Topological Motion Planning

Subhrajit Bhattacharya
University of Pennsylvania
Topological Motion Planning?

- Motion planning *(planning a set of actions that achieves a specific objective)* is fundamental to autonomy:
  - An efficient, versatile and effective method in motion planning is **graph search based motion planning**.
- Topology attempts to re-incorporate those **richer information** that a discrete graph representations (a 1-dimensional entity) fails to capture.
- Many specifications of goal/obective in motion planning can be formally described using the language of topology.
- Locally computed topological quantities can be used *(reliably integrated)* to provide global guarantees.
- Topological methods do not rely on precise metric information *(robust to errors)*.

Graph search-based planning
[A*: Hart, et al.; D* Stenz et al; RRT: Lavalle]

Simplicial complex
[Derenick, et al.; Ghrist et al;]

Topological Trajectory Planning
[Bhattacharya, et al.]

“go to the left of an obstacle” vs. “go to the right of an obstacle”
Outline

• Topological Trajectory Planning and its Applications
• Dimensionality Reduction using Topological Abstraction
• Sensor Coverage of Unknown, GPS-denied Environments using Robot Swarms
• Simplicial Search Algorithms
Outline

- Topological Trajectory Planning and its Applications
- Dimensionality Reduction using Topological Abstraction
- Sensor Coverage of Unknown, GPS-denied Environments using Robot Swarms
- Simplicial Search Algorithms
Topological Trajectory Planning:

Motivation

Optimal Trajectory Planning for Systems with Cable:

Deep Horizon Oil Spill Cleanup Operation:

Tethered robot:

Tasks requiring Topological Reasoning and Multi-agent exploration:

Multi-agents search/ exploration in a partially-known environment:

[Mellinger, Michael, Kumar, IJRR 2012]
We would like to be able to:
1. Make distinction between the different topological classes of trajectories.
2. Exploit that information for optimal trajectory planning in different topological classes.
3. Apply that to solving real problems in robotics.
Related Work

- **Cell-decomposition (e.g., Voronoi decomposition, Delaunay triangulation) and Semi-algebraic Description of Environments:** [Demyen, Buro, AAAI, 2006; Hershberger, Snoeyink, JCGTA, 1991; Grigoriev, Slissenko, ISSAC, 1998; Schmitzberger, Bouchet, Dufaut, Wolf, Husson, ICIRS, 2002.]
  - Often construction is difficult / expensive, especially for a environment presented as an occupancy-grid.
  - If not carefully constructed (e.g., arbitrary triangulation), the classification may not be one-to-one.
  - While possible to classify given trajectories, the representation is not best suited for search-based optimal path planning.

- **Simplicial Complex Representation and Persistence Homology:** [Pokorny, Hawasly, Ramamoorthy, RSS, 2014;]
  - Requires only a simplicial description of the free space (without an embedding)
  - Well-suited for classifying given trajectories in different **homology** classes.
  - Recent developments in computational **cohomology** on simplicial complex allows construction of topological invariants [Pokorny, et. al, RSS, 2015].

- **Topological Invariant (can be used in conjunction with graph search):**
  - Simple construction
  - Ideal for graph search-based optimal motion planning for finding optimal paths in different homology classes.
  - Suitable for both **homology** and **homotopy** path planning.
Homotopy and Homology

Homotopy:
\[ p_1 \sim p_2 \sim p_3 \sim p_4 \]

Homology:
\[ p_1 \sim p_2 \sim p_3 \sim p_4 \]

\( p_1 \) and \( p_2 \) belong to same homotopy class
\[ \Rightarrow \text{ they belong to same homology class.} \]

Converse is not necessarily true!

Homology invariant, \( H(p_i) \):
\[
H(\tau) = \int_{\tau} \begin{pmatrix} \xi_1 \\ \xi_2 \end{pmatrix} \\
\text{(H-signature of } \tau) \\
\text{(e.g., } \xi_1 = d\theta_i) \\
\]

Key concept: Find a set of linearly independent closed, non-exact differential 1-forms which forms a basis for the de Rham cohomology group,
\[ H^1_{dR}(R^N \to O) \]: \( \xi_1, \xi_2, \ldots \in \text{Ker}(d^1) \)
\[ \notin \text{Img}(d^0) \]

Homotopy invariant, \( h(p_i) \), in 2-dimensions:

Key concept: Words constructed by tracing a trajectory and inserting letters based on rays crossed.
\[
h(\tau) = " b \ b^{-1} \ b \ a \ a^{-1} " = " b " \]
\[
h(\tau_1) = " b \ a^{-1} " \neq h(\tau_2) = " a^{-1} \ b " \]

[Tovar, Cohen, LaValle, WAFR, 2008]
[Narayanan, Vernaza, Likhachev, LaValle, ICRA, 2013]
[Bhattacharya, Kim, Heidarsson, Sukhatme, Kumar. IJRR, 2014]
[Bhattacharya, Ghrist, IMAMR, 2015.]

\( \tau_1 \) and \( \tau_2 \) homologous, but not homotopic.
Use in Graph Search

**H-augmented graph construction** (Illustration in cylindrically discretized 2-D env.):

\[ H(v_{\text{start}} \to v_2) = H(v_{\text{start}} \to v_1) + H(e) \]

Original graph, \( G \)

Vertices: \( v \)

\[ H\text{-augmented graph, } G_H \]

Vertices: \( \{v, [\theta]\} \)

In 2D (homology & homotopy):

In 3D: \( X-Y-Z \text{ config space:} \)

\( X-Y\text{-Time config space:} \)

In 4D: (homology) \( X-Y-Z\text{-Time config space:} \)
Application 1

Topological Exploration

Group of robot splitting based on the available topological classes in the environment:

ROS simulation of topological exploration of an unknown environment using 8 robots.

Single-robot experiment
- Scarab mobile robot platform (differential drive, laser range sensors)
- Visual odometry localization module

[Kim, Bhattacharya, Ghrist, Kumar. IROS, 2013]
Application 2

Human-Robot Collaborative Topological Exploration for Search and Rescue Mission

- Heterogeneous team of humans and robots need to explore an environment for search & rescue missions.
- Human(s) chooses trajectories at their discretion.
- Robots need to adapt and choose complementary topological classes to maximize exploration / clearing.

A decentralized implementation in ROS (simulation):

Robots choose complementary paths to humans'.

Robots adapt to unpredictable human behavior

[Govindarajan, Bhattacharya, Kumar, DARS'14, Best paper award nomination!]
Application 3: Object Separation Using Cable

Motivation:
Field experiment in collaboration with USC:

[Bhattacharya, Kim, Heidarsson, Sukhatme, Kumar. IJRR, 2014]

Problem definition:

Basic idea:
1. Mathematically describe a “separating configuration” (identified by its homology class).
2. Find optimal trajectories in the right homotopy classes leading to a separating configuration.

Dynamic sim.:

Field experiment in collaboration with USC.:
Application 4: Planning for a Tethered Robot

**Problem definition:**

```
4 3 5
2 6
```

**Dynamic simulation:**

- **t = 0.505000**
- **t = 30.300000**
- **t = 60.600000**
- **t = 90.900000**
- **t = 120.695000**

**Method:** Perform search in h-augmented graph

**Results:**
- Initial config.
  - 300x200 env.
  - Target: 350 disc. units
  - Base: 450 disc. units

[Kim, Bhattacharya, Kumar, ICRA'14]
Other Applications

  [Park, Karumanchi, Iagnemma, T-RO, 2015.]

- Conflict minimization in multi-robot motion planning.
  [Kimmel, Bekris, 2012]

- Smooth optimal trajectory planning in different topological classes using QP and MIQP frameworks
  [Kim, Sreenath, Bhattacharya, Kumar, CDC, 2012; Kim, Sreenath, Bhattacharya, Kumar, ARK, 2012; Sikang Liu, Watterson, Bhattacharya, Kumar (under preparation).]
A Persistent Homology Approach to Topological Path Planning in Uncertainties

Probability map, $P$

How to do path planning given a probability map?
- threshold?
  At what value?

We consider $U^\varepsilon = \{ q \in W \mid P(q) \leq \varepsilon \}$ for different value of $\varepsilon$, and how the homology classes of trajectories join and split.

[Bhattacharya, Ghrist, Kumar. T-RO, 31(3), 578-590, 2014]
\[ \mathbb{Z}_2 \text{ coefficients (homology)} \]

**Advantage in graph search-based planning:**

\[ [\tau_1] \sim [\tau_3] \sim [\tau_2] \]

Eliminate trajectories that “loop” around obstacles.

Homotopy Invariants in 3D

Recall: Homotopy invariants in 2D

\[ h(\tau_1) = "b \ b^{-1} \ b \ a \ a^{-1}" = "b" \]

[\textit{Bhattacharya, Ghrist, IMAMR, 2015}]

Trivial loops can have non-empty words: \( h(\gamma) = "u_1^{-1} u_2 u_3^{-1}" \)

Need to map these words to identity (empty word)

- Quotient group \( \pi_1(X) \simeq \pi_1(X_0) * \pi_1(X_1) * \cdots * \pi_1(X_n) / N \)
Outline

• Topological Trajectory Planning and its Applications
• Dimensionality Reduction using Topological Abstraction
• Sensor Coverage of Unknown, GPS-denied Environments using Robot Swarms
• Simplicial Search Algorithms
Abstraction / Dimensionality Reduction using Topology

What we have done so far is topological abstraction
– We reduced the infinite dimensional & continuous *path/curve space* into a finite-dimensional, searchable space
– Involved classification of (the high-dimensional configuration space) paths based on homotopy/homology classes (topological invariants).

**Objectives**
- Take end-effector to a desired target location
- Optimization of trajectory of end-effector (e.g., its length)
  *(We do not care where the rest of the arm is, as long as it does not intersect an obstacle!)*

**Challenges:** (High-dimensional configuration space)
- Randomized search in configuration space gives suboptimal solution.
- Planning trajectory in end-effector space does not guarantee traversability / algorithmic completeness.
- Not sufficient to consider only the homotopy classes of arm configuration in the end-effector space (e.g., 4-bar linkage violating Grashof criterion).
Low-dimensional Sub-sampling of Configuration Space

Schematic:

- Construct the Reeb graph of the FK function (given a fixed end-effector pos. sample a configuration from each connected component of preimage)
- Find path from the start configuration to a preimage of the goal end-effector pos. in the Reeb graph. *(Guarantee: A path in the Reeb graph exists if and only if a path exists in the configuration space between the start configuration and the pre-image of goal end-effector positions (and there is a natural projection map).)*

Approach:
Construct an explicit description of the Reeb graph of the FK function as k-tuple of inverse kinematics (IK) functions.
*(closed-form solution for planar arm in absence of obstacles)*

Contamination state remains the same (maps to the same abstract state)

Topological invariant:
Connected components of the evader space and their contamination state (can be formulated as zero-th (co)homology of a sheaf).

Sheaf theory allows us to place/attach additional data on a topological space.

[Ramaithitima (Tee), Srivastava, Bhattacharya, Speranzon, Kumar, (under preparation)]
Outline

• Topological Trajectory Planning and its Applications
• Dimensionality Reduction using Topological Abstraction
• Sensor Coverage of Unknown, GPS-denied Environments using Robot Swarms
• Simplicial Search Algorithms
Topological Representation

An n-simplex for every (n+1)-tuple of sensors that are pair-wise neighbors.

The Rips Complex

- Requires only local connectivity data for construction.
  - Sensor model:

  ![Diagram of local connectivity and noisy bearing measurement]

  - Gives a faithful representation of sensor coverage

  ![Diagram of Rips Complex]

- Can be used to detect holes in sensor coverage.
- Very limited work in literature on actually controlling the mobile sensors.

[Rips Complex has been used to detect holes, but little research in controlling mobile sensors to attain coverage.]

[Derenick, Kumar, Jadbabaie, ICRA, 2010]
[de Silva, Ghrist, IJRR, 2006]
Overall Algorithm

Step 1: Identify a robot on the frontier subcomplex (closest to source in hop counts) for next deployment.

Step 2: Find a new location outside the frontier (in the local coordinate of the frontier robot), and identify shortest path through graph for robot deployment.

Step 3: “Push” robots along the path using bearing-only controller using other robots as landmarks.

Visual Homing (Bearing-only) Control for Robot Navigation

- Control velocity computed using
  - Bearing to landmarks (neighbors),
  - Desired home/goal location in local coordinates,
  - Landmarks can be moving.

\[ c_i(x) = \beta_{g_i}^T \beta_i(x), \varphi = \sum_{i=1}^{N} \varphi_i, \varphi_i = r_i f(c_i) \]
\[ u = -\text{grad}_x \varphi \]

Simulation and Experiment

ROS Simulation:
• ROS + Stage simulation – running on a 8-core Intel processor
• Non-holonomic robots
• Single source (at the entrance to the environment), unending supply of robots.

Experiment with Real-Virtual Robots:
• Heterogeneous team of live (green) and virtual (red) robots.
• New paradigm in demonstrating swarm algorithms using limited number of live robots.
• Feedback loop between simulated robots, live robots and simulated version of live robots for coherent working of real & virtual robots.

Sensor Coverage of Unknown Environments By Robots Swarms Using Limited Local Sensing

Rattanachai Ramaithitima, Mickey Whitzer
Subhrajit Bhattacharya, and Vijay Kumar

GRASP Lab, University of Pennsylvania

Experiment with Heterogeneous team of Live and Virtual Robots

We used a Scarab robot as a physical platforms and Stage robot simulator for the virtual robots to demonstrate the performance of the proposed algorithm on the real robot experiment.

[Ramaithitima (Tee), Whitzer (Mickey), Bhattacharya, Kumar, ICRA 2015]
Applications

In unknown, GPS-denied environments, with limited sensing:

• Coarse Topological Mapping

• Topological Localization and Capture of Evaders

• Persistent Surveillance

• Establishment of Landmarks for Topological Landmark-based Navigation.

Outline

- Topological Trajectory Planning and its Applications
- Dimensionality Reduction using Topological Abstraction
- Sensor Coverage of Unknown, GPS-denied Environments using Robot Swarms
- Simplicial Search Algorithms
Search Algorithm For Simplicial Complexes

Dijkstra's search:

*Paths are restricted to graph*

Consequence:

Vertex “expansion”:

*Included higher-dimensional simplices*

S* search:

*Paths can lie in a simplicial complex (a Rips complex of the given graph)*

Path reconstruction:

*Under progress.*
Conclusion

Topology helps capture richer (and meaningful/relevant) information about a system/configuration space (using topological invariants and representations), while keeping the problem tractable. Purely graph-based approaches alone fail to achieve this.

Thank you!
Questions?
Acknowledgements

Collaborators

Past and Present Supervisors
Prof. Vijay Kumar (U. Penn)
Prof. Robert Ghrist (U. Penn)
Prof. Maxim Likhachev (CMU)

Graduate/Undergraduate Students and Postdoctoral Collaborators at U. Penn
Dr. David Lipsky
Dr. Mihail Pivtoraiko
Dr. Koushil Sreenath (presently at CMU)
Soonkyum Kim
Vijay Govindarajan
Rattanachai Ramaithitima (Tee)
Mickey Whitzer
Luis Guerrero
Denise Wong
Edward Steager

Collaborators at USC
Prof. Gaurav Sukhatme
Hordur K. Heidarsson

Collaborators at Drexel University
Prof. M. Ani Hsieh
Dhanushka Kularatne

Funding (Past and Present)

ONR:
Antidote MURI grant# N00014-09-1-1031,
Grant# N00014-08-1-0696
Grant# N00014-07-1-0829
Grant# N00014-09-1-1052;
Grant# N00014-14-1-0510

NSF grant# N00014IIP-0742304.
AFOSR grant# FA9550-10-1-0567
ARL grant# W911NF-10-2-0016