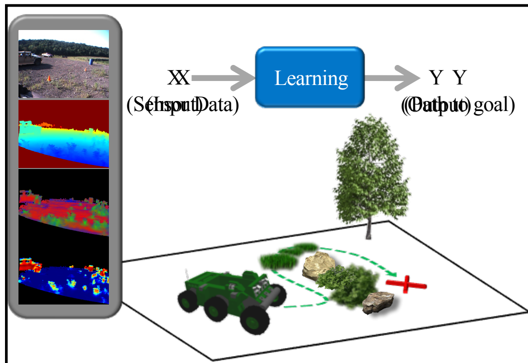
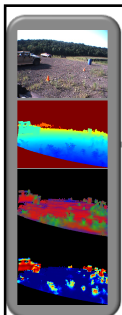
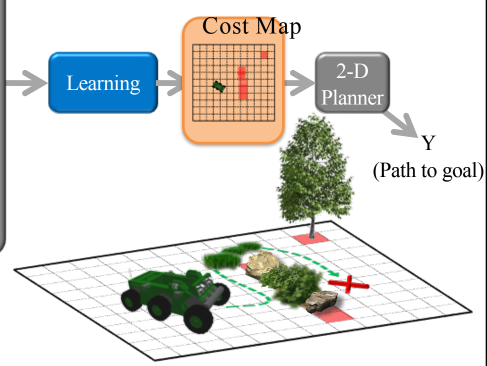
CSE 571
Inverse Optimal Control
(Inverse Reinforcement Learning)

Many slides by Drew Bagnell
Carnegie Mellon University



Optimal Control Solution

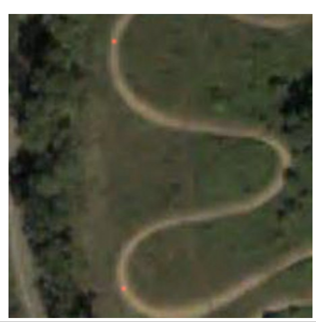




Mode 1: Training example

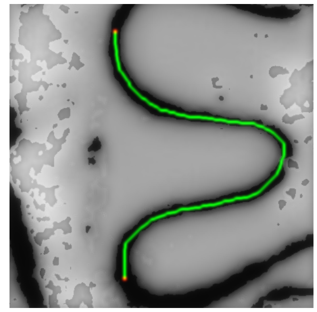
Mode 1: Training example

Mode 1: Learned behavior



Mode 1: Learned behavior

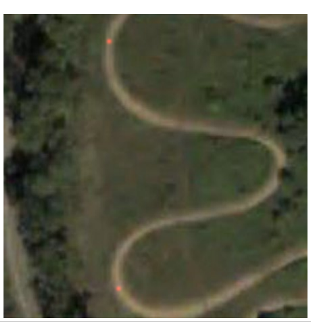
Mode 1: Learned cost map



Mode 2: Training example

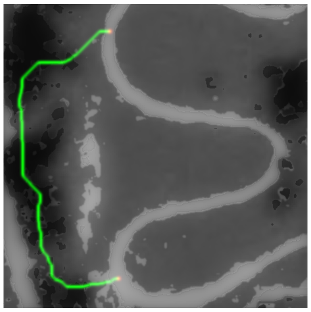
Mode 2: Training example

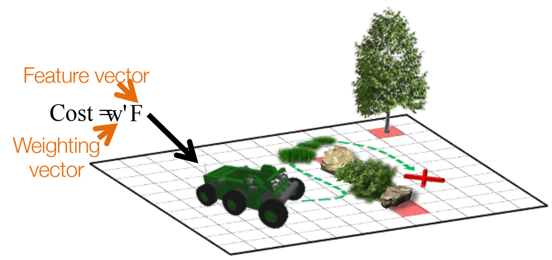
Mode 2: Learned behavior



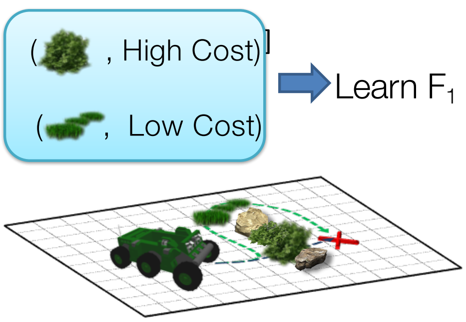
Mode 2: Learned behavior

Mode 2: Learned cost map

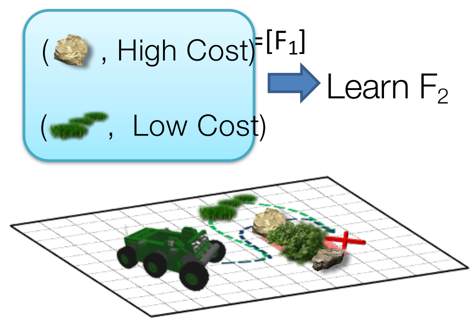




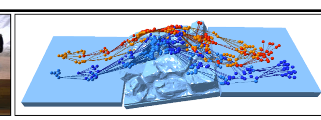
Ratliff, Bagnell, Zinkevich 2005
Ratliff, Bradley, Bagnell, Chestnutt, NIPS 2006
Silver, Bagnell, Stentz, RSS 2008

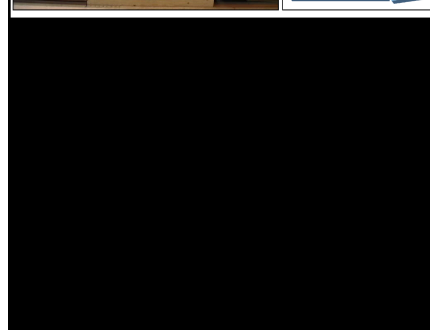


Ratliff, Bagnell, Zinkevich, ICML 2006
Ratliff, Bradley, Bagnell, Chestnutt, NIPS 2006
Silver, Bagnell, Stentz, RSS 2008



Ratliff, Bagnell, Zinkevich, ICML 2006
Ratliff, Bradley, Bagnell, Chestnutt, NIPS 2006
Silver, Bagnell, Stentz, RSS 2008

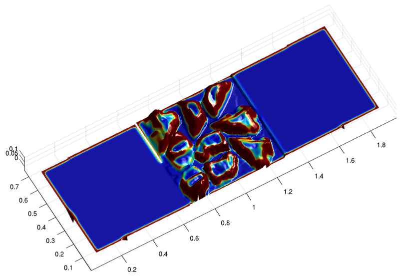




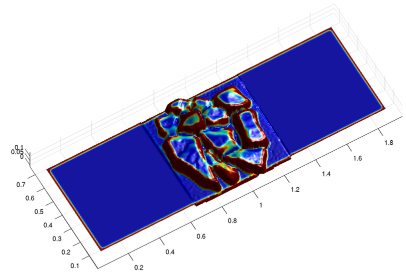
Ratliff, Bradley, Chesnutt, Bagnell 06

Zucker, Ratliff, Stolle, Chesnutt, Bagnell, Atkeson, Kuffner 09

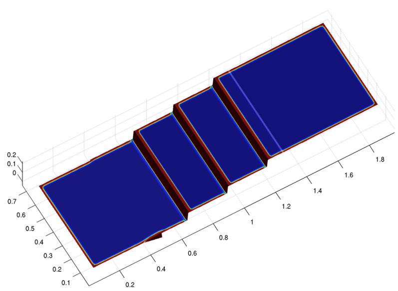
Learned Cost Function Examples



Learned Cost Function Examples

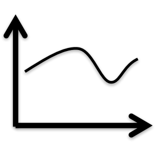


Learned Cost Function Examples

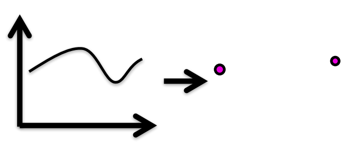


Learning Manipulation Preferences

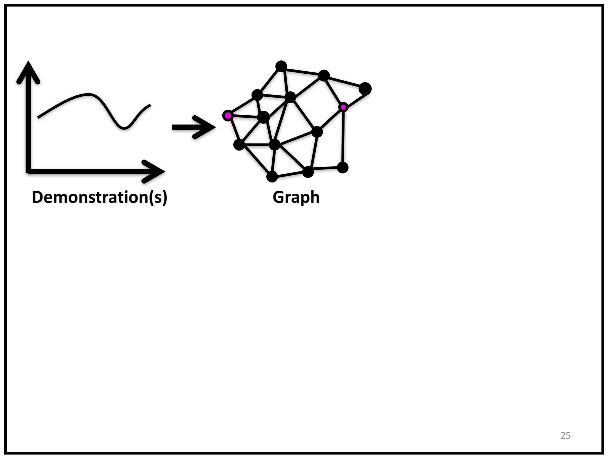
- ***Input:*** Human demonstrations of preferred behavior (e.g., moving a cup of water upright without spilling)
- ***Output:*** Learned cost function that results in trajectories satisfying user preferences

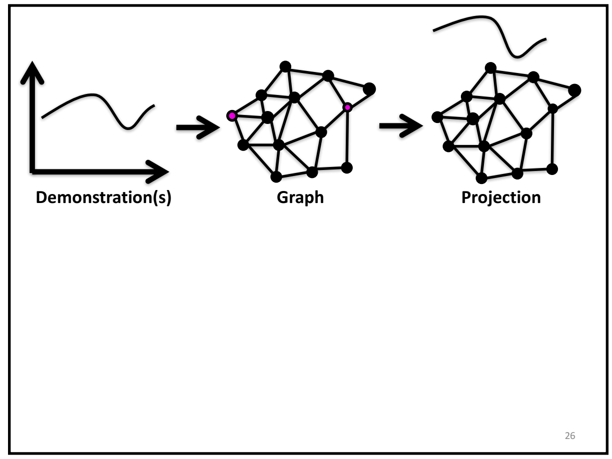


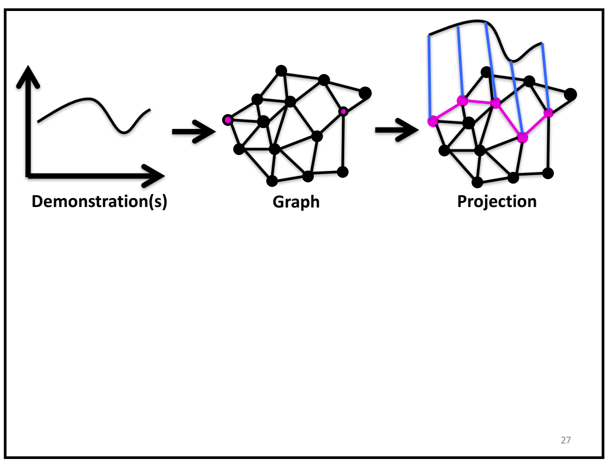
Demonstration(s)

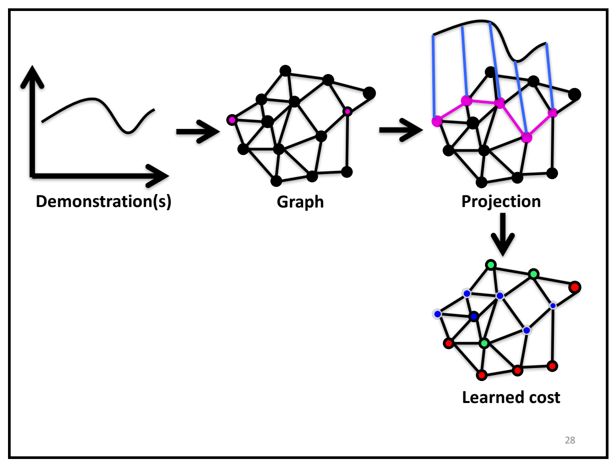


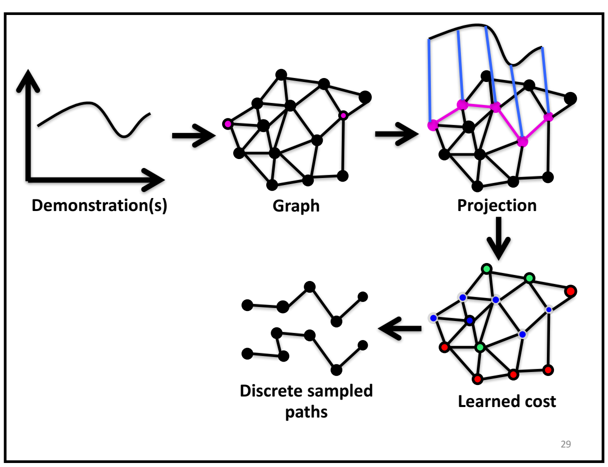
Demonstration(s) **Graph**

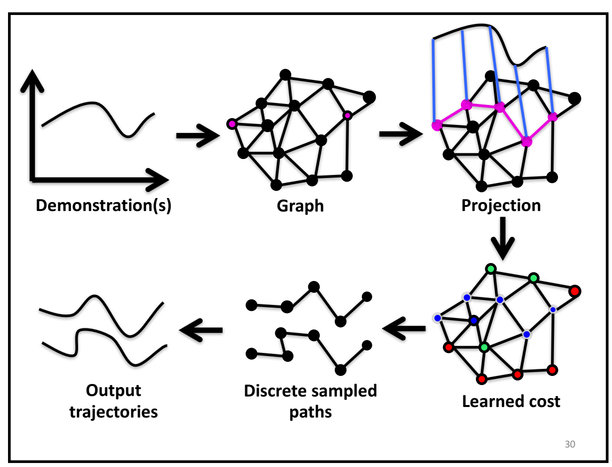

Demonstration(s)
Graph
25

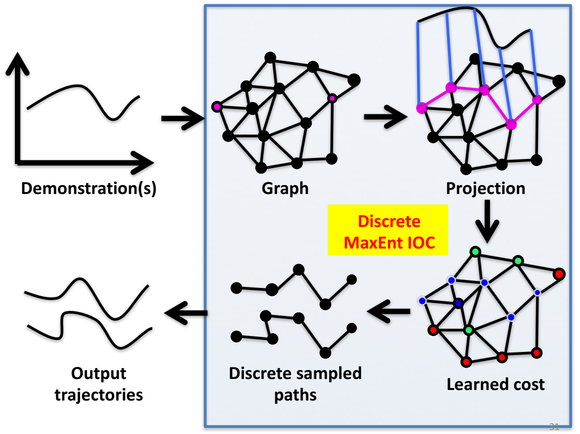

Demonstration(s)
Graph
Projection
26

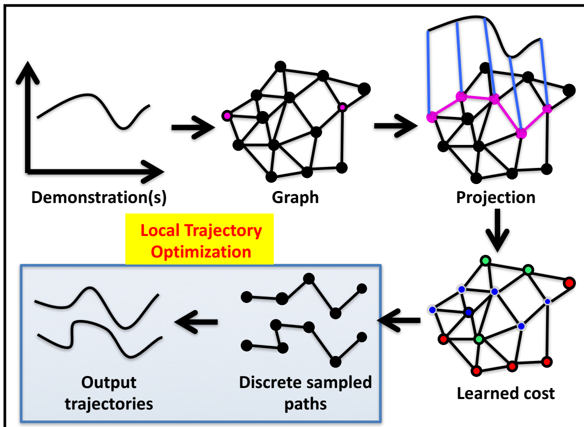

Demonstration(s)
Graph
Projection
27


Demonstration(s)
Graph
Projection
Learned cost
28


Demonstration(s)
Graph
Projection
Discrete sampled paths
Learned cost
29

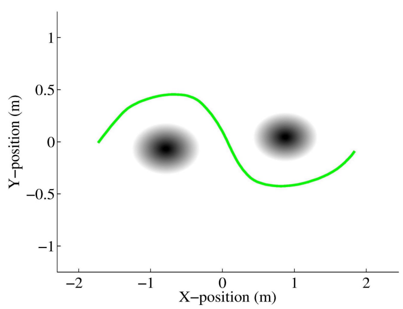

Demonstration(s)
Graph
Projection
Output trajectories
Discrete sampled paths
Learned cost
30





32

2D obstacle avoidance task

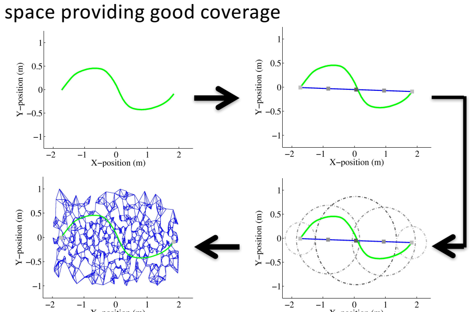


2D state: (x,y)

33

Graph generation

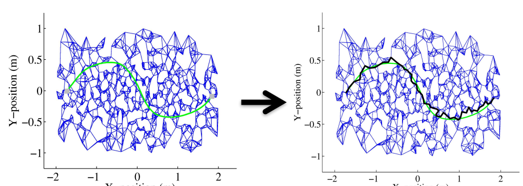
- ***Goal:*** Construct a graph in the robot's configuration space providing good coverage



34

Projection

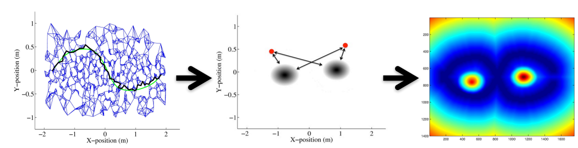
- ***Goal:*** Project the continuous demonstration onto the graph, resulting in a discrete graph path
- Use a **modified Dijkstra's** algorithm minimizing sum of:
  - Length of discrete path (Euclidean)
  - Distance to continuous demonstration



35

Learning the cost function

- ***Goal:*** Given projected demonstrations, learn the cost function
- Learn feature weights (θ *) using **softened value iteration** on the discrete graph (MaxEnt IOC - Ziebart et al., 2008)
  - State dependent features (eg: Distance to obstacles)



36

Experimental Results

Setup

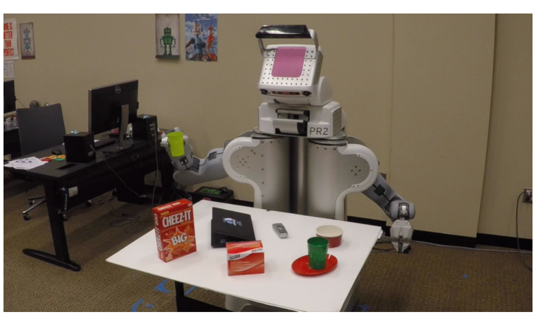
- ***Binary*** state-dependent features (~95)
  - Histograms of distances to objects
  - Histograms of end-effector orientation
  - Object specific features (electronic vs non-electronic)
  - Approach direction w.r.t goal
- **Comparison**:
  - Human demonstrations
  - Obstacle avoidance planner (CHOMP)
  - Locally optimal IOC approach (similar to Max-Margin planning, Ratliff et. al., 2007)

Laptop task: Demonstration

(Not part of training set)

Laptop task: LTO + Discrete graph path

Laptop task: LTO + Smooth random path



Statistics for Laptop task

Method	*% Points in collision*	*End-Effector normal deviation (deg.)*	*% Points above laptop*
Human Demonstration	2.7	7.4	2.1
Obstacle avoidance planner	12.9	18.2	17.3
Coarse, discrete graph sample	12.8	9.9	11.1
Local Trajectory Optimizer + Graph samples	4.0	5.3	1.2
Local Trajectory Optimizer + Random path	4.5	5.5	3.1