

Bat Nemekhbald

✉ bxtbold@uw.edu [bxtbold.github.io](https://github.com/bxtbold) [bxtbold](https://www.linkedin.com/in/bxtbold) [in bxtbold](https://www.instagram.com/bxtbold)

Objective

Master's student specializing in learning-based planning and control, with experience in MPC, motion planning, and multimodal policies deployed on real manipulators

Education

University of Washington, Seattle *Seattle, Washington*
Master of Science in Mechanical Engineering *Sept 2025 – present*
 ○ Relevant Coursework: Machine Learning, Nonlinear Optimal Control, Intro to Control Theory

Kookmin University *Seoul, South Korea*
Bachelor of Science in Mechanical Engineering *Mar 2018 – Feb 2022*

Research Experience

Research Assistant, Robot Learning Lab @ UW *Seattle, Washington*
University of Washington, Seattle *Nov 2025 – present*
 ○ Built IsaacSim and real Franka environments for manipulation tasks, collected expert demonstrations via space mouse
 ○ Trained, deployed, and evaluated multimodal RL, IRL and BC agents for real manipulation tasks

Undergraduate Researcher, IAS Lab *Irvine, California*
University of California, Irvine *Dec 2019 – Feb 2020*
 ○ Built Gazebo simulation setups for robots to study autonomous navigation in sensor-based mapped environments
 ○ Applied SLAM and path planning methods to enable robots to traverse unseen environments from generated maps

Professional Experience

Robotics Software Engineer, AI Motion Team *Seongnam-si, South Korea*
Plaiif *Jan 2023 – Aug 2025*
 ○ Designed and deployed a real-time 100 Hz MPC-based local planner in C++, integrating differential IK and velocity constraints; coupled with Kalman-filtered state estimation for stable execution of teleoperation and learned policies
 ○ Designed and implemented task-space motion planners in C++ using OMPL, enabling collision-free pick-and-place and general manipulation with robust planning across diverse goal configurations
 ○ Built a high-fidelity MuJoCo-based simulation stack mirroring real robot interfaces, enabling validation of agents and motion planners and reducing deployment-related safety risks and robot/environment damage to 0
 ○ Architected an asynchronous task-motion pipeline decoupling perception, motion planning, and execution, enabling parallel processing across modules and reducing per-cycle latency by 1.5 seconds (40%) on the company benchmark
 ○ Designed and implemented full-stack C++ robotics infrastructure including low-level hardware drivers, trajectory generation modules, and state estimation, enabling stable 100-500 Hz closed-loop control of 6-7 DoF manipulators

Robotics Engineer Intern, Dev Team *Seoul, South Korea*
Cheung Won SEA (CWSFA) *Mar 2022 – Nov 2022*
 ○ Improved mobile robot navigation robustness by integrating SLAM localization with Nav2 and tuning planners and controllers, reducing field localization failures by 50%
 ○ Optimized smart-farm railroad perception by accelerating existing CV pipelines and training/deploying a deep network, reducing CPU usage 30% and achieving 95% detection accuracy
 ○ Developed Qt-based GUIs for monitoring, scenario management, and high-level control of a mobile-manipulator system, reducing operator interaction time by up to 1 minute per cycle
 ○ Designed circuit diagrams and implemented firmware in C for the FADUINO-32TA, integrating the board with ROS to provide reliable I/O and sensor/actuator interfacing

Publications

T. Han, S. Shen, R. Baijal, H. Ravichandiran, **B. Nemekhbald**, K. Huang, S. Jung, B. Boots, “Planning from Observation and Interaction”, *Robotics: Science and Systems (RSS)*, 2026 (**Under Review**)

Skills

Languages: Python, C++, Rust, Bash, Matlab, Latex, Java, JavaScript, SQL

Robotics: ROS/ROS2, Mujoco, IsaacSim, Genesis, Pinocchio, OMPL, MoveIt, Nav2

Math & ML Tools: Gymnasium, Pytorch, Jax, Eigen, OSQP, Numpy

Development Tools: Linux, Git, Docker, Wandb, CMake, FastAPI, React