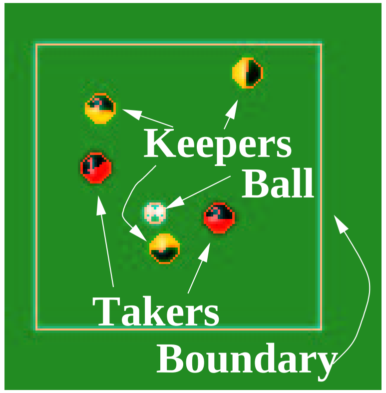
Guiding a Reinforcement Learner with Natural Language Advice

Initial Results in RoboCup Soccer



Gregory Kuhlmann

Department of Computer Sciences

University of Texas at Austin

Joint work with

Peter Stone, Raymond Mooney, and Jude Shavlik

Department of Computer Sciences

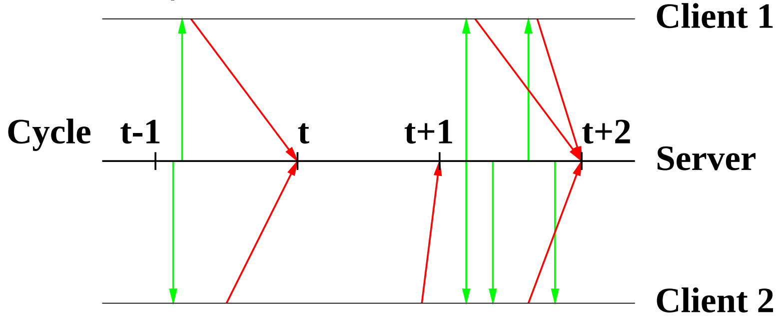
The University of Texas at Austin

Project Overview

- Human provides assistance to learning agents
- Many types of interaction possible
- Interaction:
  - Human observes agent learning to perform task by RL
  - Gives advice in natural language
    - specifies condition and advised action
- Components:
  1. Translate natural language advice into formal representation
  2. Integrate advice into learning agent

Domain: RoboCup Simulator

- Distributed: each player a separate client
- Server models dynamics and kinematics
- Clients receive sensations, send actions



- Parametric actions: dash, turn, kick, say
- Abstract, noisy sensors, hidden state
  - Hear sounds from limited distance
  - See relative distance, angle to objects ahead
- $> 10^{9^{23}}$ states
- Limited resources : stamina
- Play occurs in real time ($\approx$ human parameters)

CLang

- Standardized Coach Language
  - independent of coachable player's behavior representation
- If-then rules:
  $\{condition\} \rightarrow \{action\}$
- Example:
  If our player 7 has the ball, then he should pass to player 8 or player 9

```
(definerule pass789 direc
  ((bowner our {7})
   (do our {7} (pass {8 9}))))
```

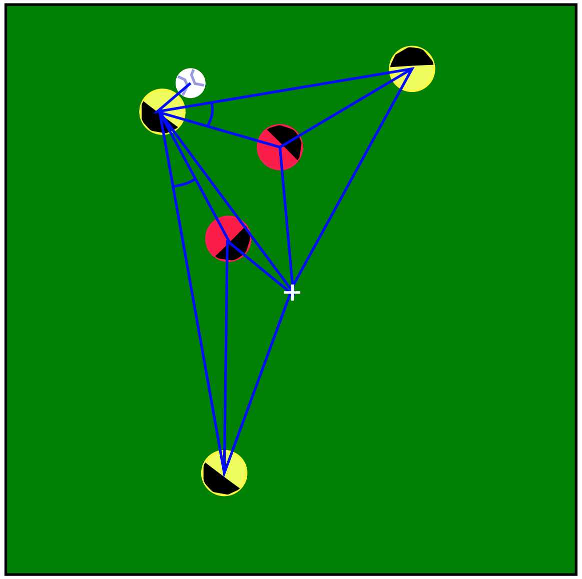
Learning to Map NL to CLang

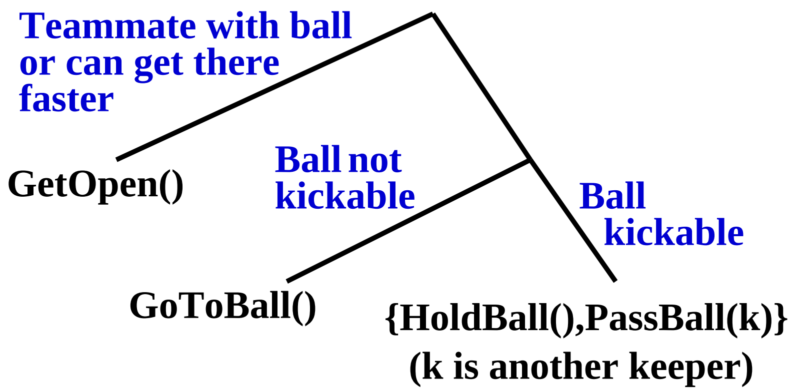
- Parsing NL and translating into formal language
  - Manageable with current NLP technology for restricted task
  - Labor-intensive to construct parser by hand
- Instead learn parser from input/output pairs
- Exploring several methods

Task: 3 vs. 2 Keepaway

- Play in a **small area** (20m $\times$ 20m)
- **Keepers** try to keep the ball
- **Takers** try to get the ball
- **Episode:**
  - Players and ball reset randomly
  - Ball starts near a keeper
  - Ends when taker gets the ball or ball goes out of bounds
- Performance measure: average episode duration

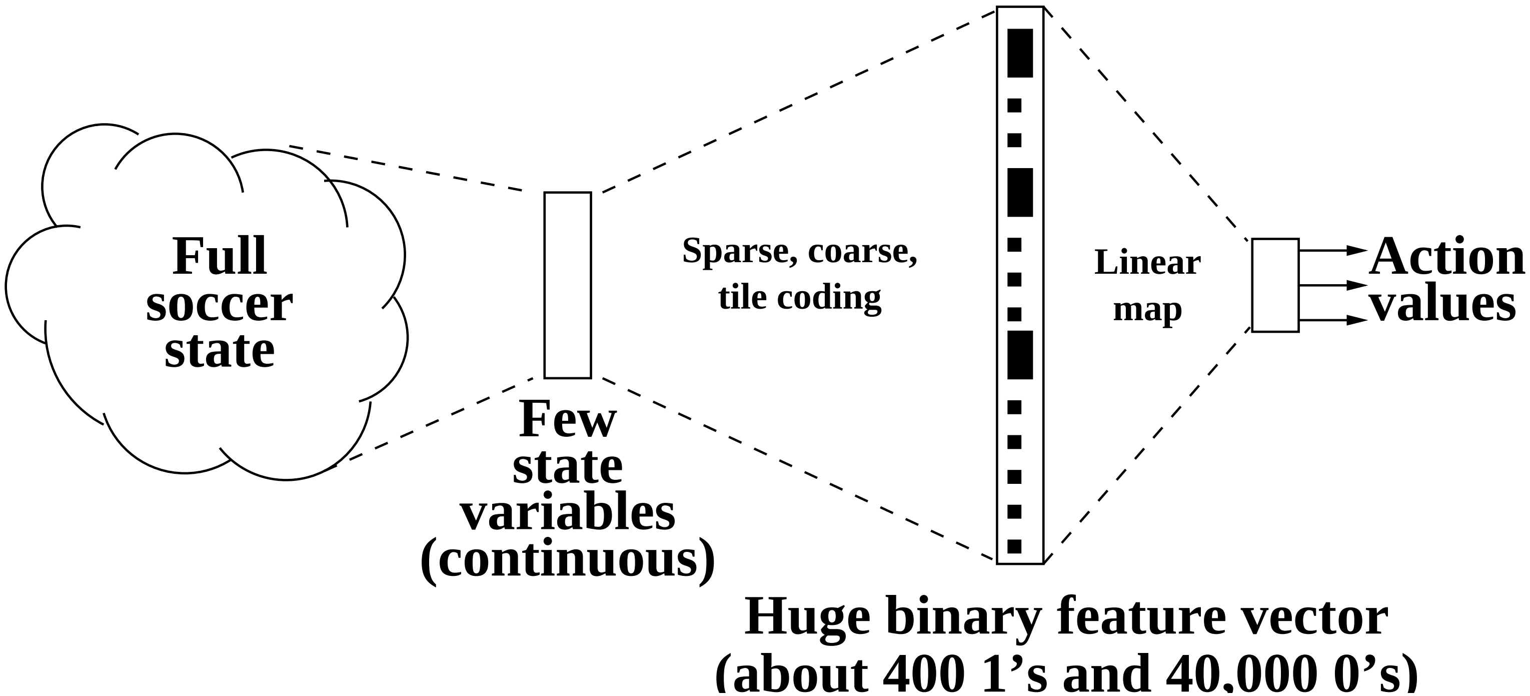
Keeper's State/Action Space

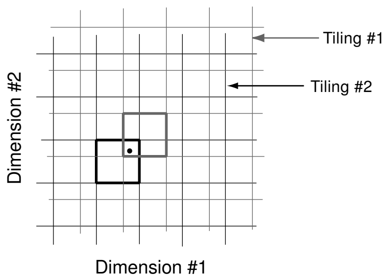




- Inputs: 11 distances among players, ball, and center and 2 angles to takers along passing lane
- Actions: Basic skills from CMUnited-99 team

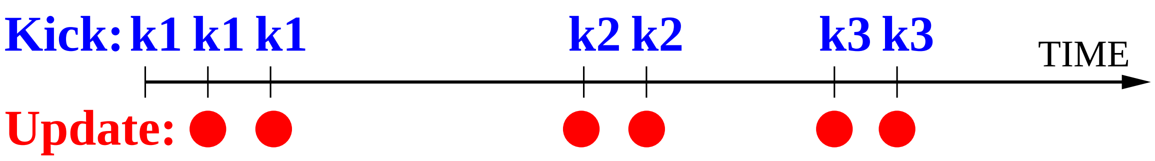
Function Approximation: Tile Coding



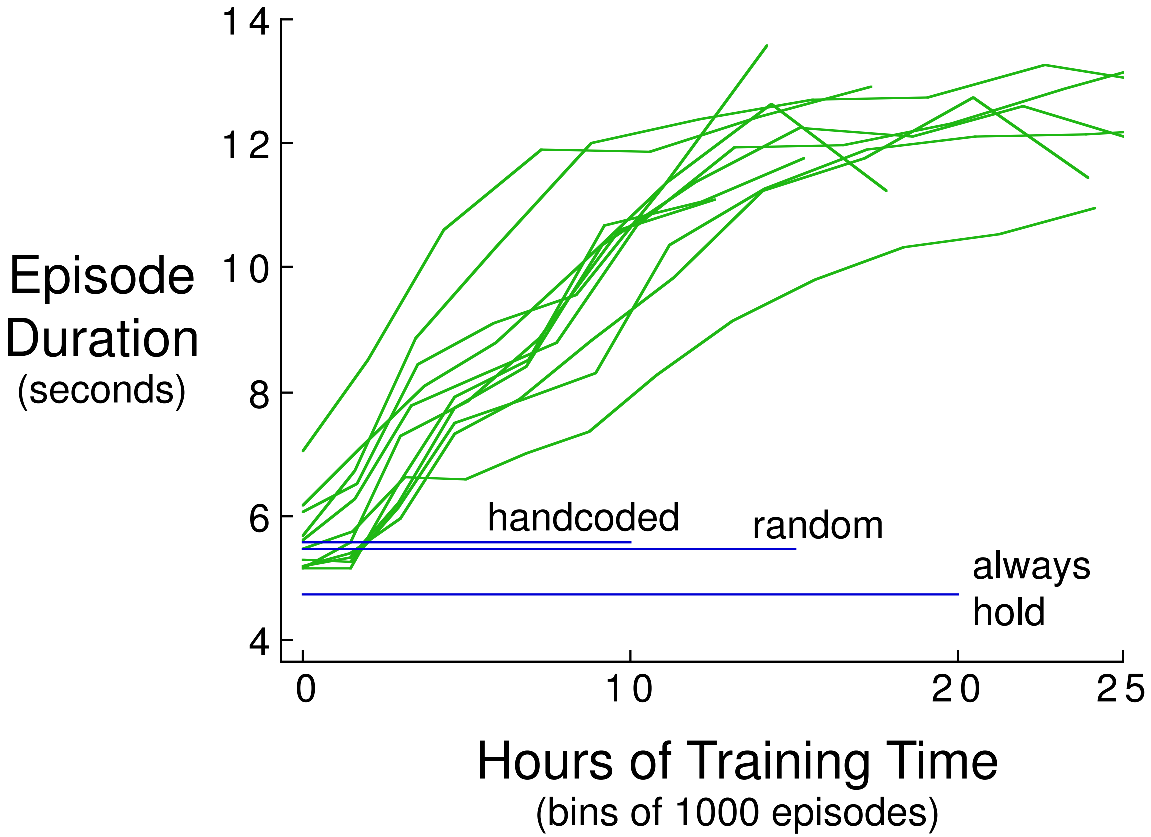


SMDP Sarsa(λ)

- Linear Sarsa(λ)
  - On-policy method: advantages over e.g. Q-learning
  - Not known to converge, but works (e.g. [Sutton, 1996])
- Only update when ball is kickable for **someone**:
  Semi-Markov Decision Process

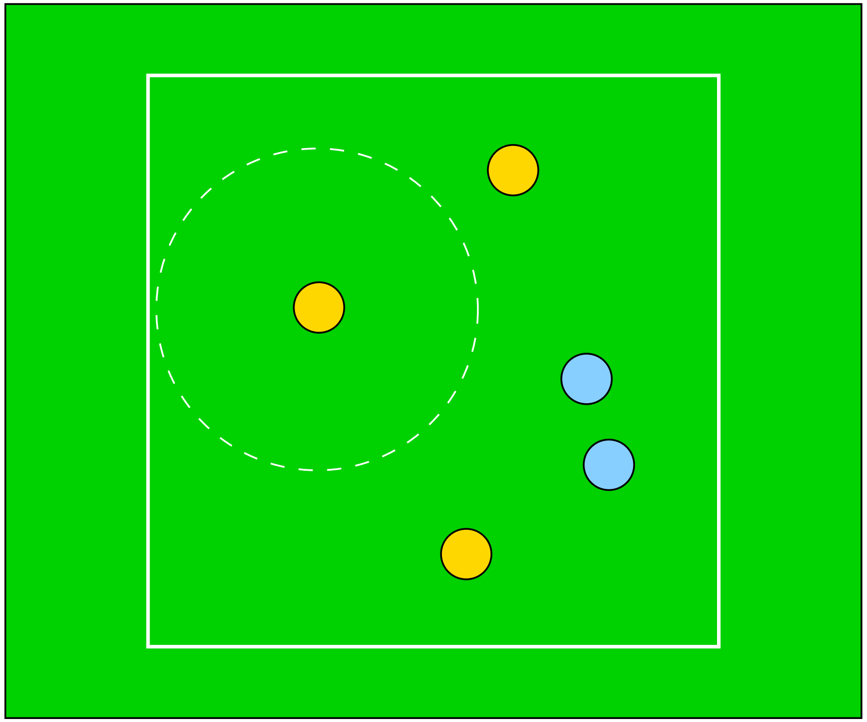


Prior Results Without Advice (Stone & Sutton, 2001)



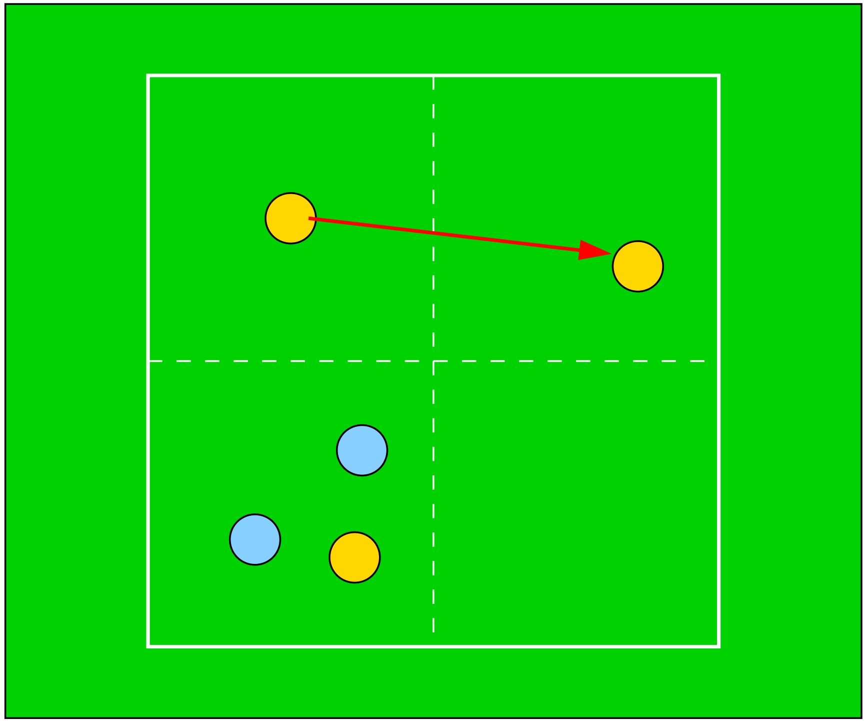
- Results scaled up to 6 vs. 5
- Robust to limited vision, and varying field sizes and state representations.

Example Advice



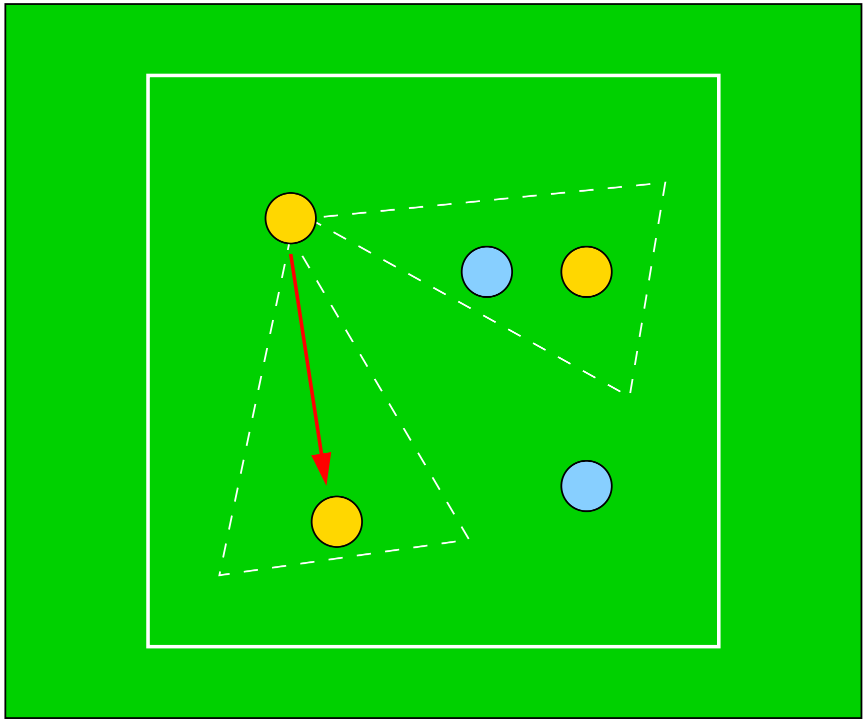
- If no opponents are within 8m then hold.

Example Advice (contd.)



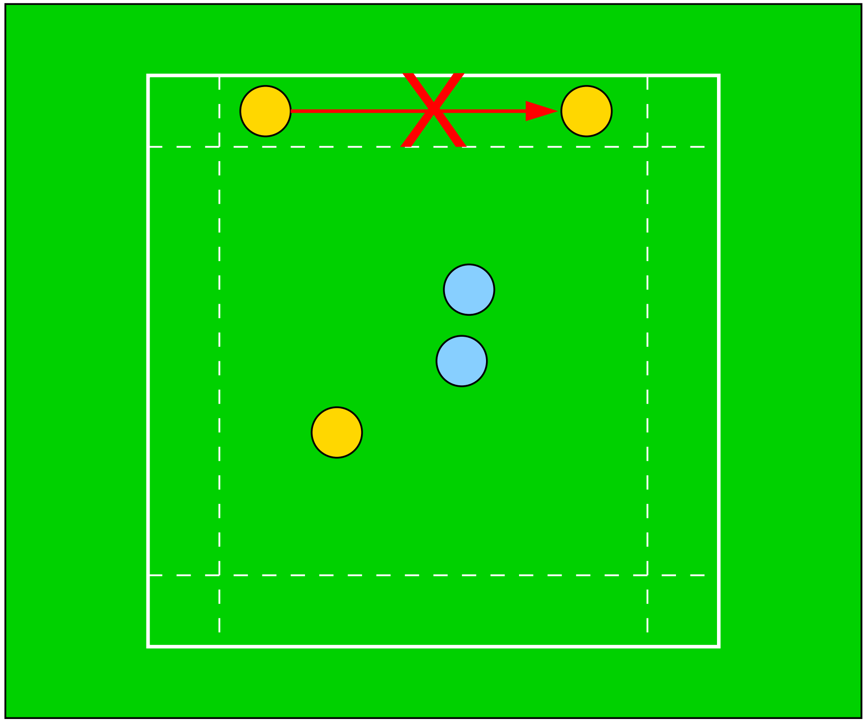
- If a teammate is in a quadrant with no opponents then pass to that teammate.

Example Advice (contd.)



- If a passing lane is open
  then use it.

Example Advice (contd.)

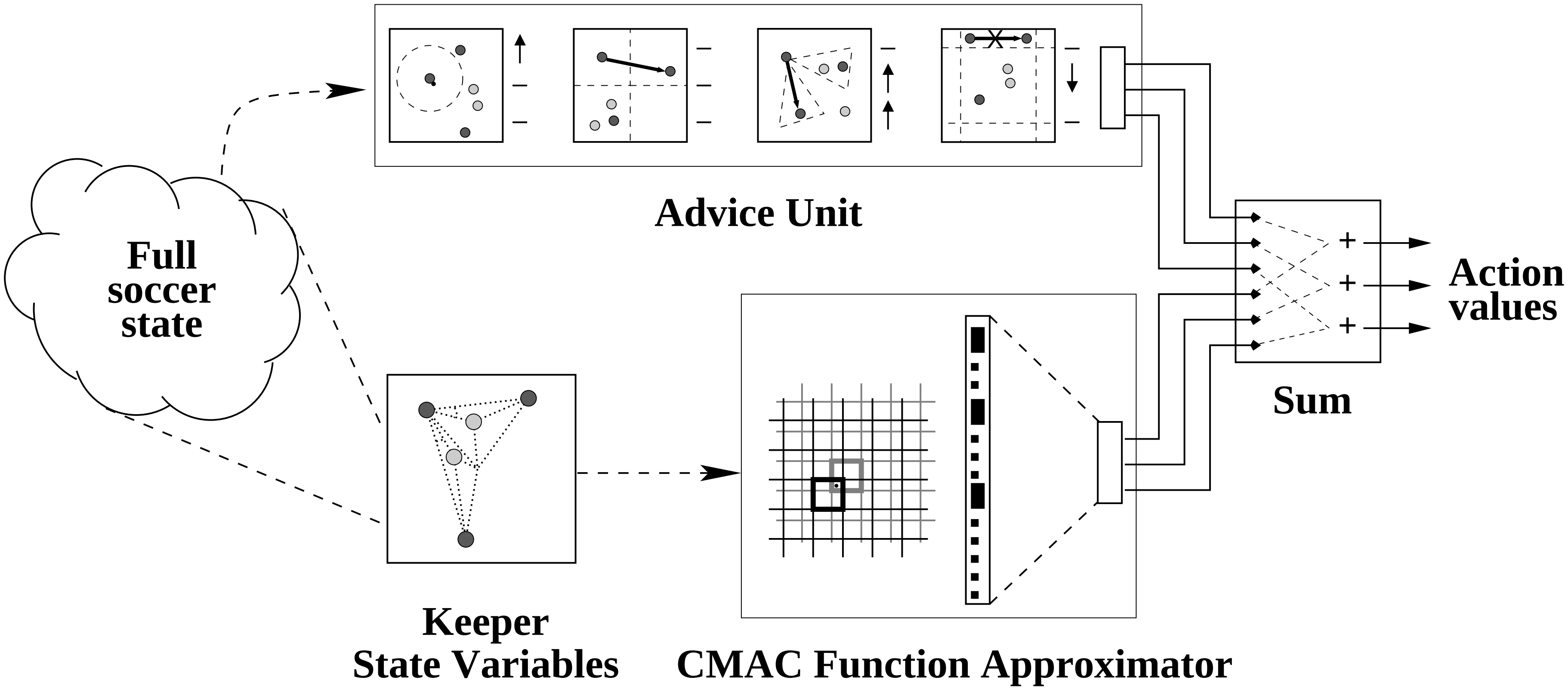


- Don't pass along edges.

Integrating Advice

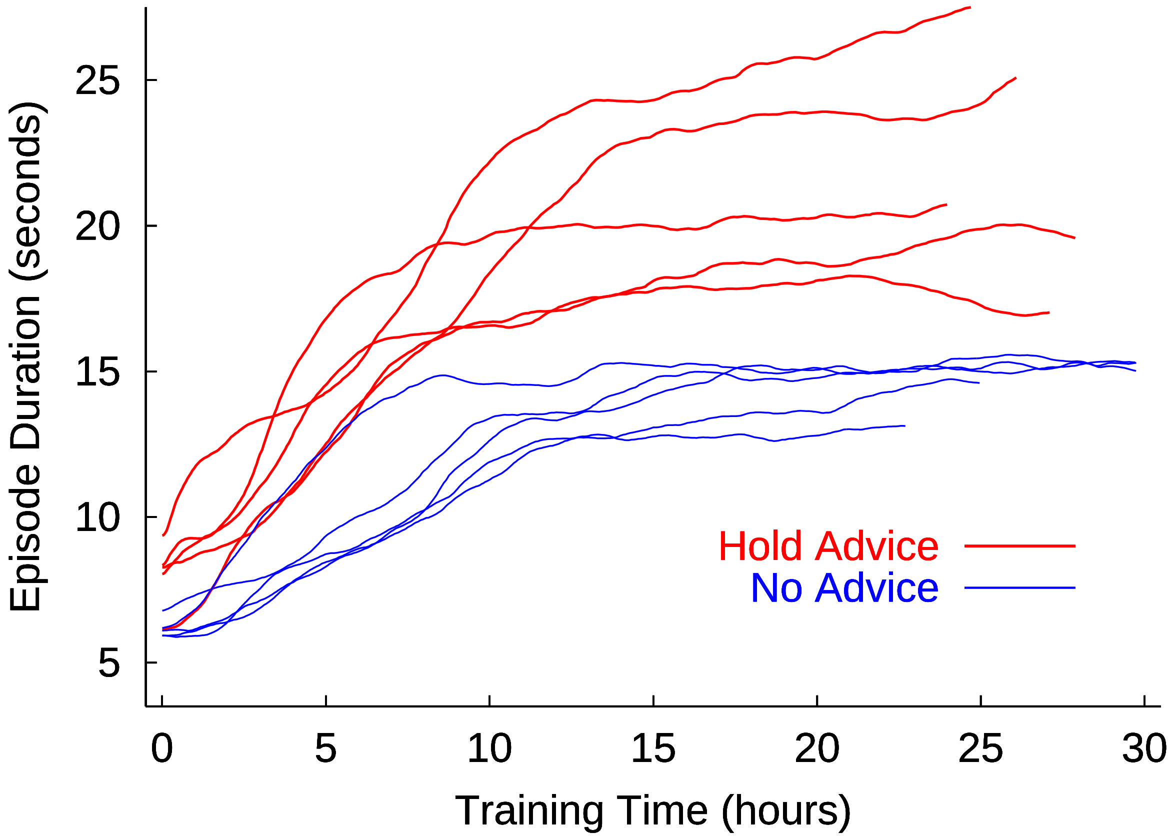
- Unchanged CMAC computes action value.
- New Advice Unit computes advice (0,+/-2)
- Values added to compute Q-value.
  - Q(s,a) = CMAC(s,a) + Advice(s,a)
- Example: hold advice
  - If no opponents are within 8m in s
  - then Q(s,*hold*) = CMAC(s,*hold*) + 2
  - else Q(s,*hold*) = CMAC(s,*hold*)

Integrating Advice (contd.)



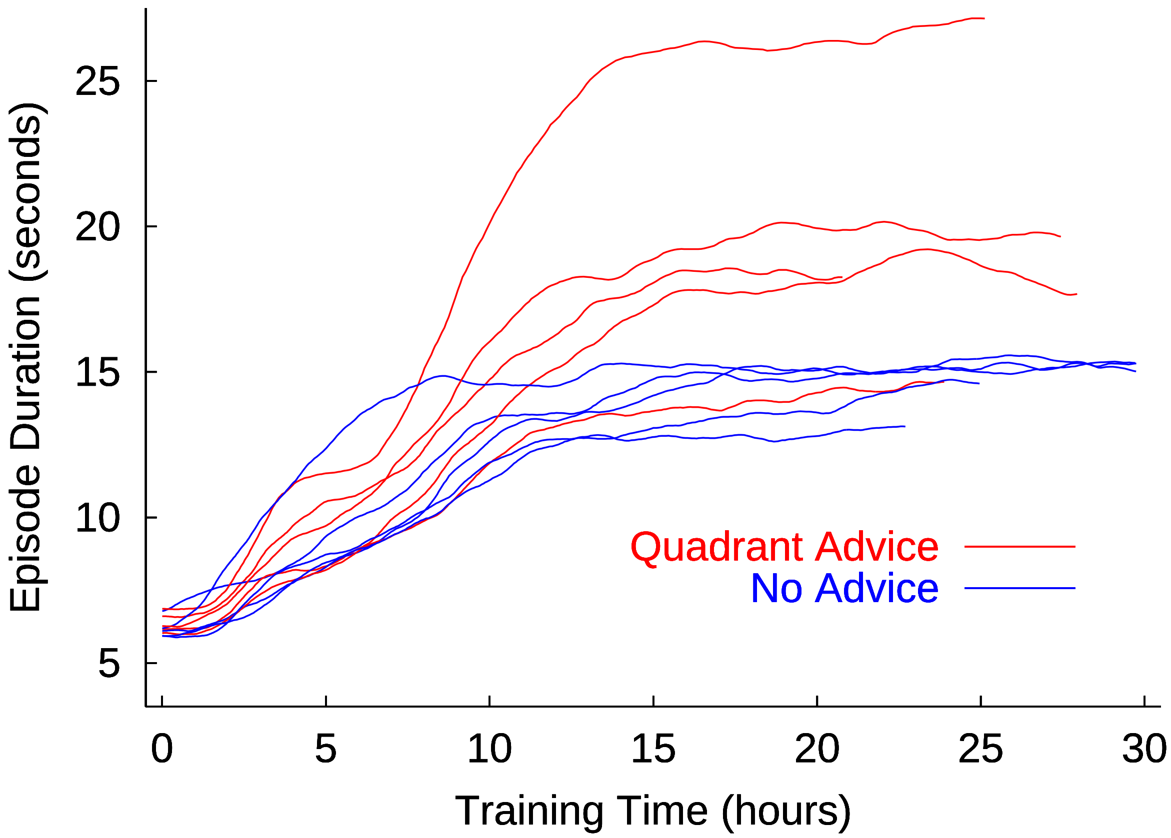
- Learner and advisor can have different state representations
- Should still be able to refine advice

“Hold” Advice

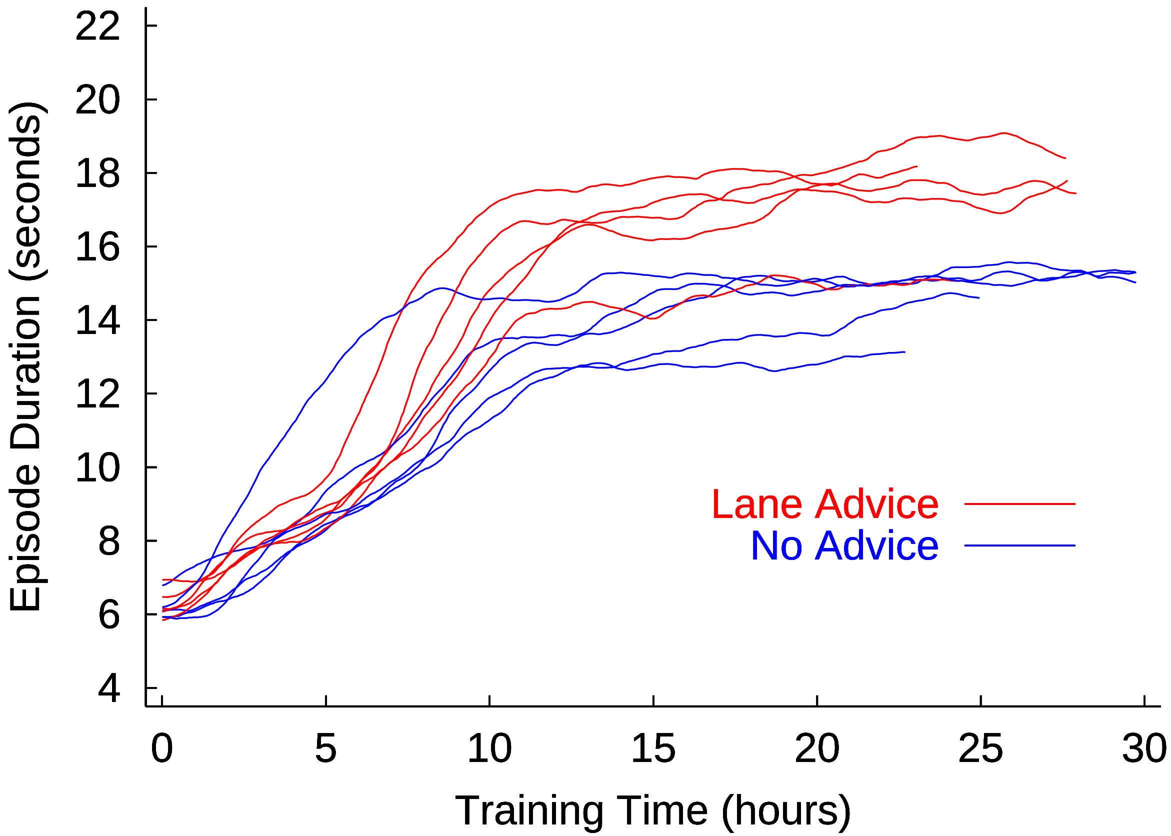


Episode Duration (seconds)
25
20
15
10
5
0
5
10
15
20
25
30
Training Time (hours)
Hold Advice
No Advice

“Quadrant” Advice

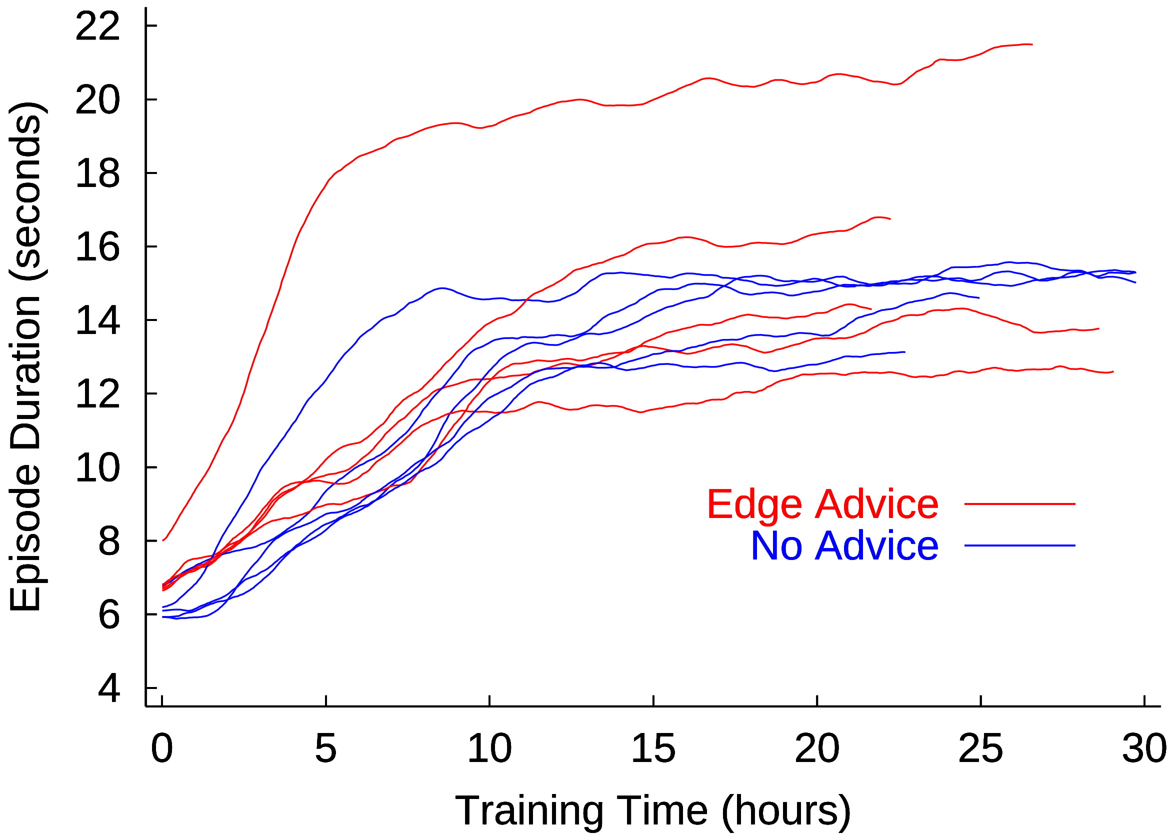


“Lane” Advice



Episode Duration (seconds)
22
20
18
16
14
12
10
8
6
4
0
5
10
15
20
25
30
Training Time (hours)
Lane Advice
No Advice

“Edge” Advice



Episode Duration (seconds)
22
20
18
16
14
12
10
8
6
4
0
5
10
15
20
25
30
Training Time (hours)
Edge Advice
No Advice

Conclusion and Future Work

- Simple, intuitive high-level advice can improve learning in a challenging, dynamic task.
- Advice helps learner find better policies
- Future enhancements:
  - Combined advice produces additive effect
  - Advice speeds up learning
  - Bad advice can be unlearned
- Future work in learning English to CLang mapping