Caltech
Student-Faculty Programs

Session H Abstracts

Actor-critic reinforcement learning for model predictive control for the Autonomous Indy Car Challenge
Ettienne Caban-Klepac
Mentors: Soon-Jo Chung and Gilchrist X. Johnson

This project implements AC4MPC (Reiter et al., arXiv:2406.03995), in which a reinforcement-learning agent is trained on the exact cost a model predictive controller (MPC) optimizes, and its learned value function (critic) is embedded in the MPC as a terminal cost. The terminal cost summarizes the infinite-horizon consequences of a state, granting the finite-horizon controller long-range foresight, while the trained policy warm-starts the solver. The correctness requirement is that the RL reward equals the negative of the MPC's own stage cost; only then is the critic a valid terminal value function for this controller. To guarantee this, we exposed the production single-track racing MPC's dynamics, stage cost, and 23 of 28 constraint rows to Python via pybind11, so the environment steps the exact dynamics and is scored by the exact cost. A Soft Actor–Critic agent with smooth, compact networks was trained, exported to ONNX with a verified value-scale inversion (relative error 10^{-7}), and integrated as a first-order terminal-cost term in the C++ solver. The full pipeline runs in production on Laguna Seca with zero failed solves. Experiments show the critic faithfully transmits its learned preference, driving closed-loop behaviour across the full range from over-conservative (3.2 m/s) to over-aggressive (solver-breaking), and can be tuned to a stable operating point near the 17.7 m/s nominal baseline. Achieving a net speedup is now a training-scale question rather than an integration one: single-step SAC at 2×10^5 steps has not yet learned feasible high-speed cornering, versus the reference paper's 2×10^6 . the negative of the MPC's own stage cost; only then is the critic a valid terminal value function for this controller. To guarantee this, we exposed the production single-track racing MPC's dynamics, stage cost, and 23 of 28 constraint rows to Python via pybind11, so the environment steps the exact dynamics and is scored by the exact cost. A Soft Actor–Critic agent with smooth, compact networks was trained, exported to ONNX with a verified value-scale inversion (relative error 10^{-7}), and integrated as a first-order terminal-cost term in the C++ solver. The full pipeline runs in production on Laguna Seca with zero failed solves. Experiments show the critic faithfully transmits its learned preference, driving closed-loop behaviour across the full range from over-conservative (3.2 m/s) to over-aggressive (solver-breaking), and can be tuned to a stable operating point near the 17.7 m/s nominal baseline. Achieving a net speedup is now a training-scale question rather than an integration one: single-step SAC at 2×10^5 steps has not yet learned feasible high-speed cornering, versus the reference paper's 2×10^6 .

Disturbance-aware trajectory planning with learned linear-in-latent disturbance dynamics
Erik S. Bryland
Mentors: Soon-Jo Chung and Min Kim

High-precision autonomous maneuvers may fail in the presence of onboard disturbance sources, such as a sloshing liquid payload. In this study, disturbances are predicted as outputs of a latent state-space model whose time-varying coefficients are learned functions of observable quantities. Using this model, we implement an open-loop Monte Carlo Tree Search trajectory planner that incorporates the learned disturbance model. The planner's action sets are generated with spectral analysis. Chance constraints are incorporated through Monte Carlo sampling. We validate the planner on two nonlinear systems with hidden disturbance dynamics, including a system of one-dimensional coupled Duffing oscillators and a two-dimensional system of particles coupled by a nonlinear spring. The performance of the learned model is evaluated against a Koopman-based approach across two tasks: disturbance prediction and downstream planner performance. The planner demonstrates successful disturbance-aware trajectory generation by rerouting around obstacles while accounting for state uncertainty.

Flow Monte Carlo tree search
Jason H. Childs
Mentors: Soon-Jo Chung and Min Kim

This project aims to implement Conditional Flow Matching (CFM) on Monte Carlo Tree Search (MCTS) distributions. MCTS algorithms utilize loops of selection, expansion, simulation, and backpropagation to determine a most optimal set of behaviors to reach a desired outcome. CFM models the dynamics evolving one distribution to the next distribution to normalize data. Its use in this project is to evolve a root state distribution to its successor state distribution. Flow Monte Carle Tree Search (FMCTS) is a hybrid algorithm for more efficient planning where the entirety of the MCTS tree no longer needs to be built. Instead, the tree search is compressed into a neural network. The benefits of this new system design aim to improve speed, improve batch inferences, and create smoother actions within the running system. Preliminary data will be discussed in presentation.

Online adaptation for general humanoid whole-body tracking
Yuyang Wang
Mentors: Soon-Jo Chung and Vrushabh Zinage

Humanoid robots must follow complex whole-body motions while their physical conditions change after deployment. This project develops a general motion controller that can recognize and respond to such changes without retraining its neural network. The controller compares predicted motion with sensor measurements after every control step and uses the difference to estimate the robot's current physical condition. This estimate then guides subsequent actions. Experiments show that the estimated condition captures which motor has weakened and how much strength was lost. Initial tests include physical X2 Ultra experiments with changes in knee strength, together with simulated payload and wind-like external forces. These results suggest that online physical-condition estimation can complement a working general motion tracker and improve its ability to operate beyond controlled laboratory settings. More broadly, interactions such as picking up a box or opening a door also change the forces acting on the robot. The framework could therefore, in principle, treat these interactions in the same way as payloads, wind, motor faults, and other real-world disturbances. Future work will test whether this inferred condition improves closed-loop tracking and recovery during diverse physical tasks.

Perceptive terrain traversal for humanoid robots via route-conditioned reference generation and learned tracking
Minran Wang
Mentors: Soon-Jo Chung and Vrushabh Zinage

Humanoid robots require terrain-aware motion plans that are both geometrically feasible and dynamically trackable, especially when traversal involves contact-rich behaviors such as stepping, jumping, climbing, vaulting, or crawling. This work proposes a three-layer framework for perceptive humanoid terrain traversal that separates route feasibility, whole-body reference generation, and physics-based tracking. First, fused LiDAR and camera observations are converted into a labeled terrain graph whose edges encode traversal modes rather than explicit footsteps or velocity commands. Second, a PARC-style reference generator produces dense future whole-body motion and full-body contact labels conditioned on the selected route, terrain map, and motion history. The generator is trained from a physics-corrected motion library and supervised with route, contact, penetration, and forward kinematic consistency losses. Third, a history-aware reinforcement learning tracker follows the generated future reference in simulation while accounting for contact feasibility and tracking error. This formulation avoids hand-specifying footstep timing, enables multiple traversal modes to be represented through route labels, and provides a deployment pathway in which a compact generator proposes diverse candidate references while a learned feasibility model and tracker select dynamically executable motions.

Learning based resilient autonomy for robotic systems under actuation and perception faults
Baaqer M. Farhat
Mentors: Soon-Jo Chung and Mahdi Taheri

Robots operating in real environments must maintain stable behavior when perception becomes unreliable or task conditions change. This project studies methods that combine multimodal learning, reinforcement learning, and motion retargeting to improve robotic autonomy. The first part of the work uses a multimodal Transformer to process histories of state estimates, control inputs, and diagnostic signals. The learned representation updates model predictive control cost parameters and residual dynamics during operation. This approach adapts the controller instead of replacing it, which preserves the planning and safety structure of model predictive control. We test the method on a spacecraft simulator in the laboratory under controlled perception faults to evaluate its ability to maintain stable control when perception degrades.

The second part of the work applies robot learning to locomotion and whole body interaction tasks. Reinforcement learning policies are trained using about 4096 parallel simulation environments and produce joint position targets at 50 Hz. A locomotion policy is transferred from simulation to physical hardware and follows commanded forward, lateral, and turning velocities. Human motion data from a 52 joint model is also retargeted to a 31 degree of freedom robot and used as a reference for box manipulation and crawling policies.

Disturbance-compensated reinforcement learning control for trajectory tracking
Junze Zhang
Mentor: Soon-Jo Chung

Unmodeled forces, including actuator mismatches, environmental disturbances, and surface resistance, can often amplify a controller's trajectory tracking errors during experimental operation. This project thus develops a novel reinforcement learning-based adaptive control algorithm that improves trajectory tracking by learning and correcting for these differences. The method consists largely of a model-based tracking controller and adds a learning-based disturbance correction. It first uses reinforcement learning (RL) to learn a shared disturbance representation offline from randomized trajectories and continuously varying force and moment disturbances, which improves upon the supervised learning models by learning compensation that accounts for and actuator limits and delayed effects without requiring exact disturbance models and expert demonstrations. During operation, a small set of coefficients is updated online to adapt this representation to the current disturbance. The method was then developed and evaluated in simulation before being tested on the Caltech Multi-Spacecraft Testbed for Autonomy Research (M-STAR). Performance was compared to both model-based methods, such as pure PD controller, and learning-based methods, such as pure RL controller and controller using RL to learn the disturbance term, on 1,000 unseen trajectories. The addition of an online coefficient adaptation produced a further modest improvement in simulation. Learning only a disturbance correction improved all four tracking measurements, while using RL to learn the total control command directly produced the largest errors. Initial hardware testing demonstrated an average position error of approximately 0.0698 meters and an average heading error near zero (0.0301 radians) while following a closed trajectory, for both constant and varying desired headings. These results indicate that reinforcement learning is more effective as a correction to an existing controller than as its complete replacement, and that combining offline learning with online adaptation is the most promising approach for further hardware testing and eventual multi-spacecraft fault recovery.

Trajectory-conditioned reinforcement learning for multimodal control of M4 robots
Warren C. Zhou
Mentors: Mory Gharib and Matthew Anderson

The Multi-Modal Mobility Morphobot (M4) is a family of reconfigurable robots whose appendages can be repurposed for aerial and terrestrial locomotion. Controlling these robots is challenging because their dynamics, actuator geometry, contact conditions, and number of controllable degrees of freedom vary across configurations and operating modes. Previous reinforcement learning work demonstrated morpholanding for fixed point-to-point objectives, but did not provide explicit trajectory tracking or unified control across ground and aerial operation. This project develops a trajectory-conditioned reinforcement learning framework for two contrasting M4 variants: ATMO, the lowest degree-of-freedom M4, and Stork, the highest degree-of-freedom M4. For ATMO, a single policy coordinates four operating modes: driving, takeoff, flight, and landing. The policy commands individual rotor thrusts, morphological reconfiguration, and wheel motion while tracking time-varying position, velocity, and heading references. The same training framework is used to train Stork policies despite its substantially larger observation and action spaces. Both the ATMO and Stork policies perform well in Isaac Lab across their targeted behaviors, demonstrating that the training approach can accommodate M4 platforms with widely different levels of mechanical complexity. The ATMO policy has also demonstrated functionality in a PX4-Gazebo simulation with actuator dynamics and interfaces that differ from the training environment. An ATMO-specific hardware deployment pipeline has been developed using PX4, a Jetson companion computer, and OptiTrack localization. Initial physical testing will focus on ATMO, after which the Stork policy may also be deployed, while the shared reinforcement learning framework can be used to train policies for additional M4 variants.

Reducing blade-wake interaction noise from upstream propeller shroud geometries on M4 Phoenix
Anshi Paul
Mentors: Mory Gharib, Matthew Anderson, and Reza M. Nemovi

The Multi-Modal Mobility Morphobot (M4) Phoenix is a morphing autonomous robot that combines aerial and ground locomotion using shrouded propellers surrounded by treads to traverse environments where terrain, access, and mission requirements are variable. However, its high acoustic signature may limit operation near people. This project investigates whether redesigning the structural spokes shrouds upstream of the propeller can reduce noise while preserving the geometry required for ground mobility. A single-propeller test rig was designed and fabricated to allow interchangeable shroud panels. Controlled design families vary spoke count while holding total blockage approximately constant, and include systematic comparisons of tapered, rounded, and various spiral spoke geometries intended to spread out and reduce interactions. Acoustic measurements are recorded with a fixed measurement microphone and sound pressure level meter at consistent motor operating points, and frequency spectra compare overall sound level, blade-passing tones, their harmonics, and prominent geometry-dependent tones. The resulting comparisons will provide a better understanding of the noise sources on M4 Phoenix and from upstream propeller shrouding in general, identify quieter spoke configurations, and guide a quieter shroud design for integration into the full M4 Phoenix robot.