Sayantani Bhattacharya

Portfolio | Github | Mail ID
IL, Chicago

EDUCATION

Master of Science in Robotics	Northwestern University	2024-2025
Bachelor of Technology in ME	Indian Institute of Technology, ISM Dhanbad	2018-2022

SKILLS

- **Robotics Core:** ROS2, SLAM, Motion Planning & Control, Path planning, Graph-theory, Dynamics, Optimization.
- **Machine Learning:** Classical techniques, CNN, RNN, OpenCV, Filters, Deep Reinforcement Learning, MARL.
- **Programming:** C++, C, Python, Concurrency, gdb, TDD, Linux, Github, Gitlab, Docker, rabbitmq, MongoDB.
- **Libraries:** PyQt, PyTorch, Sympy, numpy, matplotlib, Modern robotics, LP solver, Behavior tree, SLAM-Nav ROS pkgs.
- **Software Tools:** MATLAB, Simulink, Coppelia-Sim (V-Rep), AutoCAD, LABVIEW, Inventor, Unity Gym Env.

PUBLICATIONS

1. Conflict-Free Node-to-Robot Scheduling for Lifelong Operation in a Warehouse with Narrow-Corridor Environment.
   Published in **IEEE Conference on Decision and Control (CDC)** 2023 -Link **Affiliation:** Addverb ***Second author***

Multiple damage detection using point contact excitation and detection method using signal processing techniques.

2. *Published in **MDPI sensors**.* - Journal Link **Affiliation:** Arctic University ***First author***
3. *Presented in **Symposium of Ultrasonic Electric Japan** -* Proceeding Link ***First author***

WORK EXPERIENCES

Addverb — Robotics Engineer *(Full Time)* *Aug' 22 - Jun' 24*

- Formulated heuristic, graph-based & linear programming based solutions for multi-agent path-finding problems.
- Developed the Behaviour tree, Task-Scheduler, Assignment and hardware client's TCP interface module of AMR fleet.
- Developed an end-to-end module using mixed integer linear programming for obstacle avoidance and priority based robot-task assignment for MPV (Mobile picking vehicle) Fleet *(C++)* .
- Built the MPV Fleet simulation configurator, an internal tool for the Sales team, from scratch with the Product Team *(Python, PyQt, Networkiz library)* that reduced the throughput calculation time from 2 days to three hours.
- Developed modules for velocity profiling, simulated the Warehouse Control System's pipeline, Topological feature extraction, Path-follower, Bezier curves, Motion Model, and Behaviour tree in AGV (Automated Guided Vehicle) fleet.
- Devised a Proof of Concept for AGV velocity profiling using Deep Q-Learning. *(Python, PyTorch)*
- Refactored codebase for reduced computation time to solve simulation lag issues for processing 50+ robots in fleet.
- Responsible for client demos for fleet system and remote site support.

IIT Delhi & IIT Dhanbad — Research Internship *July 21' - Apr'22*

- Designed closed-loop force control algorithms for serial and parallel robot manipulators using Sim-Mechanics.
- Implemented forward & inverse dynamics of robot manipulators using Euler-Lagrangian equations and designed Graphical User Interface for dynamics simulations in MATLAB App Designer for teaching and research purposes.

Uit-The Arctic University of Norway — Research Internship *Jan 21' - July 21'*

- Implemented and experimented with Fourier and Wavelet Transform for Denoising time-series received signal data.
- Devised a novel method to differentiate structural damages up to 100 µm difference. *(Labview, MATLAB, Python)*
- Implemented multiple filters for image processing for damage detection of diseased biological samples.

NANO-DEGREES

- **C++ Developer Nanodegree** Certificate
- **Deep Reinforcement Learning Nanodegree** Certificate

PROJECTS

Multi-Agent Exploration on Unitree GOs *(SLAM, ROS2, C++)* *- Ongoing*

- A Fleet of Unitree (GO1 and GO2) capable of decentralised heterogeneous collaborative exploration in outdoor premises.

Multi-Agent Reinforcement Learning in Table-Tennis *(MARL, Unity Gym, PyTorch)*

- Formulated Deep-Deterministic-Policy-Gradient model to collaborate for max game time, while competing to win.

Pen Grasping Robot *(OpenCV, PyTorch)*

- PincherX100 Arm programmed to detect, calibrate, manipulate and grasp a pen using a RealSense depth camera.

Linux System Monitor *(C++, Linux)*

- A system monitor system parallel to htop, that displays the CPU, process IDs, memory usage, etc of a Linux system.