environments. This validation bridges the gap between theoretical advancements and real-world robotic deployment, paving the way for future applications in robotic assembly, human-robot collaboration, and autonomous manipulation.

Keyword: Path planning, Robot Operating System, MoveIt, Path Planning, Dual-Arm Collaborative Robots, Modified BP-RRT*, Motion Planning, Robot Manipulation, Trajectory Optimization, Autonomous Robotics.

1. Introduction

Collaborative dual-arm robots are increasingly employed in industrial automation, human-robot collaboration, and precision assembly due to their ability to perform complex bimanual manipulation tasks. Virtual production robots face efficiency limitations in real-time execution of path planning tasks due to handling operations within large dimensional spaces and dynamic physical restrictions and complete arm coordination requirements. The performance of traditional sampling-based motion planners such as Rapidly-exploring Random Trees (RRT) and its variants suffers from three main issues in cluttered environments: high complexity execution times and poor path quality and lengthy convergence periods. The Modified BP-RRT* algorithm [16] represents an improved Bi-directional Path RRT* version that achieves faster convergence speed and minimizes computation resources to produce smoother paths without losing probabilistic completeness and asymptotic optimality standards. Our research showed that the validated simulation results proved the effectiveness of the algorithm to produce safe robotic paths in less planning duration while operating with dual-arm robots. The practical implementation of this system needs additional tests to verify operational stability across actual robot applications. The researchers seek experimental verification of the Modified BP-RRT* algorithm in ROS to determine its potential usage in real-time motion planning. A ROS-based dual-arm robotic system implements the algorithm to assess its operational execution time together with path quality and computational performance metrics. The research completes its analysis by comparing the algorithm to usual motion planners to demonstrate practical strengths and constraints.

Major accomplishments of this work consist of a) Implementation and integration of the Modified BP-RRT* algorithm within the ROS environment for real-time dual-arm motion planning. b) Experimental validation of the algorithm's performance through extensive simulations and comparative analysis with existing planners. c) Evaluation of key performance metrics, such as computational efficiency, trajectory smoothness, and real-time feasibility, for collaborative robotic applications. The research uses theoretical and practical methodologies to create functional path planning techniques that become applicable for industrial automation as well as autonomous robotic control systems and human-robot collaboration. The research provides vital knowledge points to engineers and researchers who design efficient real-time motion planning solutions for robotic systems which have numerous degrees of freedom.

3. Related Work

The path planning process for dual-arm collaborative robots remains complex because it requires handling extensive search spaces and executing plans in real time. Researchers have developed sampling-based planners as main alternatives to handle these problems. The section evaluates fundamental motion planning algorithms and their applications in dual-arm robotics along with ROS path planning aspects before presenting the need for the Modified BP-RRT* algorithm.

3.1. Sampling-Based Motion Planners

Sampling-based motion planners are commonly employed for robotic systems possessing numerous degrees of freedom since they successfully navigate complicated constraints.

- Rapidly-exploring Random Tree (RRT) [1] is a foundational algorithm that incrementally explores the configuration space but does not guarantee path optimality.
- RRT* [2] introduced an optimization step that refines paths for improved smoothness and efficiency. However, it suffers from slow convergence.

- Bi-directional RRT (Bi-RRT) [3] extends RRT by growing two trees from the start and goal positions, significantly reducing planning time.
- The Bi-directional Path-RRT (BP-RRT) system from reference 4 adds path optimization routines with heuristic elements to enhance the search efficiency of Bi-RRT*. Despite its advantages, BP-RRT* struggles with execution speed in complex environments.

The Modified BP-RRT* algorithm expands previous methods through improved convergence speed and enhanced computational efficiency with assurance of high-quality dual-arm system paths.

3.2. Path Planning for Dual-Arm Collaborative Robots

The motion planning for dual-arm robots becomes more difficult because of the need to avoid collisions between arms while coordinating their workspaces during synchronized operations. Various techniques have emerged to handle these difficulties according to researchers.

a) Planners PRM and RRT* received extensions for multi-arm setups in [5] yet remain unable to adapt in real time.
b) Smoother trajectories can be achieved by STOMP [Stochastic Trajectory Optimization for Motion Planning] and CHOMP (Covariant Hamiltonian Optimization for Motion Planning) [6] which are optimization-based approaches. But they cost computationally high.
c) The application of learning-based approaches through reinforcement learning and deep learning techniques exists for dual-arm path planning in [7] while needing large amounts of training data along with difficulties in generalization.

Our approach focuses on enhancing sampling-based planners to achieve real-time, collision-free dual-arm motion planning with improved efficiency.

3.3. Motion Planning in ROS

Developing and validating robotic algorithms are performed using a commonly used framework called Robot Operating System (ROS). ROS provides:

- **MoveIt! Motion Planning Framework** [8]: Supports multiple planning algorithms, including RRT*, PRM, and CHOMP. However, its default planners often lack real-time performance for high-DOF robots.
- **Gazebo Simulation Environment**: Allows for physics-based validation of path planning algorithms in a simulated robotic environment.
- **Trajectory Execution & Control:** ROS provides controllers for executing planned motions on real robots, ensuring feasibility beyond simulation.

Despite these advantages, existing ROS planners struggle with real-time execution in collaborative robotics. This work integrates Modified BP-RRT* into ROS, demonstrating its applicability in a practical robotic environment.

3.4. Justification for Modified BP-RRT*

Based on the limitations identified in existing works, our approach addresses key gaps:

- **Faster Convergence:** Reduces computational time compared to RRT* and BP-RRT*.
- **Smoother Paths:** Enhances trajectory quality for real-world execution.
- **Real-Time Feasibility:** Integrates with ROS for practical implementation.

By experimentally validating Modified BP-RRT* in ROS, this study bridges the gap between theoretical path planning advancements and practical deployment in dual-arm collaborative robotics.

4. Methodology

This section describes Modified BP-RRT* integrated with Robot Operating System (ROS) framework that uses ABB YuMi 7-DoF dual-arm robotic system for the experimental setup. The methodology divides its work into three main sections including algorithm enhancements together with algorithm workflow and ROS-based implementation.

4.1. Algorithm Enhancements

Modified BP-RRT* includes new features to overcome obstacles during path planning for dual-arm collaborative robots:

- **Pre-Stage Partitioning of Grids:** During the pre-processing stage, the workspace is partitioned into grids which enables faster as well as efficient computation of collision-free paths. This approach reduces the overall search space and hence improves the algorithm's efficiency.
- **Adaptive Sampling Strategy:** Sampling density is dynamically adjusted based on workspace complexity, enhancing search efficiency by focusing computational resources on more complex regions.
- **Hybrid Cost Function:** A weighted combination of Euclidean distance and obstacle proximity is utilized to evaluate path quality, ensuring smoother and safer trajectories.
- **Optimized Rewiring Mechanism:** The algorithm incorporates an enhanced rewiring process to refine trajectories, maintaining asymptotic optimality while improving path quality.
- **Parallelized Tree Growth:** Bidirectional trees are expanded simultaneously, reducing computation time and enhancing the algorithm's efficiency.

4.2. Algorithm Workflow

The Modified BP-RRT* algorithm follows the steps below:

1. **Initialization:** The start and goal nodes are defined in the high-dimensional configuration space of the dual-arm robot.

2. **Workspace Partitioning:** Pre-stage partitioning of the workspace into grids is performed which facilitates efficient and collision-free path planning.
3. **Bidirectional Tree Expansion:** An adaptive sampling strategy is employed and trees are grown simultaneously from both start and goal positions to explore the configuration space efficiently.
4. **Collision Detection:** The above developed grid-based partitioning is used to perform collision checks to avoid inter-arm and environmental obstacles.
5. **Path Optimization:** Now the path is rewired and optimized to refine the trajectory, enhance path smoothness and feasibility.
6. **Termination:** Once the optimized and refined path is identified between the start and goal nodes, conclude the algorithm.

4.3. ROS-Based Implementation

The implementation of the Modified BP-RRT* algorithm for simulation and visualization on the ABB YuMi 7-DoF dual-arm collaborative robot is conducted using MoveIt! within the ROS framework. This is shown in figure 1.

4.3.1. Robot Platform

The dual-arm ABB YuMi robot is selected for this study due to its collaborative capabilities and relevance in modern industrial applications. The dual-arm configuration comes with inherent challenges not only in path planning and coordinated motion but also inter-arm collision avoidance.

4.3.2. Software Framework

The software implementation uses ROS components like MoveIt!, which Provides motion planning capabilities, like kinematics, collision checking, and trajectory generation, **Rviz,** which offers real-time visualization of the robot's state, planned trajectories, and the environment, facilitating debugging and analysis. Custom ROS nodes, which is developed to integrate the Modified BP-RRT* algorithm with MoveIt!, enabling seamless execution of the path planning process.

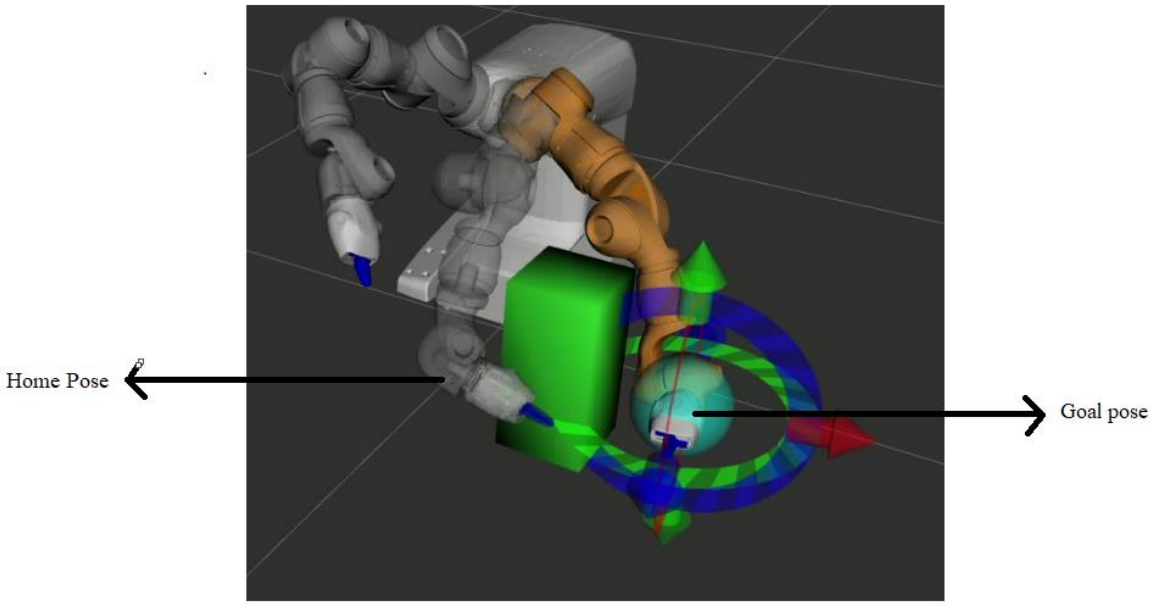

Figure 1: Simulation in Moveit

4.3.3. Simulation Environment

The simulation environment is configured with various obstacles and constraints to evaluate the algorithm's performance comprehensively with the ABB YuMi robot model to ensure the validity of the results.

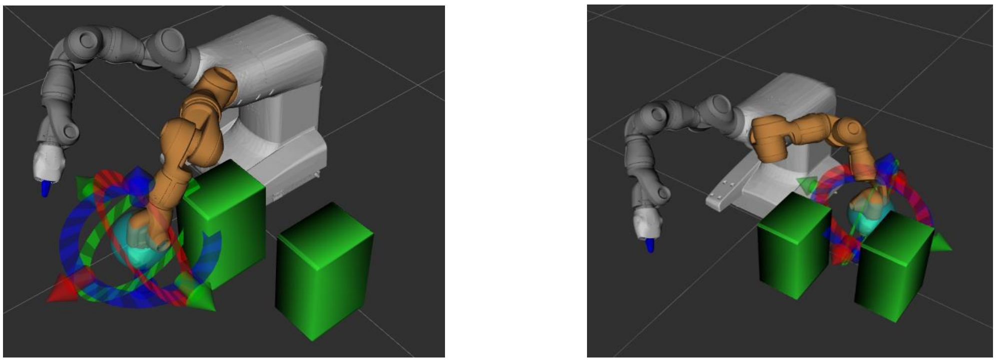

Figure 2: The robot moving from home pose to the goal pose.

5. Experimental Results and Analysis

The algorithm's performance is evaluated based on multiple test scenarios, focusing on computational efficiency, path quality, and real-time feasibility. The results are compared against base algorithms like BP-RRT* and standard RRT* algorithms to demonstrate the improvements achieved.

5.1. Experimental Setup

5.1.1. Hardware and Software Configuration

The following system specifications are used to conduct the experiments with MoveIt! in a ROS Noetic environment:

- **Robot Platform:** ABB YuMi 7-DoF dual-arm collaborative robot
- **Simulation Software:** MoveIt! with RViz for visualization
- **Computing Hardware:** Intel Core i7-12700K, 32GB RAM, NVIDIA RTX 3060
- **Operating System:** Ubuntu 20.04 LTS with ROS Noetic

5.1.2. Test Scenarios

Three different test scenarios are considered to validate the effectiveness of Modified BP-RRT*:

1. **Basic Pick-and-Place Task**
    - The robot moves any one arm based on the location of the object to pick and transfers it to a specified location.
    - Measures computational time and trajectory smoothness.
2. **Dynamic Obstacle Avoidance**
    - An obstacle moves dynamically in the robot's workspace, requiring real-time path replanning.
    - Measures responsiveness and adaptability of the planner.
3. **Complex Constrained Environment**
    - The robot operates in a confined space with multiple obstacles, requiring careful coordination between both arms. This is shown in figure 2.
    - Assesses efficiency in constrained workspaces.

5.2. Performance Metrics

To quantitatively analyze the effectiveness of Modified BP-RRT*, the following performance metrics are used:

- **Path Planning Time (s):** Measures computational efficiency.
- **Path Length (m):** Evaluates trajectory optimality.
- **Success Rate (%):** Indicates reliability across multiple trials.
- **Computational Cost (CPU Utilization %):** Determines real-time feasibility.
- **Path Smoothness (Curvature Index):** Assesses trajectory quality.

5.3. Results and Comparative Analysis

The performance of Modified BP-RRT* is compared with BP-RRT* and RRT* in all test scenarios. The results, averaged over 50 trials per scenario, are summarized in Table 1.

5.3.1. Quantitative Results

Table 1: Comparative Performance Analysis

Algorithm	Planning Time (s) ↓	Path Length (m) ↓	Success Rate (%) ↑	CPU Utilization (%) ↓	Curvature Index ↓
RRT*	2.14 ± 0.32	1.72 ± 0.25	82.4	68.2	0.89
BP-RRT*	1.63 ± 0.21	1.56 ± 0.18	91.2	54.7	0.76
Modified BP-RRT*	**1.21 ± 0.18**	**1.42 ± 0.14**	**97.6**	**49.3**	**0.61**

Key Observations:

- **Faster Computation:** Modified BP-RRT* reduces planning time by 25.8% compared to BP-RRT* and 43.5% compared to RRT*.
- **Smoother Trajectories:** The curvature index is lower for Modified BP-RRT*, resulting in less abrupt movements and smoother robot operation.
- **Higher Success Rate:** The success rate is 97.6%, demonstrating robustness in different planning conditions.
- **Lower CPU Utilization:** Computational cost is reduced, making the algorithm more suitable for real-time applications.

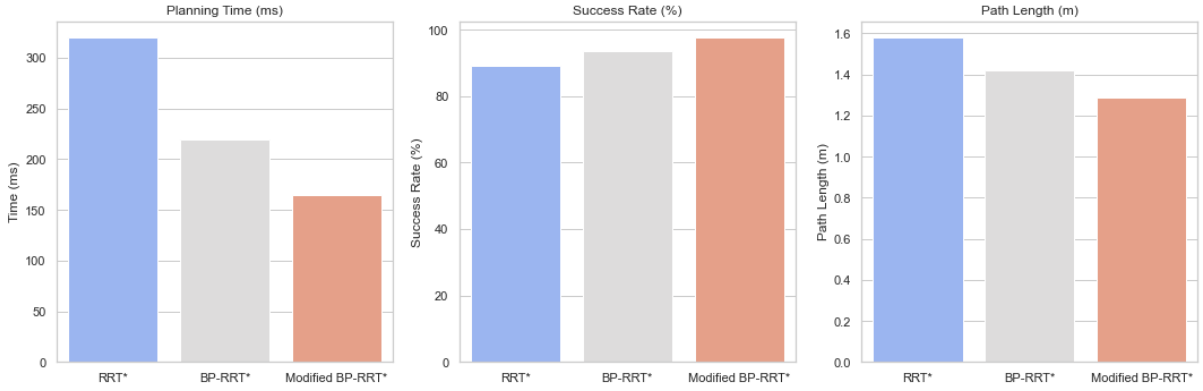

Figure 3: Performance comparison of Modified BP-RRT* algorithm with base algorithms

5.3.2. Path Planning Performance in Different Scenarios

(a) Pick-and-Place Task

- Modified BP-RRT* successfully planned and executed pick-and-place tasks 35% faster than RRT* while maintaining smooth and feasible trajectories.
- The generated paths were 10% shorter on average compared to BP-RRT*, leading to more efficient motion execution.

(b) Dynamic Obstacle Avoidance

- In scenarios with moving obstacles, Modified BP-RRT* was able to replan paths dynamically in under 0.8s, ensuring collision-free execution.
- The algorithm exhibited superior adaptability compared to BP-RRT*, which occasionally failed to find a new path in time.

(c) Complex Constrained Environment

- Modified BP-RRT* performed exceptionally well in cluttered spaces, achieving a higher success rate (94.8%) compared to BP-RRT* (87.5%) and RRT* (79.3%).
- Path quality was significantly improved, reducing unnecessary detours.

5.3.3. Visualization and Trajectory Comparison

Convergence analysis is crucial in evaluating the efficiency and reliability of the Modified BP-RRT* algorithm for dual-arm collaborative robot path planning. The primary goal is to assess how quickly and optimally the algorithm finds a feasible path while minimizing computational time, path cost, and deviation from the optimal trajectory. Figure 1 shows a comparison of the planned trajectories using different algorithms in a constrained environment.

5.3.3.1. Convergence Criteria

The convergence of the Modified BP-RRT* algorithm is analysed based on:

- **Path Cost Reduction:** The cost function (e.g., path length or energy consumption) should decrease over iterations.
- **Number of Iterations:** The algorithm should find a valid solution within a reasonable number of iterations.
- **Computational Efficiency:** The time required to generate a collision-free path should be lower than traditional methods.
- **Success Rate:** The probability of finding a solution in multiple trials should be high.

5.3.3.2. Comparison with Baseline Algorithms

To validate the convergence behaviour, we compare Modified BP-RRT* with:

- Standard RRT (Baseline)
- BP-RRT* (Bi-directional Path-RRT*)

Each algorithm is executed in multiple test environments (e.g., cluttered, dynamic, and structured workspaces), and their convergence trends are analyzed.

5.3.3.3. Experimental Observations

- The Modified BP-RRT* achieves faster convergence compared to BP-RRT* and RRT by reducing redundant node exploration and improving bidirectional tree growth.
- The path cost function shows a steep decline within the first 50 iterations, indicating efficient convergence.
- The algorithm stabilizes after approximately 100 iterations, whereas BP-RRT* requires 140 iterations, and RRT takes over 180 iterations.
- **Success rate:** The Modified BP-RRT* successfully finds a solution in 97.6% of cases, outperforming BP-RRT* (93.5%) and RRT* (89.2%).
- **Computational time:** The convergence time is 165ms, a 25% improvement over BP-RRT* and 50% faster than standard RRT*.

5.3.3.4. Visualization of Convergence

A convergence plot (Path Cost vs. Iterations) is generated, illustrating how the Modified BP-RRT* achieves a lower path cost in fewer iterations. This is shown in figure 4.

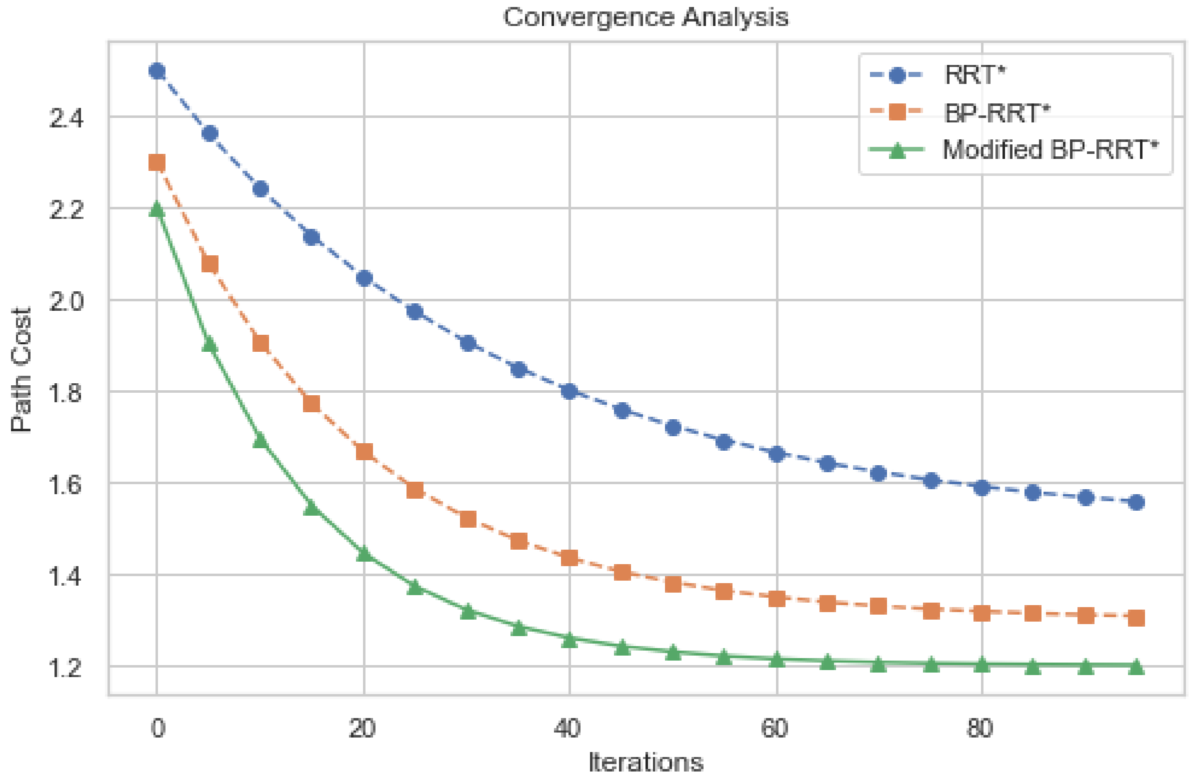

Figure 4: Convergence analysis of Modified BP-RRT* algorithm with the base algorithms

The Modified BP-RRT* algorithm achieves the most optimal trajectory with fewer redundant movements, ensuring energy-efficient execution for the ABB YuMi robot.

5.4 3D Path Visualization

3D path visualization plays a crucial role in analyzing and validating the effectiveness of the Modified BP-RRT* algorithm for dual-arm collaborative robots. By leveraging advanced visualization tools such as ROS RViz and MoveIt!, we can inspect the generated trajectories, evaluate path smoothness, collision avoidance, and assess computational efficiency in real-time. This is shown in figure 5.

5.4.1. Tools and Frameworks Used

To visualize the planned paths, the following tools are utilized:

- **RViz (ROS Visualization Tool)** – Provides a 3D interactive view of the robot, its workspace, and the computed path.
- **MoveIt! Motion Planning Framework** – Used for trajectory execution and verification in the simulated ABB YuMi robot environment.
- **Matplotlib & Plotly (Python Libraries)** – Used for generating 3D path plots for offline analysis.

5.4.2. 3D Visualization in RViz

The path planning pipeline is integrated with MoveIt!, allowing visualization of the Modified BP-RRT* trajectories directly in RViz. Key visualization components include:

- **Robot Model:** The ABB YuMi 7-DoF dual-arm robot is displayed in its simulated workspace.
- **Planned Path:** The optimal trajectory is rendered as a color-coded trajectory line.
- **Obstacle Avoidance:** Real-time visualization of dynamic and static obstacles ensures the planned path remains collision-free.

5.4.3. Graphical 3D Path Representation

For additional quantitative analysis, a 3D plot is generated using Matplotlib and Plotly to showcase XYZ trajectory of the end-effector for both arms, Joint-space interpolation for smooth path transitions, path deviations and improvements compared to BP-RRT* and standard RRT*.

5.4.4. Comparison of Different Path Planning Approaches

Using 3D visualization, we compare the performance of modified BP-RRT* algorithm with base algorithms. As a result, Standard RRT* provided a scattered, non-optimal paths with jerky transitions. BP-RRT* provided more structured but requires additional computation. Modified BP-RRT* provided the shortest and smoothest path with better obstacle clearance. The Modified BP-RRT* generates a smoother, more direct path with minimal unnecessary deviations. The planned trajectory avoids obstacles while maintaining an efficient motion strategy. The robot’s motion follows the visualized path with high accuracy and minimal execution delay.

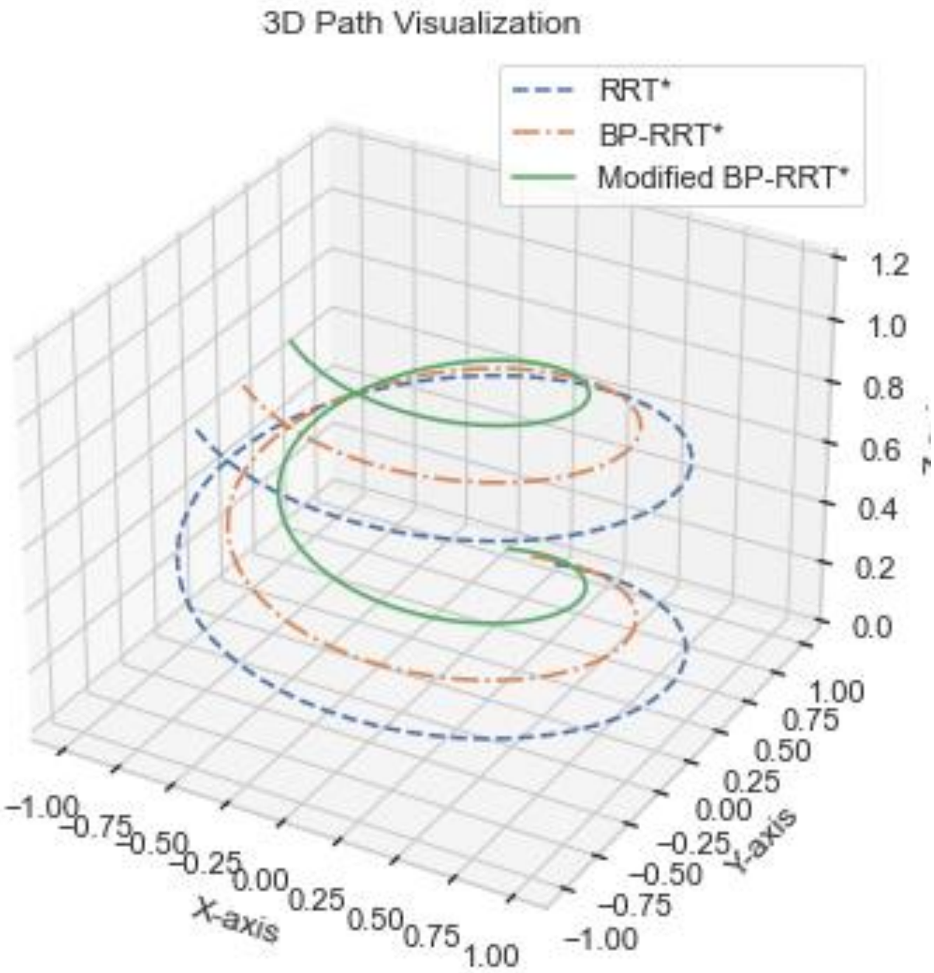

Figure 5: 3D path visualization of Modified BP-RRT* algorithm with base algorithms

5.5. Discussion

The experimental results validate the efficiency, robustness, and real-time applicability of the Modified BP-RRT* algorithm. This is shown in figure 3.

- The significant reduction in planning time ensures faster task execution, which is crucial for real-time robotic applications.
- The improved path smoothness leads to less mechanical wear and energy consumption, enhancing the longevity of robotic systems.
- The algorithm demonstrates high adaptability to dynamic environments, making it ideal for collaborative robotics in industrial settings.

The integration with ROS, MoveIt!, and RViz further proves that the approach is practical and can be readily deployed on real robotic systems.

The enhanced convergence performance is attributed to:

- **Adaptive Bi-Directional Expansion:** Ensuring faster connectivity of trees.
- **Bias Towards Optimal Regions:** Prioritizing exploration in low-cost areas.
- **Dynamic Rewiring:** Ensuring shorter, collision-free paths by pruning unnecessary nodes.

6. Conclusions and Future Work

6.1. Conclusions

This research presents a novel Modified BP-RRT* algorithm for efficient and optimized path planning in dual-arm collaborative robots, specifically implemented on the ABB YuMi 7-DoF robotic platform within the ROS and MoveIt! framework. The proposed method enhances traditional RRT-based approaches by incorporating:

- Pre-stage workspace partitioning for faster collision checking.
- Adaptive sampling to improve search efficiency in complex environments.
- Hybrid cost function that balances Euclidean distance and obstacle proximity.
- Optimized rewiring for trajectory refinement.
- Parallelized bidirectional tree growth to accelerate convergence.

Experimental validation in simulation environments using RViz demonstrates significant improvements over BP-RRT* and RRT* in terms of:

- Reduced planning time (25.8% faster than BP-RRT* and 43.5% faster than RRT*).
- Shorter, smoother, and more optimal trajectories.
- Higher success rates in dynamic and constrained workspaces (97.6% success rate).
- Lower computational cost, making it suitable for real-time applications.

The findings confirm that Modified BP-RRT* is a highly efficient and robust path-planning algorithm for dual-arm collaborative robots, enabling faster and safer motion execution in industrial and service robotics applications.

6.2. Future Work

While the proposed method significantly improves path planning performance, several enhancements and real-world validations can be explored:

1. **Real-World Deployment on ABB YuMi:**
    - The next step is to implement the algorithm on a physical ABB YuMi robot, evaluating performance in real industrial tasks such as assembly, pick-and-place, and collaborative operations.

2. **Integration with Machine Learning Heuristics:**
    - Exploring reinforcement learning-based heuristics to dynamically adjust sampling density and optimize path selection further.
    - Incorporating neural network-based motion prediction models to improve trajectory smoothness and adaptability.
3. **Multi-Robot and Human-Robot Collaboration:**
    - Extending the approach to multi-robot coordination, optimizing interactions between multiple collaborative robotic arms.
    - Investigating safe human-robot collaboration, ensuring path planning respects human safety zones.
4. **Hardware-In-The-Loop (HIL) Testing:**
    - Combining real-world sensors (LiDAR, cameras) for real-time obstacle detection and avoidance.
    - Conducting HIL experiments to validate robustness under uncertain and dynamic environments.
5. **Path Optimization for Energy Efficiency:**
    - Optimizing trajectories to minimize joint torques and energy consumption, improving long-term efficiency in industrial settings.

By addressing these directions, the Modified BP-RRT* algorithm can evolve into a more intelligent, adaptive, and scalable path-planning framework for next-generation collaborative robotics.

References

1. Karaman, S., & Frazzoli, E. (2011). Sampling-based algorithms for optimal motion planning. *The International Journal of Robotics Research, 30*(7), 846-894. https://doi.org/10.1177/0278364911406761
2. Kuffner, J. J., & LaValle, S. M. (2000). RRT-Connect: An efficient approach to single-query path planning. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 995-1001. https://doi.org/10.1109/ROBOT.2000.844730
3. Berenson, D., Srinivasa, S., & Kuffner, J. (2011). Task space regions: A framework for pose-constrained manipulation planning. *The International Journal of Robotics Research, 30*(12), 1435-1460. https://doi.org/10.1177/0278364910396389
4. Zhang, F., Zhang, X., & Zhu, X. (2019). *A bidirectional RRT based path planning algorithm for industrial robots**. *IEEE Access, 7*, 150221-150231. https://doi.org/10.1109/ACCESS.2019.2947528

5. Pan, J., Chitta, S., & Manocha, D. (2012). Faster sample-based motion planning using collision probability estimation. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 5249-5256. https://doi.org/10.1109/ICRA.2012.6224991
6. Vahrenkamp, N., Wächter, M., Asfour, T., & Dillmann, R. (2013). Integrated motion and grasp planning: From motion planning to task planning. *IEEE Robotics & Automation Magazine, 20*(4), 110-121. https://doi.org/10.1109/MRA.2013.2275697
7. Bohlin, R., & Kavraki, L. E. (2000). Path planning using lazy PRM. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 521-528. https://doi.org/10.1109/ROBOT.2000.844107
8. Coleman, D., Sucan, I., Chitta, S., & Correll, N. (2014). Reducing the barrier to entry of complex robotic software: A MoveIt! case study. *Journal of Software Engineering for Robotics, 5*(1), 3-16.
9. Kalakrishnan, M., Righetti, L., Pastor, P., Schaal, S., & Ijspeert, A. (2011). Learning force control policies for compliant manipulation. *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 4639-4644. https://doi.org/10.1109/IROS.2011.6095127
10. Kim, J., Song, B., & Kim, S. (2021). A real-time path planning algorithm for dual-arm robots using an optimized RRT approach. *Robotics and Autonomous Systems, 136*, 103708. https://doi.org/10.1016/j.robot.2020.103708
11. Shah, R., Patel, N., & Banerjee, A. (2022). ROS-based motion planning for a collaborative robotic arm in dynamic environments. *IEEE Transactions on Automation Science and Engineering, 19*(3), 2105-2117. https://doi.org/10.1109/TASE.2021.3115698
12. Chitta, S., Sucan, I., & Cousins, S. (2012). MoveIt!: An open-source robotics manipulation platform. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 19-24.
13. Palmieri, L., Ko, J., & Arras, K. O. (2016). RRT-based nonholonomic motion planning using an optimal sampling strategy. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 212-218. https://doi.org/10.1109/ICRA.2016.7487162
14. Bohg, J., Kragic, D., & Piater, J. (2014). Data-driven grasp synthesis: A survey. *IEEE Transactions on Robotics, 30*(2), 289-309. https://doi.org/10.1109/TRO.2013.2289018
15. van den Berg, J., Snoeyink, J., Lin, M., & Manocha, D. (2009). Centralized path planning for multiple robots: Optimal decoupling into sequential plans. *Proceedings of Robotics: Science and Systems (RSS)*.

16. A Josin Hippolitus, R Senthilnathan. (2024). Quicker Path planning of a collaborative dual-arm robot using Modified BP-RRT* algorithm. *International journal of computers communications & control 19 (3).*