CS344M
Autonomous Multiagent Systems

Todd Hester

Department of Computer Science
The University of Texas at Austin
Good Afternoon, Colleagues

Are there any questions?
Good Afternoon, Colleagues

Are there any questions?

- Changes from 2011 to now
- Do different formations in different situations?
- How does UT’s walk engine work?
- Has the formation code been released? copied?
- Why does world model give 0s for some players? Unseen?
Good Afternoon, Colleagues

Are there any questions?

- Changes from 2011 to now
- Do different formations in different situations?
- How does UT’s walk engine work?
- Has the formation code been released? copied?
- Why does world model give 0s for some players? Unseen?
- Todd: Why not run CMA-ES to optimize role positions too?
Logistics

- Assignment 4 due today
Logistics

- Assignment 4 due today
- Next week’s readings posted
Logistics

- Assignment 4 due today
- Next week’s readings posted
- Final project proposal assigned
Final Projects

- Proposal (10/11): 3+ pages
  - What you’re going to do; graded on writing
Final Projects

- *Proposal (10/11):* 3+ pages
  - What you’re going to do; graded on writing

- *Progress Report (11/8):* 5+ pages + binaries + logs
  - What you’ve been doing; graded on writing
Final Projects

- **Proposal (10/11):** 3+ pages
  - What you’re going to do; graded on writing

- **Progress Report (11/8):** 5+ pages + binaries + logs
  - What you’ve been doing; graded on writing

- **Peer Review (11/15):** review 2 progress reports
  - Clear? suggestions?; graded on writing and feedback quality
Final Projects

- *Team (12/4): source + binaries*
  - The tournament entry; make sure it runs!
Final Projects

- **Team (12/4):** source + binaries
  - The tournament entry; make sure it runs!

- **Final Report (12/6):** 8+ pages
  - A term paper; the main component of your grade
Final Projects

- **Team (12/4):** source + binaries
  - The tournament entry; make sure it runs!

- **Final Report (12/6):** 8+ pages
  - A term paper; the main component of your grade

- **Tournament (12/17):** nothing due
  - Oral presentation
Final Projects

- **Team (12/4):** source + binaries
  - The tournament entry; make sure it runs!

- **Final Report (12/6):** 8+ pages
  - A term paper; the main component of your grade

- **Tournament (12/17):** nothing due
  - Oral presentation

**Due at beginning of classes**
Final Project info

• All writing is individual!
Final Project info

- All writing is individual!
- Two hard copies and one electronic copy
Final Project info

- All writing is individual!
- Two hard copies and one electronic copy
- Due at beginning of class
Final Project info

- All writing is individual!
- Two hard copies and one electronic copy
- Due at beginning of class
- One idea: Re-implement an idea from one of the readings
Final Project info

- All writing is individual!
- Two hard copies and one electronic copy
- Due at beginning of class
- One idea: Re-implement an idea from one of the readings
- Be careful with machine learning
Final Project info

- All writing is individual!
- Two hard copies and one electronic copy
- Due at beginning of class
- One idea: Re-implement an idea from one of the readings
- Be careful with machine learning
- Example final report on website
Overview of the Readings

- *Darwin*: genetic programming approach
Overview of the Readings

- *Darwin*: genetic programming approach
- *Stone and McAllester*: Architecture for action selection
Overview of the Readings

- *Darwin*: genetic programming approach
- *Stone and McAllester*: Architecture for action selection
- *Riley et al*: Coach competition, extracting models
Overview of the Readings

- *Darwin*: genetic programming approach
- *Stone and McAllester*: Architecture for action selection
- *Riley et al*: Coach competition, extracting models
- *Kuhlmann et al*: Learning for coaching
Overview of the Readings

- **Darwin**: genetic programming approach
- **Stone and McAllester**: Architecture for action selection
- **Riley et al**: Coach competition, extracting models
- **Kuhlmann et al**: Learning for coaching
- **Withopf and Riedmiller**: Reinforcement learning
Overview of the Readings

- **Darwin**: genetic programming approach
- **Stone and McAllester**: Architecture for action selection
- **Riley et al**: Coach competition, extracting models
- **Kuhlmann et al**: Learning for coaching
- **Withopf and Riedmiller**: Reinforcement learning
- **MacAlpine et al**: UT Austin Villa 2011
Overview of the Readings

- **Darwin**: genetic programming approach
- **Stone and McAllester**: Architecture for action selection
- **Riley et al**: Coach competition, extracting models
- **Kuhlmann et al**: Learning for coaching
- **Withopf and Riedmiller**: Reinforcement learning
- **MacAlpine et al**: UT Austin Villa 2011
- **Barrett et al**: SPL Kicking strategy
Evolutionary Computation

- Motivated by biological evolution: GA, GP
Evolutionary Computation

- Motivated by biological evolution: GA, GP
- Search through a space
Evolutionary Computation

- Motivated by biological evolution: GA, GP

- Search through a space
  - Need a representation, fitness function
  - Probabilistically apply search operators to set of points in search space
Evolutionary Computation

- Motivated by biological evolution: GA, GP
- Search through a space
  - Need a representation, fitness function
  - Probabilistically apply search operators to set of points in search space
- Randomized, parallel hill-climbing through space
Evolutionary Computation

- Motivated by biological evolution: GA, GP

- Search through a space
  - Need a representation, fitness function
  - Probabilistically apply search operators to set of points in search space

- Randomized, parallel hill-climbing through space

- Learning is an optimization problem (fitness)
Evolutionary Computation

- Motivated by biological evolution: GA, GP
- Search through a space
  - Need a representation, fitness function
  - Probabilistically apply search operators to set of points in search space
- Randomized, parallel hill-climbing through space
- Learning is an optimization problem (fitness)

Some slides from *Machine Learning* (Mitchell, 1997)
Darwin United

- More ambitious follow-up to Luke, 97 (made 2nd round)
Darwin United

- More ambitious follow-up to Luke, 97 (made 2nd round)
- Motivated in part by Peter’s detailed team construction
Darwin United

- More ambitious follow-up to Luke, 97 (made 2nd round)
- Motivated in part by Peter’s detailed team construction
- Evolves whole teams — lexicographic fitness function
Darwin United

- More ambitious follow-up to Luke, 97 (made 2nd round)
- Motivated in part by Peter’s detailed team construction
- Evolves whole teams — lexicographic fitness function
- Evolved on huge (at the time) hypercube
Darwin United

- More ambitious follow-up to Luke, 97 (made 2nd round)
- Motivated in part by Peter’s detailed team construction
- Evolves whole teams — lexicographic fitness function
- Evolved on huge (at the time) hypercube
- Lots of spinning, but figured out dribbling, offsides
Darwin United

- More ambitious follow-up to Luke, 97 (made 2nd round)
- Motivated in part by Peter’s detailed team construction
- Evolves whole teams — lexicographic fitness function
- Evolved on huge (at the time) hypercube
- Lots of spinning, but figured out dribbling, offsides
- 1-1-1 record. Tied a good team, but didn’t advance
Darwin United

- More ambitious follow-up to Luke, 97 (made 2nd round)
- Motivated in part by Peter’s detailed team construction
- Evolves whole teams — lexicographic fitness function
- Evolved on huge (at the time) hypercube
- Lots of spinning, but figured out dribbling, offsides
- 1-1-1 record. Tied a good team, but didn’t advance
- Success of the method, but not pursued
Architecture for Action Selection

• (other slides, video)
Architecture for Action Selection

- (other slides, video)
- downsides
Architecture for Action Selection

- (other slides, video)
- downsides
- Keepaway
Coaching

- Learn best strategy to play a fixed team
Coaching

- Learn best strategy to play a fixed team
- Give high level advice to players at low frequency
Coaching

• Learn best strategy to play a fixed team
• Give high level advice to players at low frequency
• Focus on learning formations
Coaching

- Learn best strategy to play a fixed team
- Give high level advice to players at low frequency
- Focus on learning formations
- Learn when successful teams passed/kicked
Coaching

- Learn best strategy to play a fixed team
- Give high level advice to players at low frequency
- Focus on learning formations
- Learn when successful teams passed/kicked
- Learn when opponent will pass and try to block
Coaching

• Learn best strategy to play a fixed team
• Give high level advice to players at low frequency
• Focus on learning formations
• Learn when successful teams passed/kicked
• Learn when opponent will pass and try to block
• What if players switch roles?
Coaching

- Learn best strategy to play a fixed team
- Give high level advice to players at low frequency
- Focus on learning formations
- Learn when successful teams passed/kicked
- Learn when opponent will pass and try to block
- What if players switch roles?
- Why just imitate another team?
Coaching

- Learn best strategy to play a fixed team
- Give high level advice to players at low frequency
- Focus on learning formations
- Learn when successful teams passed/kicked
- Learn when opponent will pass and try to block
- What if players switch roles?
- Why just imitate another team?
- Other slides
Reinforcement Learning

• RL Slides
Reinforcement Learning

- RL Slides
- Extend to grid soccer
Reinforcement Learning

- RL Slides
- Extend to grid soccer
- Large state space, joint actions
Reinforcement Learning

- RL Slides
- Extend to grid soccer
- Large state space, joint actions
- Address this with state aliasing, options
Reinforcement Learning

- RL Slides
- Extend to grid soccer
- Large state space, joint actions
- Address this with state aliasing, options
- Successfully learn the task, use for some of team behavior
Reinforcement Learning

- RL Slides
- Extend to grid soccer
- Large state space, joint actions
- Address this with state aliasing, options
- Successfully learn the task, use for some of team behavior
- However, takes 12 million actions to learn
• Other slides
• Other slides

• Why not use CMA-ES on role positions as well?
• Other slides

• Why not use CMA-ES on role positions as well?

• Changes for 2012?
Kicking Under Uncertainty

- Previous SPL approach: always rotate to kick at goal
Kicking Under Uncertainty

- Previous SPL approach: always rotate to kick at goal
- Kick engine to kick at various distances/headings
Kicking Under Uncertainty

- Previous SPL approach: always rotate to kick at goal
- Kick engine to kick at various distances/headings
- Adjust to seen ball location
Kicking Under Uncertainty

• Previous SPL approach: always rotate to kick at goal

• Kick engine to kick at various distances/headings

• Adjust to seen ball location

• Select first kick that moves ball up field
Kicking Under Uncertainty

- Previous SPL approach: always rotate to kick at goal
- Kick engine to kick at various distances/headings
- Adjust to seen ball location
- Select first kick that moves ball up field
- Figure
Kicking Under Uncertainty

- Previous SPL approach: always rotate to kick at goal
- Kick engine to kick at various distances/headings
- Adjust to seen ball location
- Select first kick that moves ball up field
- Figure
- Emphasis on quickness
Kicking Under Uncertainty

- Previous SPL approach: always rotate to kick at goal
- Kick engine to kick at various distances/headings
- Adjust to seen ball location
- Select first kick that moves ball up field
- Figure
- Emphasis on quickness
- Now: Better model of opponents -> Know if we have more time
Learning Keepaway

KEEPAWAY SLIDES
Learning Commentary

• David Chen and Ray Mooney
Coordination Graphs

- $n$ agents, each choose an action $A_i$
Coordination Graphs

- $n$ agents, each choose an action $A_i$

- $A = A_1 \times \ldots \times A_n$
Coordination Graphs

- $n$ agents, each choose an action $A_i$

- $A = A_1 \times \ldots \times A_n$

- $R_i(A) \rightarrow \mathbb{R}$
Coordination Graphs

- $n$ agents, each choose an action $A_i$
- $A = A_1 \times \ldots \times A_n$
- $R_i(A) \mapsto \mathbb{R}$
- Coordination problem: $R_1 = \ldots = R_n = R$
Coordination Graphs

- $n$ agents, each choose an action $A_i$

- $A = A_1 \times \ldots \times A_n$

- $R_i(A) \mapsto \mathbb{R}$

- Coordination problem: $R_1 = \ldots = R_n = R$

- Nash equilibrium: no agent could do better given what others are doing.
Coordination Graphs

- $n$ agents, each choose an action $A_i$
- $A = A_1 \times \ldots \times A_n$
- $R_i(A) \rightarrow \mathbb{R}$
- Coordination problem: $R_1 = \ldots = R_n = R$
- Nash equilibrium: no agent could do better given what others are doing.
- May be more than one (chicken)
Example from the paper

- Understand the rule syntax
Example from the paper

- Understand the rule syntax
- Form the coordination graph
Example from the paper

- Understand the rule syntax
- Form the coordination graph
- First eliminate rules based on context
Example from the paper

- Understand the rule syntax
- Form the coordination graph
- First eliminate rules based on context
- What does it mean for $G_3$ to collect all relevant rules?
Example from the paper

- Understand the rule syntax
- Form the coordination graph
- First eliminate rules based on context
- What does it mean for $G_3$ to collect all relevant rules?
- What does it mean for $G_3$ to maximize over all actions of $a_1$ and $a_2$?
Example from the paper

- Understand the rule syntax
- Form the coordination graph
- First eliminate rules based on context

- What does it mean for $G_3$ to collect all relevant rules?
- What does it mean for $G_3$ to maximize over all actions of $a_1$ and $a_2$?
- How are the results propagated back?
Example from the paper

- Understand the rule syntax
- Form the coordination graph
- First eliminate rules based on context
- What does it mean for $G_3$ to collect all relevant rules?
- What does it mean for $G_3$ to maximize over all actions of $a_1$ and $a_2$?
- How are the results propagated back?
- Let’s try again with $G_1$ eliminated first
Application to soccer

• Make the world discrete by assigning roles, using high-level predicates
Application to soccer

- Make the world discrete by assigning roles, using high-level predicates
- Assume global state information
Application to soccer

- Make the world discrete by assigning roles, using high-level predicates
- Assume global state information
- Finds pass sequences and starts players moving ahead of time.
Application to soccer

- Make the world discrete by assigning roles, using high-level predicates
- Assume global state information
- Finds pass sequences and starts players moving ahead of time.
- Note the results: with and without coordination.
Reactive Deliberation

- A hybrid approach

- Executor: carry out reactive behaviors

- Deliberator: evaluate possible high-level schema with parameters; generate bids

- Deliberator takes time, but something keeps happening always.

- In effect: deliberator commits to schema for some time