Research Theme: Supervised Autonomy

• **Autonomous Navigation in Unstructured Environments**
  • How can we enable robots to plan their own dynamically-feasible motions to successfully navigate in unstructured environments?

• **Task-Level Autonomy**
  • How can we make robots more capable to complete complex tasks on their own?

• **Human-Robot Collaboration**
  • How can we alleviate the cognitive workload placed on human supervisors working with a team of robots?
  • How can the robots interact with their human supervisors to provide feedback and incorporate new objectives?
Research Theme: Supervised Autonomy

• **Autonomous Navigation in Unstructured Environments**
  - How can we enable robots to plan their own dynamically-feasible motions to successfully navigate in unstructured environments?

• **Task-Level Autonomy**
  - How can we make robots more capable to complete complex tasks on their own?

• **Human-Robot Collaboration**
  - How can we alleviate the cognitive workload placed on human supervisors working with a team of robots?
  - How can the robots interact with their human supervisors to provide feedback and incorporate new objectives?
Overview

• Research Theme: Supervised Autonomy

• **Navigation in Unstructured Environments**

• Task-Level Autonomy

• Human-Robot Collaboration

• Highlighted Applications: Marine Robotics, Aerial Robotics

• Other Areas: Medical Robotics, Robot Manipulation, Cyber-Physical Systems

• Discussion
Motion Planning in Unstructured Environments

- Fundamental for autonomous robotics
- Numerous applications
  - exploration
  - transportation
  - navigation
  - search-and-rescue
  - video games
  - medical robotics
  - ...

Autonomous Navigation

compute a collision-free and dynamically-feasible trajectory from the initial location to the goal
Motion Planning Poses Hard Problems

- High-dimensional continuous state spaces
- Obstacle-rich and unstructured environments
- Feasible motions constrained by underlying dynamics
  - Often nonlinear, nonholonomic, and high-dimensional
- PSPACE-complete: geometry, no dynamics
- Undecidable: with dynamics
Driving Research Questions

• How can we develop motion planners that are generally applicable?

• How can we achieve planning efficiency even for robots with nonlinear dynamics operating in unstructured, obstacle-rich environments?

• How can we improve the solution quality?

• What formal guarantees can we provide?
Sampling-based Motion Planning

Expand tree whose branches correspond to collision-free and dynamically-feasible motions
repeat until solution or runtime limit is reached:
• select state from which to expand the tree
• select target position
• generate trajectory from selected state toward target
• add the collision-free portion as new branch to the tree

$MP = (s_{init}, \text{GOAL, SUCCESSOR})$ : search problem
• accounts for obstacles and dynamics
Expand tree whose branches correspond to collision-free and dynamically-feasible motions

repeat until solution or runtime limit is reached:

• select state from which to expand the tree
• select target position
• generate trajectory from selected state toward target
• add the collision-free portion as new branch to the tree

**MP** = \((s_{\text{init}}, \text{GOAL}, \text{SUCCESSOR})\): search problem

• accounts for obstacles and dynamics
Sampling-based Motion Planning

Expand tree whose branches correspond to collision-free and dynamically-feasible motions

repeat until solution or runtime limit is reached:

• select state from which to expand the tree
• select target position
• generate trajectory from selected state toward target
• add the collision-free portion as new branch to the tree

\[
\text{MP} = (s_{\text{init}}, \text{GOAL}, \text{SUCCESSOR}) : \text{search problem}
\]

• accounts for obstacles and dynamics
Sampling-based Motion Planning

Expand tree whose branches correspond to collision-free and dynamically-feasible motions

repeat until solution or runtime limit is reached:

• select state from which to expand the tree
• select target position
• generate trajectory from selected state toward target
• add the collision-free portion as new branch to the tree

\[ MP = (s_{init}, \text{GOAL}, \text{SUCCESSOR}) : \text{search problem} \]

• accounts for obstacles and dynamics
Sampling-based Motion Planning

Expand tree whose branches correspond to collision-free and dynamically-feasible motions
repeat until solution or runtime limit is reached:
- select state from which to expand the tree
- select target position
- generate trajectory from selected state toward target
- add the collision-free portion as new branch to the tree

\[ MP = (s_{\text{init}}, \text{GOAL, SUCCESSOR}) : \text{search problem} \]
- accounts for obstacles and dynamics
Sampling-based Motion Planning

On difficult problems:
- Expansion frequently gets stuck
- Progress slows down
- Exploration guided by limited information
- Difficult to discover new promising directions

State-of-the-art motion planners have difficulty solving these challenging problems
Our approach
Framework

- Pioneered framework to treat motion planning as a *search* problem in a *hybrid* space composed of *discrete* and *continuous* components.

### AI
- High-level reasoning over discrete abstractions
- Provide simplified planning layer
- Guide search in the continuous space

### Sampling-Based Motion Planning
- Probabilistic sampling to selectively explore the space of feasible motions
- Expand tree by adding collision-free and dynamically-feasible trajectories as branches

Autonomous Navigation
Framework

• Pioneered framework to treat motion planning as a search problem in a hybrid space composed of discrete and continuous components

AI

• High-level reasoning over discrete abstractions
• Provide simplified planning layer
• Guide search in the continuous space

Sampling-Based Motion Planning

• Probabilistic sampling to selectively explore the space of feasible motions
• Expand tree by adding collision-free and dynamically-feasible trajectories as branches

Simplified planning layer

• relaxed problem: no dynamics, point robot
Framework

• Pioneered framework to treat motion planning as a search problem in a hybrid space composed of discrete and continuous components

AI

• High-level reasoning over discrete abstractions
• Provide simplified planning layer
• Guide search in the continuous space

Sampling-Based Motion Planning

• Probabilistic sampling to selectively explore the space of feasible motions
• Expand tree by adding collision-free and dynamically-feasible trajectories as branches

Simplified planning layer

• relaxed problem: no dynamics, point robot
• decomposition, adjacency graph
Framework

• Pioneered framework to treat motion planning as a *search* problem in a *hybrid* space composed of *discrete* and *continuous* components.

**AI**

• High-level reasoning over discrete abstractions
• Provide simplified planning layer
• Guide search in the continuous space

**Sampling-Based Motion Planning**

• Probabilistic sampling to selectively explore the space of feasible motions
• Expand tree by adding collision-free and dynamically-feasible trajectories as branches

**Simplified planning layer**

• relaxed problem: no dynamics, point robot
• decomposition, adjacency graph
• shortest-path from each region to goal
Pioneered framework to treat motion planning as a search problem in a hybrid space composed of discrete and continuous components.

**AI**
- High-level reasoning over discrete abstractions
- Provide simplified planning layer
- Guide search in the continuous space

**Sampling-Based Motion Planning**
- Probabilistic sampling to selectively explore the space of feasible motions
- Expand tree by adding collision-free and dynamically-feasible trajectories as branches

iterate until solution found or runtime limit is reached:
- select non-empty region with maximum weight
- expand motion tree along shortest path
- update weights based on progress made
Framework: Enhancements/Improvements

abstractions via subdivisions
Framework: Enhancements/Improvements

abstractions via roadmaps
Framework: Enhancements/Improvements

Autonomous Navigation

direct path superfacades
Framework: Enhancements/Improvements

Autonomous Navigation

clearance-driven motion planning
Framework: Enhancements/Improvements

Leveraging ML

- Train model to predict problem difficulty
- Use predictions to guide motion tree expansion
Motion Planning in Unknown Environments

Autonomous Navigation
Summary of Motion-Planning Capabilities

- Ground, aerial, and marine robots
- High-dimensional nonlinear dynamics
- Differential equations or physics-based engines
- Unstructured, obstacle-rich, and even unknown environments
- Any cost/risk metric
- Real-time planning
- Speedups of one to two orders of magnitude over related work

Selected Publications for Motion Planning with Dynamics:

- IEEE Intl Conf on Automation Science and Engineering (CASE), 2023
- IEEE Intl Conf on Intelligent Robots and Systems (ICRA), 2022
- IEEE Intl Conf on Automation Science and Engineering (CASE), 2022
- Robotica, 2018
- IEEE Robotics and Automation Letters (RAL), 2017
- IEEE Trans on Robotics (TRO), 2015
- Springer LNCS Towards Autonomous Robotic Systems, 2015 (Best Student Paper, my undergrad student)
Multi-Goal Motion Planning

Physical TSP
- visit each goal
Multi-Goal Motion Planning

Generalized Physical TSP
- visit at least one goal from each group
Multi-Goal Motion Planning

Physical TSP with Limited Energy and Recharging Stations
Multi-Goal Motion Planning

Physical TSP with Time Windows
- visit each goal within specified timeframe
Multi-Goal Motion Planning

Physical TSP with Time Windows, Pickups, Deliveries, and Limited-Load Capacity
Summary of Multi-Goal Motion Planning

- Physical Traveling Salesman Problem (TSP)
- Generalized TSP
- Physical TSP with Limited Energy and Recharging Stations
- Physical TSP with Time Windows, Pickups, Deliveries, and Limited Load Capacity

[IEEE Intl Conf on Automation Science and Engineering (CASE), 2020]
[Springer LNCS Advances in AI, 2019]
[IEEE/RSJ Intl Conf on Intelligent Robots and Systems (IROS), 2018]
[Springer LNCS Towards Autonomous Robotic Systems, 2017]
Multi-Robot Motion Planning

- Computationally efficient, scaling up to tens of robots in 1-3s planning time

- IEEE Intl Conf on Robotics and Automation (ICRA), 2021
- IEEE Robotics and Automation Letters, 2019
- J Artificial Intelligence Research, 2018
- Intl Joint Conf on AI, 2018
- Intl Conf on Planning and Scheduling, 2017 (Best Robotics Paper)
- IEEE Trans on Automated Science and Engineering (TASE), 2015
Research Theme: Supervised Autonomy

• Autonomous Navigation in Unstructured Environments
  • How can we enable robots to plan their own dynamically-feasible motions to successfully navigate in unstructured environments?

• Task-Level Autonomy
  • How can we make robots more capable to complete complex tasks on their own?

• Human-Machine Collaboration
  • How can we alleviate the cognitive workload placed on human supervisors working with a team of robots?
  • How can the robots interact with their human supervisors to provide feedback and incorporate new objectives?
Missions via High-Level Languages

Natural Language

- propositions and predicates to express general statements: “safe; reached area; measurement taken; object picked up”
- verbs to express actions relevant to the overall mission: “move to; inspect; avoid; track; pick up; release”
- logical connectives to combine multiple objectives: “and; or; not; if; if and only if”
- temporal connectives to express objectives along time spans: “next; always; eventually; until; time intervals”
- preconditions and postconditions to express effects of actions
- sentences formed by combining propositions/predicates/verbs with logical/temporal connectives

\[(\text{always safe}) \text{ and (eventually inspect areas A1, A2, \ldots, An}) \text{ and (if damage detected then take pictures or attempt repair)) and (next return to the base)}\]

\[(\text{for each package } p_i: \text{ pickup } p_i \text{ whose weight is } w_i \text{ from location } P_i \text{ within time } [t_{\text{start}}, t_{\text{end}}] \text{ and deliver to location } D_i \text{ within time } [T_{\text{start}}, T_{\text{end}}]) \text{ and do not exceed max weight capacity } C \text{ and reduce time to complete deliveries and distance traveled}\]

Formal Models

- Regular Languages (RL)
- Linear Temporal Logic (LTL)
- Signal Temporal Logic (STL)
- Planning-Domain Definition Languages (PDDL)
Combined Task/Mission and Motion Planning

- Framework tightly couples AI, Motion Planning, and Control
- Enables robots to complete high-level missions on their own
- Missions given by Regular Languages, Linear Temporal Logic, PDDL

**Complex Tasks**
- Regular Languages
- Linear Temporal Logic
- AI Languages

**Complex Systems**
- High-dimensional
- Nonlinear dynamics
- Ground, aerial, marine robots

**Discrete Planning Layer**
- High-Level Reasoning
- AI, Logic

**Continuous Planning Layer**
- Sampling-based Motion Planning
- Motion Control

- Task specification
- World model
- Robot model
- Planning
- Collision-free and dynamically-feasible trajectory that enables the robot to accomplish its task

- Task-Level Autonomy

**Task: visit goals 1-10 before 11-20**

**Task: pick and place each object X in goal X; when carrying an object go only through empty rooms**

[IEEE Trans on Automation Science and Engineering (TASE), 2022]
[IEEE Intl Conf on Automation Science and Engineering (CASE), 2021]
[IEEE Intl Conf on Robotics and Automation (ICRA), 2021]
[Robotica, 2017]
[Journal of Experimental and Theoretical Artificial Intelligence, 2016]
[AI Communications, 2015]
[IEEE/RSJ Intl Conf on Intelligent Robots and Systems (IROS), 2013]
Autonomous Surface and Underwater Vehicles

Pursuing research toward long-endurance missions

Adaptive mission and motion planning to enhance autonomy of marine vehicles

[collaborations with NRL, Australian Defence Science and Technology Group]
Complex Missions via Linear Temporal Logic

Enable AUV to avoid collisions and complete complex missions given by Linear Temporal Logic

Mission examples:
- Sequencing (goals in order)
- Coverage (goals in any order)
- Partial ordering (some goals before others)
- Group coverage (all goals in a group before moving on to the next group)

Ocean Server Iver-2
- 55-inch length
- 5.8-inch diameter

Field Testing at NRL Chesapeake Bay Detachment
- Mission duration ≈ 2 hours

[collaboration with NRL]

Autonomous Data Collection with Limited Time

**Bluefin-21 Heavyweight AUV**
- 18-hour endurance at 2.5 kts
- 314-inch length
- 21-inch diameter
- DVL & INS navigation

**Field Testing at Boston Harbor**
- Upper bound on mission duration is 42 minutes
- Each circle represents a target location
- Reward associated with each target
- Radius represents the distance required to sample data from target

- Enable AUV to reach several locations within a given time bound
- Each region of interest has a reward associated with it
- When not all regions can be reached, maximize overall reward

[IEEE Robotics and Automation Letters, 2016]
[collaboration with NRL]
Simultaneous Survey and Inspection with Communication Constraints for Teamed Autonomous Marine Vehicles

- Survey vehicle
  - moves along pre-planned path
  - uses its on-board sensors to detect objects of interest
  - acoustically communicates locations and rewards to Inspection vehicle
- Inspection vehicle seeks to inspect detected objects, seeking to maximize total reward
- Survey and inspection vehicles must be in communication range at all times

---

Execution time: 22800.8s/25578.5s
Collected reward: 2213/2397/2397 · 92.3% : 92.3%

[USV speed: 1m/s] [AUV desired speed: 1.50m/s] [communication range: 130m]
Simultaneous Survey and Inspection with Communication Constraints for Teamed Autonomous Marine Vehicles

Field Experiments:
- Bantry Bay within Sydney Harbor
- 2 Remus 100 AUVs

- How can the planner ensure that the Inspection vehicle always stays within the communication range of the Survey vehicle?
- How can the planner determine the order in which to inspect goals as they are detected and reported by the Survey vehicle in real time so as to increase the sum of the rewards?
- How do we plan efficiently to support real-time applications?

[collaboration with NRL and Australian Defence Science and Technology Group]

[IEEE/RSJ Intl Conf on Intelligent Robots and Systems (IROS), 2023]
[IEEE Trans Automation Science and Engineering (TASE), 2022]
[IEEE Intl Conf Automation Science and Engineering (TASE), 2021 (Finalist Best Application Paper)]
[IEEE Intl Conf Robotics and Automation (ICRA), 2021]
Autonomous Inspection and Persistent Surveillance

Search-and-find

Camera-based Inspection of 3D Structures

Persistent Surveillance of Risk-Sensitive Areas by a Team of UAVs

Inspection of Nonflat Terrains via Microwave Remote Sensing (MRS)

Tracking using MRS

[Springer LNCS Towards Autonomous Robotic Systems, 2017]

[USNC-URSI National Radio Science, 2017]

[IEEE Intl Conf on Automated Science and Eng, 2021]
Research Theme: Supervised Autonomy

• Autonomous Navigation in Unstructured Environments
  • How can we enable robots to plan their own dynamically-feasible motions to successfully navigate in unstructured environments?

• Task-Level Autonomy
  • How can we make robots more capable to complete complex tasks on their own?

• Human-Machine Collaboration
  • How can we alleviate the cognitive workload placed on human supervisors working with a team of robots?
  • How can the robots interact with their human supervisors to provide feedback and incorporate new objectives?
• Interaction with human supervisors: Essential to effectively handle challenges that arise when operating in complex domains

• The robot team will adapt its plans based on difficulties that it encounters or new information that it gathers

• Human supervisors will also be able to modify the overall mission or specify additional tasks based on the feedback information provided by the robot team
Enhancing Autonomy and Providing Assistance in Human-Machine Cooperative Tasks

Ground/Aerial/Marine Robotics

Sensor-based Manipulation

Medical Robotics

Cyber-Physical Systems

Collision-Avoidance Protocol

Protocol modeled as a hybrid system

Planning framework automatically discovers if protocol guarantees safety