

Rutwik Rajesh Bonde

✉ rutwik.bonde@gmail.com

☎ +1 (774)-701-8374

in [linkedin.com/in/rutwikbonde](https://www.linkedin.com/in/rutwikbonde)

📍 Fort Smith, AR, USA

EDUCATION

Worcester Polytechnic Institute

Masters in Robotics Engineering; GPA: 3.9/4.0

Worcester, MA

Aug'21 - May'23

Vishwakarma Institute of Technology, Pune

Bachelor of Mechanical Engineering; GPA: 9.51/10.00

Pune, India

Aug'17 - May'21

SKILLS

Programming Languages : C++, Python, C

Software Tools : ROS1, ROS2, Matlab, Linux, Docker, Github, Bitbucket

Libraries : Numpy, Pytorch, Matplotlib, Keras, Pandas, Scipy, TensorFlow

Robotics : Motion Planning, Robot Routing, Learning Based Planning, Behavior Trees, Behavior Planning

EXPERIENCE

ArcBest Technologies

Robotics Engineer II - Motion Planning

Arkansas, USA

April'25 - Present

- Leading the architectural redesign and migration of the **global planning navigation stack** from ROS1 to ROS2 to improve scalability, modularity, and real-time performance across multiple autonomous warehouse vehicle platforms.
- Architected and developed an **Auto-Guidance Goal Map Generation System** for a cloud-based robot fleet manager, enabling dynamic generation of navigation node maps for large-scale robot deployments.
- Led the architecture design of a **Multi-Robot Path Planning** system to manage robot fleets in a warehouse environment.

ArcBest Technologies

Robotics Engineer - Path Planning I

Arkansas, USA

June'23 - April'25

- Architected and developed the **autonomous navigation stack in C++** for autonomous warehouse vehicles and forklifts across three robot variants.
- Led the research and development of a **discrete path planner (customized hybrid A* algorithm)** from scratch for a **non-holonomic robot**, integrating **semantic map data** into its evaluation functions.
- Implemented **behavior tree** modules for autonomous **recovery behaviors**, enabling robots to handle blocked paths and navigation failures.
- Optimized **local motion planning algorithms** including **ARA* based replanning** and improved trajectory following using a **Pure Pursuit controller** for stable and smooth vehicle motion.
- **Software deployment experience:** Successfully deployed navigation software at **three customer sites** in two years.
- **Testing and Validation:** Developed automated tests using **Rostests and GTests**, and validated navigation software on robotic hardware (autonomous vehicles/robots).

ArcBest Technologies

Robotics Software Intern

Arkansas, USA

Jan'23 - May'23

- Contributed to **behavior planning and path planning stacks** for autonomous mobile robots.
- Designed and **simulated a virtual warehouse** environment for autonomous mobile robots using **Webots**.

Nanyang Technological University

Research Intern – Computational Geometry

Singapore

Aug'20 - Dec'20

- Researched a **diagonalization based algorithm** to address the **Art Gallery Problem** in computational geometry.
- Implemented algorithmic variations in **Python** to determine optimal guard placement in complex polygonal environments, with and without cavities (e.g., galleries, museums, warehouses).
- Implemented computational geometry structures including **Voronoi diagrams, Delaunay triangulations, and polygon skeletons** in Python.

ACADEMIC PROJECTS

Constructing a Watchmen Route (shortest path) for an Indoor Environment

Jan'22 - April'22

- Developed a **watchman route** to determine the shortest path for scanning an entire indoor environment.
- Utilized scan locations derived from the **Art Gallery Problem** and connected them to form an optimal route.
- Analyzed **RRT, RRT*, Informed RRT, PRM, and Held-Karp** in Python, comparing their performance

Implementation of Deep Reinforcement Learning Algorithms on Racetrack Environment

Sept'22 - Dec'22

- Conducted a comparative study of reinforcement learning **policy gradient algorithms: PPO, DDPG, and A3C**.
- Implemented the algorithms on the racetrack-v0 environment within **OpenAI Gym's Highway Environment**.
- Analyzed and compared the algorithms based on achieved rewards, training time, and implementation complexity.