Matthew Robards
ANU and NICTA
12/09/2010

Willow Garage

Reinforcement Learning Methods for Robotics

- Reinforcement learning has been seldom applied beyond small “toy” applications
- Attempted control using reinforcement learning with *some* positive results
- Created a value prediction stack which can be used for deciding between policies

Willow Garage

Introduction To Reinforcement Learning

- Given at time t
  - States $s_t \in S$
  - Actions $a_t \in A$
  - Rewards $r_t \in \mathbb{R}$
- Assume Markov property:

$$P(s_{t+1}, a_{t+1}, r_{t+1} | s_t, a_t, r_t, \ldots, s_0, a_0, r_0) = P(s_{t+1}, a_{t+1}, r_{t+1} | s_t, a_t, r_t)$$

- State value function: $V : S \to \mathbb{R}$
- State-action value function: $Q : S \times A \to \mathbb{R}$
- Policy $p : S \to A$

Willow Garage

State(-Action) Value Functions

- Given rudimentary reward signal, eg. $r \in \{0, 1\}$
- Predict long term discounted reward:

$$V(s_t) = \sum_{i=0}^{\infty} c^i r_{t+i} \qquad \text{state value}$$

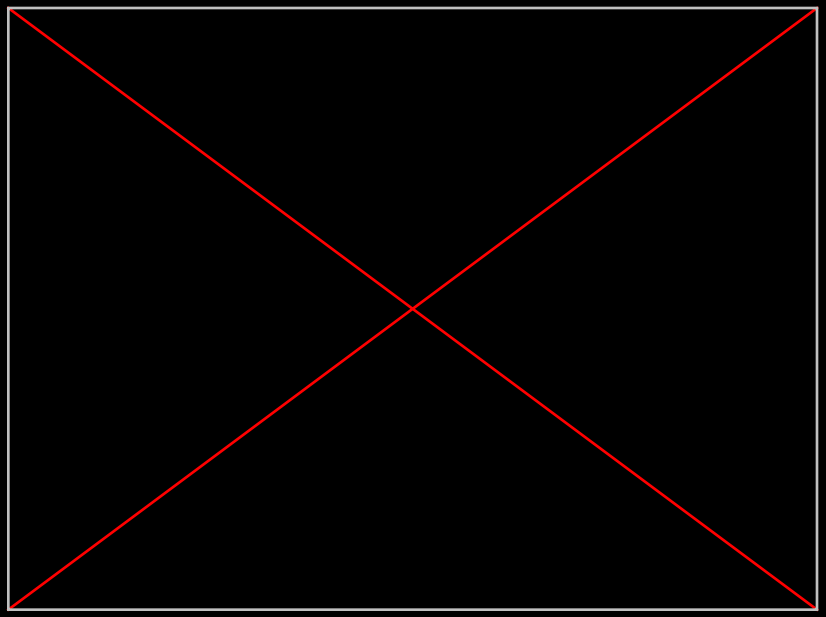
$$Q(s_t, a_t) = \sum_{i=0}^{\infty} c^i r_{t+i} \qquad \text{state-action value}$$

$$0 \leq c \leq 1$$

- Reward signal can be very simple and still learn a very complex value function

State Value Functions

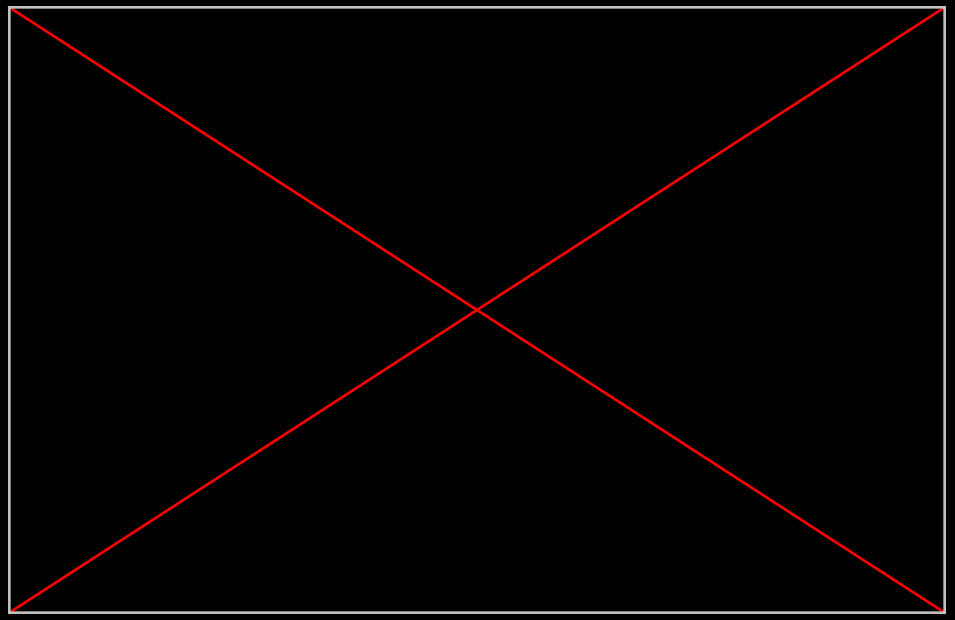
- eg. $r_t = 1$ for $t = 1, \square, 10$
  $c = 0.9$



State-Action Value Functions

- eg. $r_t = \begin{matrix} 1 & \text{if a} = 0 \\ 2 & \text{if a} = 1 \end{matrix}$ for $t = 1, \square, 10$

$c = 0.9$



Willow Garage

Learning For Control

Willow Garage

Learning A Policy

- Given enough experience, try to predict a "good"

$$Q(s,a) \quad \forall \; s \in S$$

- Why?
  - This gives us actions, allowing us to choose a policy. eg.

$$p(s) = \text{argmax}_a Q(s,a)$$

Willow Garage

Learning To Drive To Goal

$S = \{w, x, \dot{y}, \dot{x}\}$

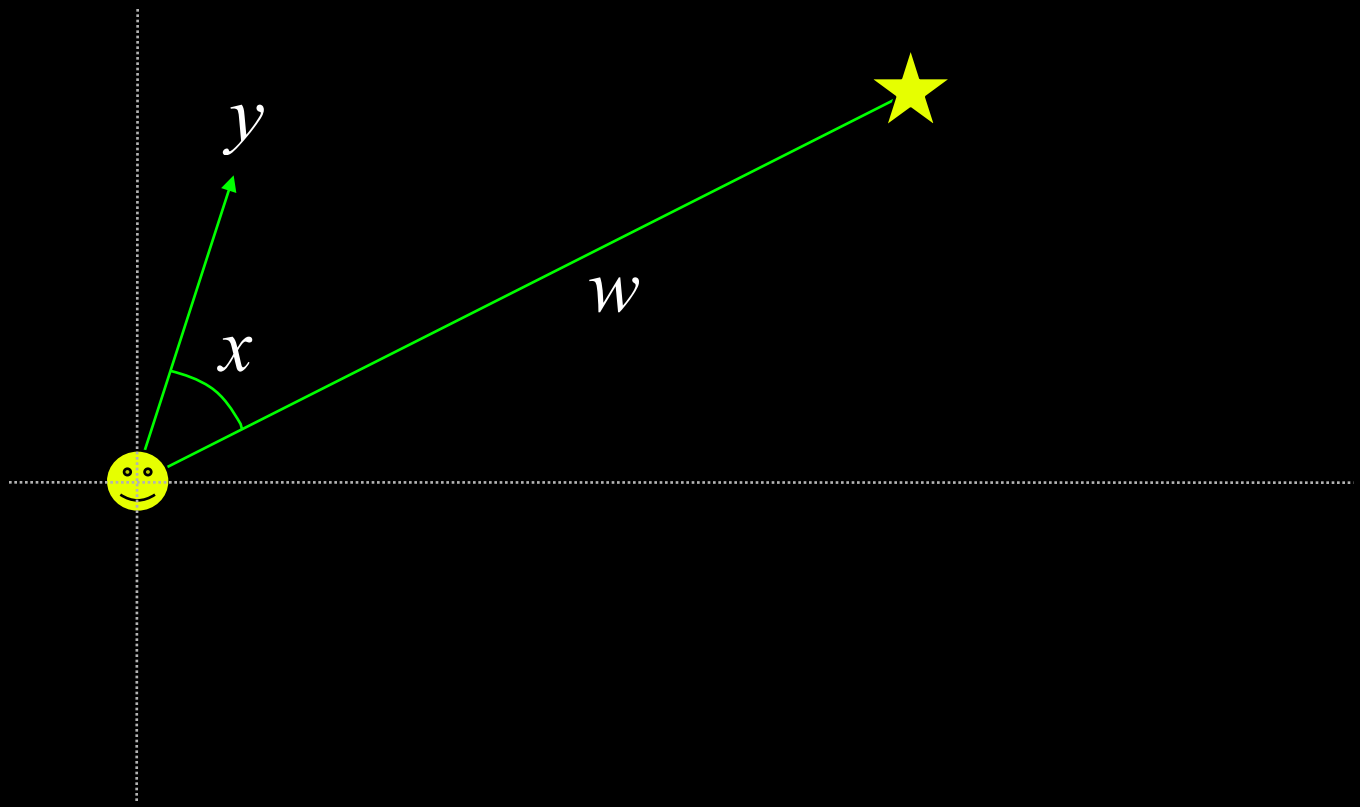
- Where w is the distance to the goal
  x is the angle to the goal
  $\dot{y}$ is the forward velocity, and,
  $\dot{x}$ is the angular velocity

- $A = \{\ddot{y}, \ddot{x}\}$ Discretized by taking fixed steps

  $R = \{-100, -1, 10\}$

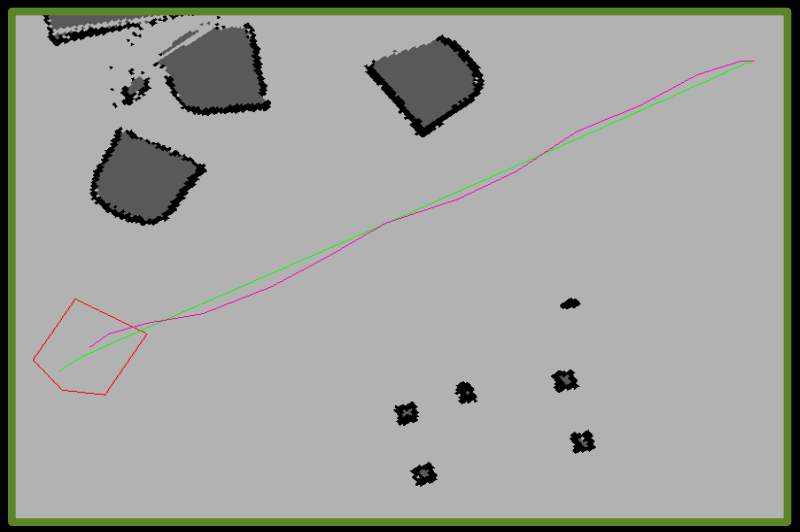
  – 10 at goal (terminal)
  – -100 if more than three meters from goal (terminal)
  – -1 elsewhere

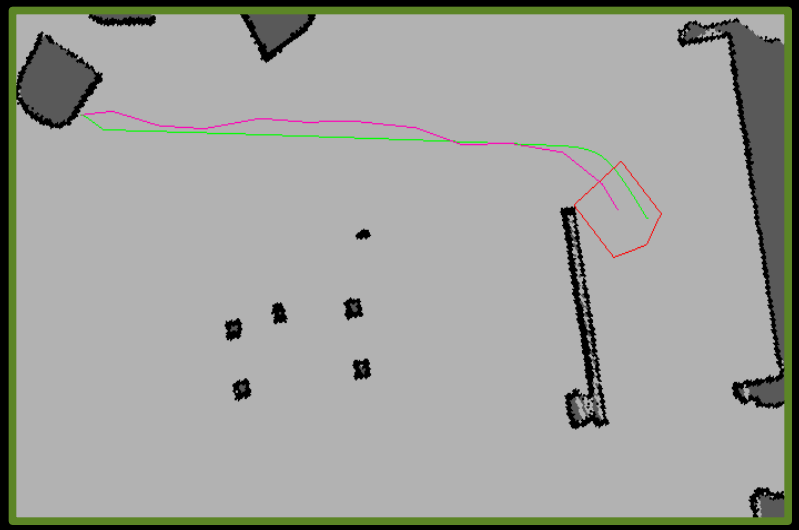
Learning To Drive To Goal



Willow Garage

Driving With The Learned Policy





Willow Garage

Driving With The Learned Policy

Willow Garage

Value Learning

Learning A Value Function Under a Fixed Policy

- Given a fixed policy $p : S \to A$ learn a value function to predict

$$V^p(s) \quad \forall \; s \in S$$

- Why?
  - This is an interesting intermediate research problem towards finding optimal policies
  - Allows us to calculate time to goal (or failure) for any given state
  - Allows us to choose between competing policies

Learning A Value Function Under a Fixed Policy

- Implemented two algorithms for value approximation
  - RKHS-SARSA
  - LSTD
- Applied each to navigation under the fixed policy set by the global and local planners

Willow Garage

RKHS-SARSA

- Developed a kernelized algorithm
- Decides which areas are important and places radial basis functions accordingly
- Designed to get maximum expressivity with minimum memory usage

Learning Values For Navigation

$S = \{x, y, \boxed{x}, \boxed{y}\}$

- Where x is the robot's global x coordinate
  y is the robot's global y coordinate
  $\boxed{x}$ is the goal's global x coordinate
  $\boxed{y}$ is the goal's global y coordinate

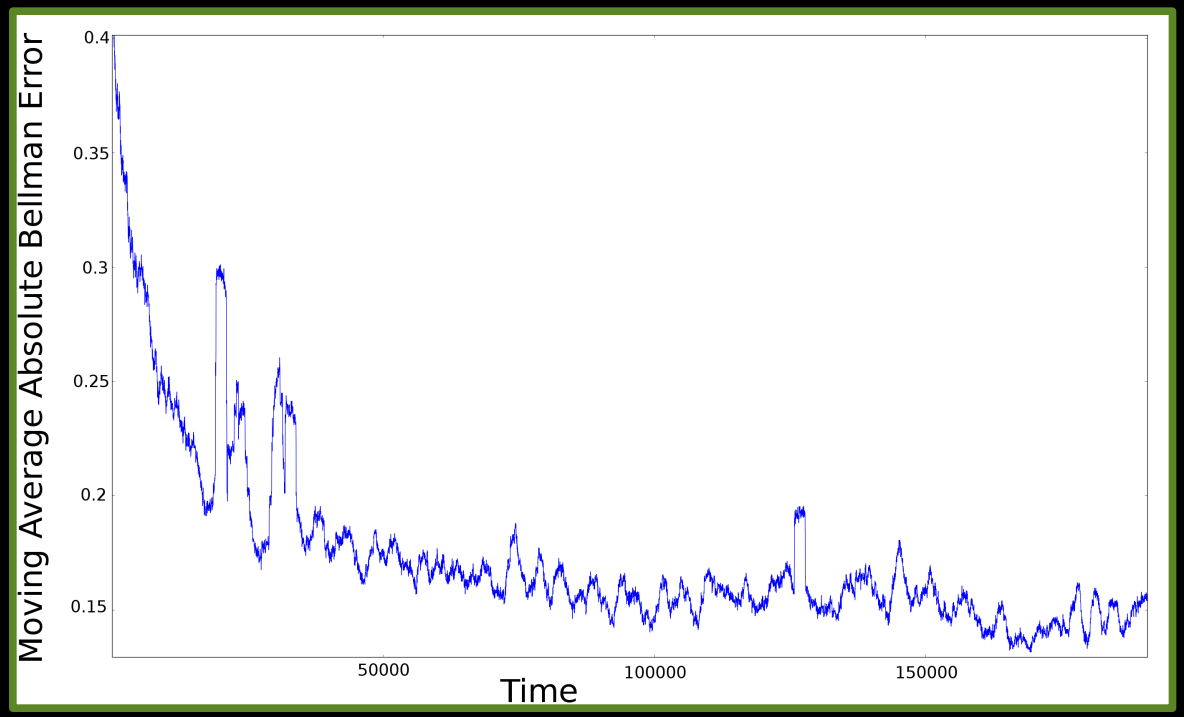
$R = \{-1, 0\}$

- 0 at goal (terminal)
- -1 elsewhere

Learning Curve

- Absolute Bellman error at time t

$$|V(s_t) - (r_t + cV(s_{t+1}))|$$



Willow Garage

Visualizing The Value Function

Willow Garage

Visualizing The Value Function

Willow Garage

Conclusions And Future Work

- Minor success achieved in reinforcement learning control
- Value prediction infrastructure created
- With more data and time we could learn a very comprehensive prediction for navigation under various policies
- Can be applied to any task where one wishes to compare policies or wants to know the time to goal


Willow
Garage