# Terry (Jiating) Lu

Bay Area Ready | (914) 356-2258 | jiatinglu99@gmail.com | U.S. Permanent Resident
Personal Website: www.terrylu.cloud | LinkedIn.com/in/aterrylu

## EDUCATION

**University of Southern California** — **Los Angeles, CA**

*Bachelor of Science in Computer Science*, GPA: 3.83/4.0 — Aug 2019 - May 2023

*Master of Science in Computer Science - Robotics*, GPA: 3.74/4.0 — Aug 2022 - July 2023

## SKILLS

**Coding Languages:** Python | Java | C/C++ | SQL | TypeScript

**Tools:** AWS (S3, DynamoDB, Lambda, ECS) | Docker | Kubernetes | ROS2 | NVIDIA Isaac Sim | Gazebo | GCP | Unity Game

**Engineering:** Robotics/System Simulation | Software-in-the-Loop Testing | Distributed System | Performance Analysis

## EXPERIENCE

**Amazon Robotics LLC** — **Westborough, MA**

*SDE II (Simulation Infrastructure)* — Dec 2024 - Present

*SDE I* — Oct 2023 - Dec 2024

- Developed infrastructure supporting large-scale distributed robotics warehouse SIL simulation for research teams.
- Spearheaded designs for distributed AWS emulation architecture solving critical simulation scalability bottlenecks, achieving **6x** service startup improvement, **2x** runtime performance, **50%** cost reduction for software-in-the-loop (SIL) simulation, with successful org-wide adoption in 3 major simulation models and across 4 engineering teams.
- Developed Java & Python applications for service discovery and network emulation (DDNS), utilizing RabbitMQ for inter-component messaging, orchestrating simulation deployments in both docker-compose and Kubernetes.
- Led componentization initiative reducing simulation model setup complexity for 5+ engineering teams.

**Amazon Robotics LLC** — **Remote**

*SDE Intern (Simulation Infrastructure)* — May 2022 - Aug 2022

- Developed Checkpoint/Restore feature for Amazon Robotics discrete event simulator, enabling developers to capture and restore multi-day simulation states across hosts, reducing debugging cycles from days to minutes.

**USC Autonomous Coordination of Teams Lab** — **Los Angeles, CA**

*Research Assistant* — Aug 2021 - May 2023

- Researched in safe asynchronous path-planning algorithms (RLSS) for multi-robot coordination systems in 2D & 3D.
- Implemented Voronoi spatial partitioning and ML trajectory scoring infrastructure with FCL/Eigen libraries, enabling continuous 3D collision detection and multi-objective optimization for drone swarms path planning.
- Contributed 1000+ lines of peer-reviewed C++ code and 200+ unit tests to multi-robot aware planning & control stack, following modern C++17/20 practices, RAII principles, and template metaprogramming.

**USC Space Engineering Research Center** — **Los Angeles, CA**

*Software Developer (Ground Station)* — Sep 2019 - Aug 2021

- Developed satellite telemetry pipeline for Magneto CubeSat missions from AX.25 protocol deduplicating to MariaDB database, implementing Python server & Node-RED automation for 24/7 fault-tolerant ground station operations.

## PROJECTS

**Fetcher-ROS: VLA-Integrated Mobile Manipulator Robot (In-Progress)** — 2025

- Experimenting with Vision-Language-Action model integration for Jetson Orin Nano robot with Ackerman chassis and manipulator arm, implementing language-to-action control for dog toy organization with NVIDIA Isaac Sim.

**MeshGraph Cloth Physics Simulation Research** — 2023

- Co-developed Transformer models merging with Google's MeshGraphNets for cloth dynamics simulation, trained on GCP, demonstrating faster GPU parallel training with performance on par with traditional physics methods.

**Pipeline: Automated IDEX 3D Printer System** — 2023

- Led development of automated dual-extruder 3D printer system with Klipper firmware, Slack bot integration, and queue management platform, implementing gantry-based part ejection automation for 24/7 unattended operation.

**Spidy/Spooder: DIY Bionic Spider Robots** — 2017-2020

- Developed two distinct 8-legged bionic robotics platforms: Arduino/C++ implementation (Spidy) with custom radio control, real-time inverse kinematics, and advanced gait algorithms featuring state machines and linear interpolation for smooth bionic locomotion sequences, and Linux/Python implementation (Spooder) on BeagleBone Blue, featuring GPS integration, OpenCV experimentation, and iOS controller app with UDP video streaming.