

Progress in Learning 3 vs. 2 Keepaway

Gregory Kuhlmann
Department of Computer Sciences
University of Texas at Austin

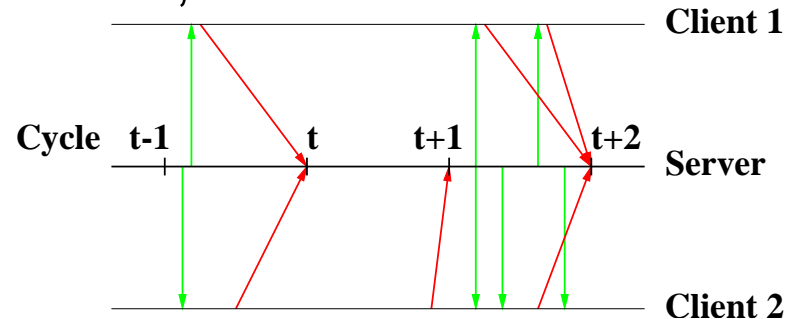
Joint work with
Peter Stone

RoboCup and Reinforcement Learning

- Reinforcement Learning — suited to soccer
 - Sequential decision making
 - Achieving delayed goals
 - Handling noise and stochasticity
 - Rapid decision-making
- Challenges
 - Multiple learning agents
 - Large state space
 - Not within realm of theoretical results

RoboCup Simulator

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

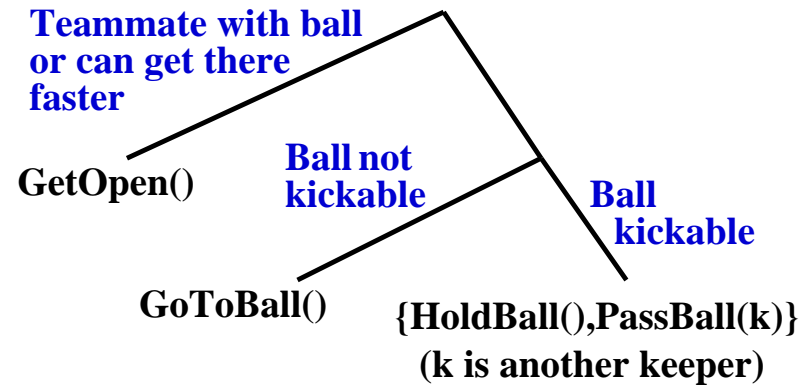


- Parametric actions: **dash, turn, kick, say**
- **Abstract, noisy** sensors, hidden state
 - **Hear** sounds from limited distance
 - **See** relative distance, angle to objects ahead
- $> 10^{9^{23}}$ states
- **Limited resources** : stamina
- Play occurs in **real time** (\approx human parameters)

3 vs. 2 Keepaway

- Play in a **small area** (20m × 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 Policy Space



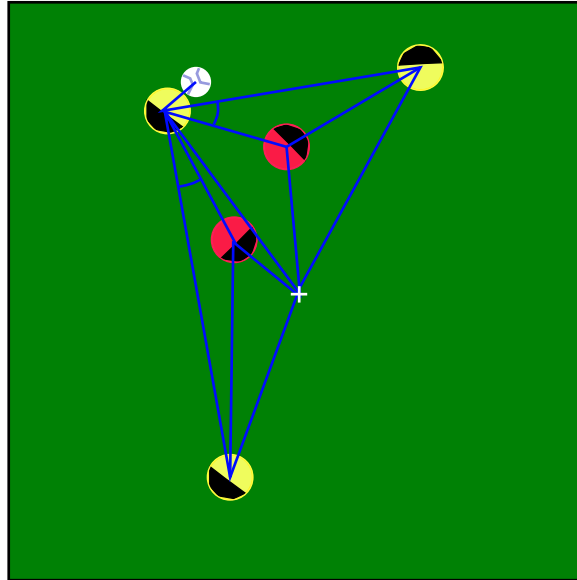
- Basic skills from CMUnited-99 team
- Example Policies
 - Random
 - Hold
 - Hand-coded

Mapping Keepaway to RL

Discrete-time, episodic, distributed RL

- Simulator operates in discrete time steps, $t = 0, 1, 2, \dots$, each representing 100 msec
- Episode: $s_0, a_0, r_1, s_1, \dots, s_t, a_t, r_{t+1}, s_{t+1}, \dots, r_T, s_T$
- $r_t = 1$
- $V^\pi(s) = E\{T \mid s_0 = s\}$
- Goal: Find π^* that maximizes V for all s

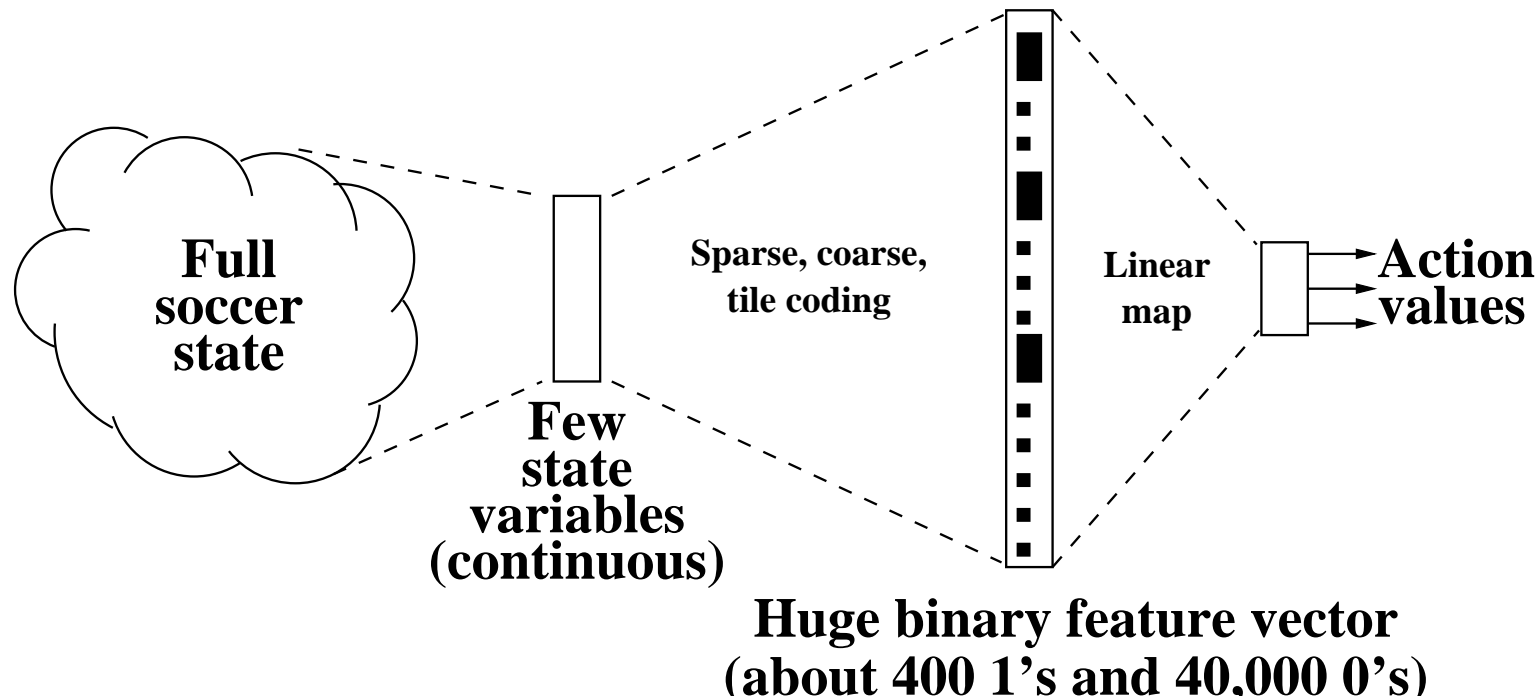
Keeper's State Variables



- 11 distances among players, ball, and center
- 2 angles to takers along passing lanes

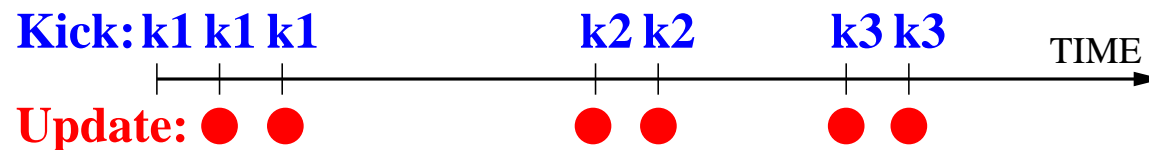
Function Approximation: Tile Coding

- Form of sparse, coarse coding based on **CMACS** [Albus, 1981]
- Tiled state variables **individually** (13)

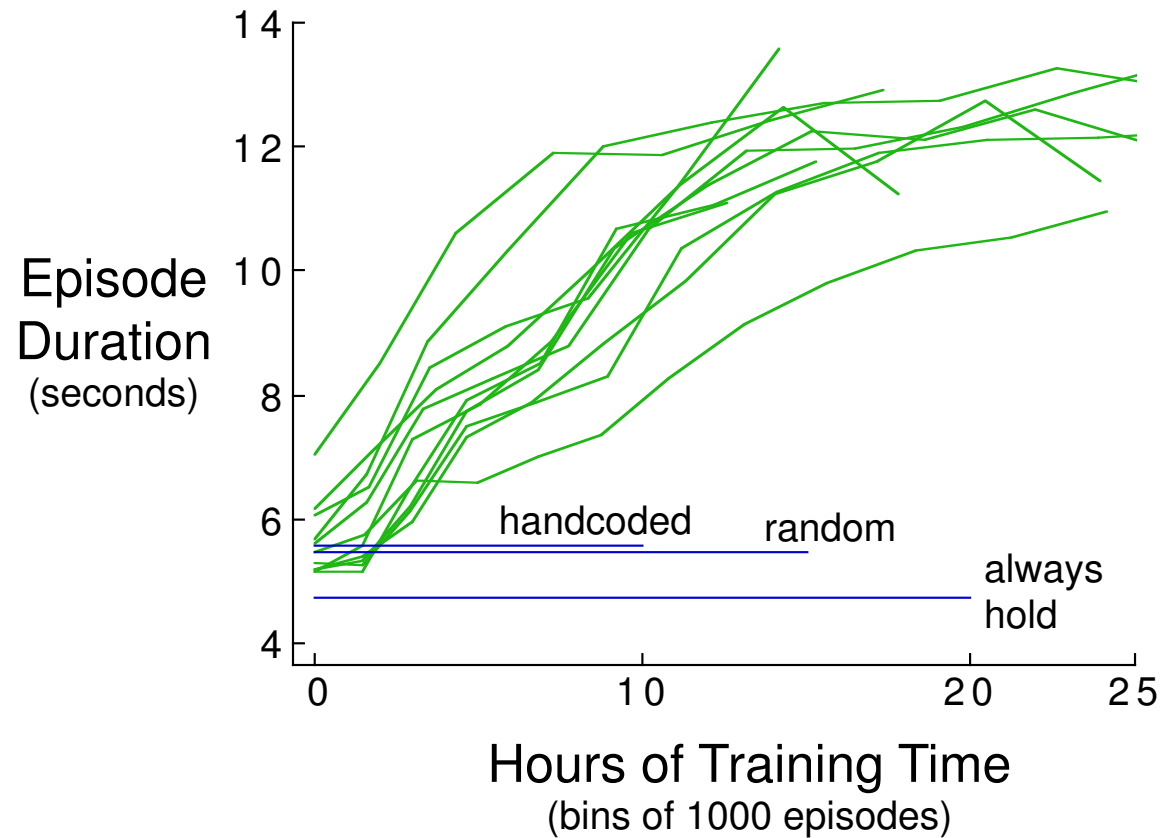


SMDP Sarsa(λ)

- 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

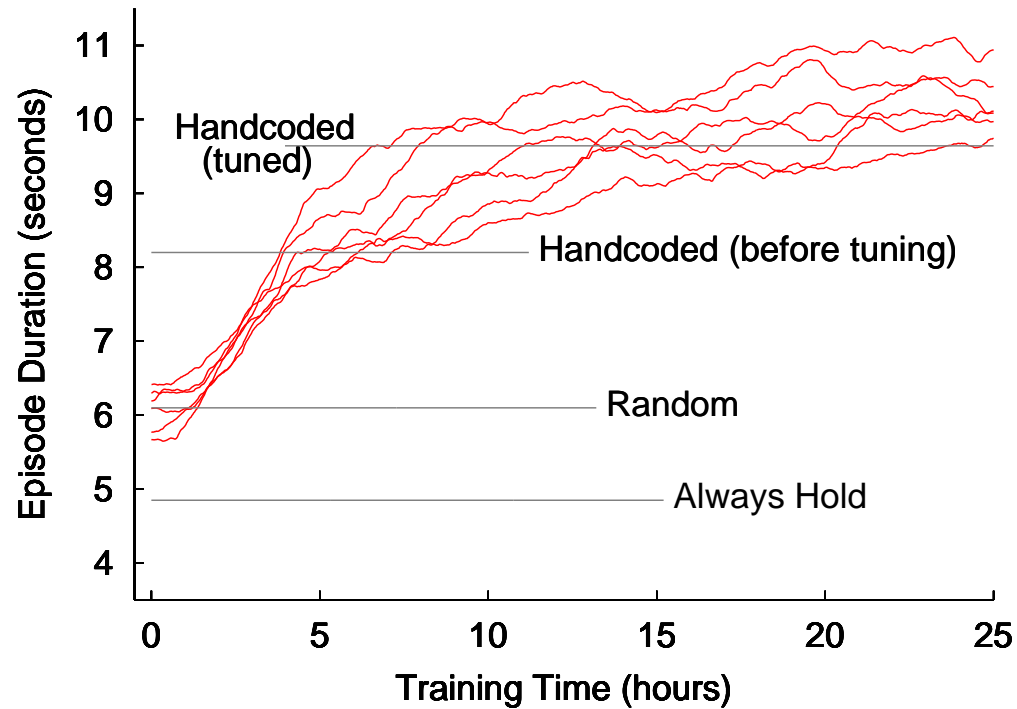


Previous Results



- Results scaled up to 4 vs. 3
- **360° view angle. No perceptual noise**

Limited Vision



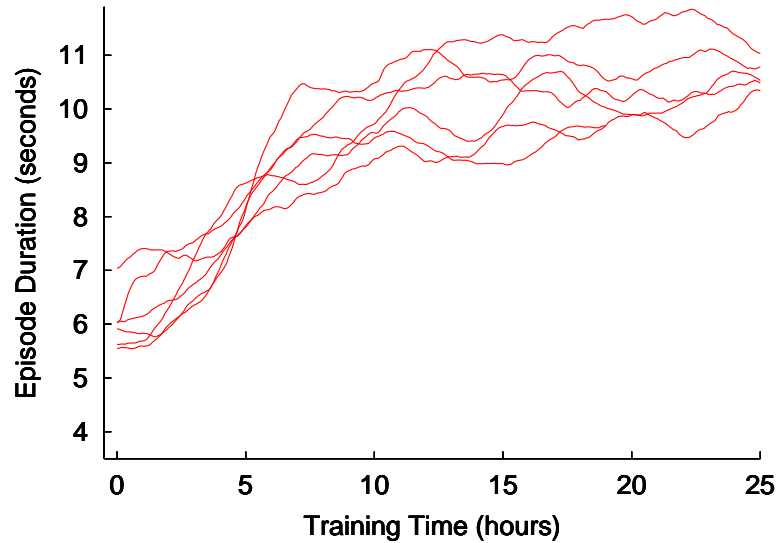
- With noise. Limited (90°) vision
- As good as **tuned** handcoded policy

Varying Field Sizes

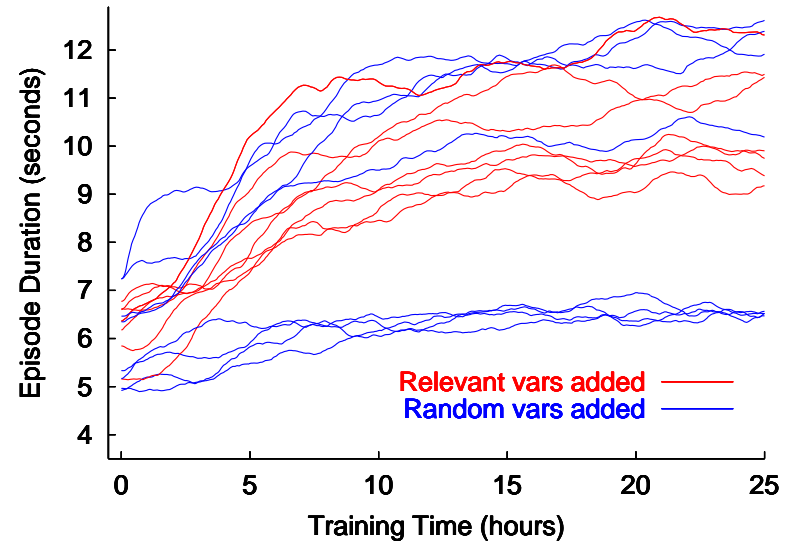
| Field Size | Keeper Policy | |
|------------|---------------|---------------------------|
| | Hand-coded | Learned ($\pm 1\sigma$) |
| 30 × 30 | 19.8 | 18.2 ± 1.1 |
| 25 × 25 | 15.4 | 14.8 ± 0.3 |
| 20 × 20 | 9.6 | 10.4 ± 0.4 |
| 15 × 15 | 6.1 | 7.4 ± 0.9 |
| 10 × 10 | 2.7 | 3.7 ± 0.4 |

- Learning does better on harder problems

Changing the State Representation



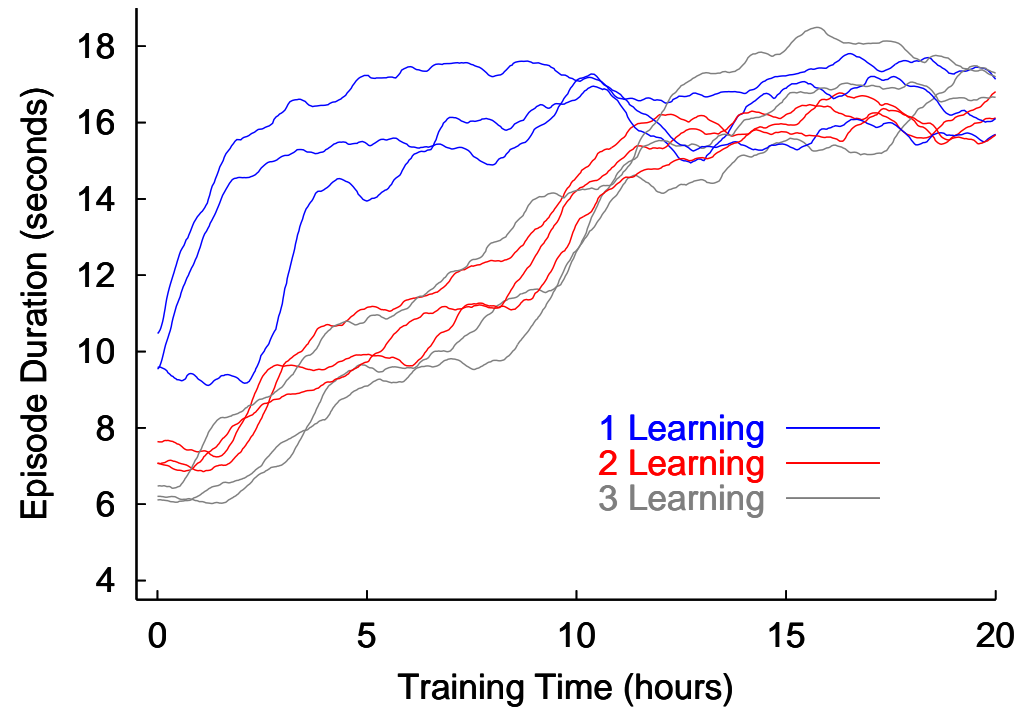
5 variables from the handcoded policy



13 original variable plus an additional 2

- Robust to redundant variables
- Sometimes confused by irrelevant variables

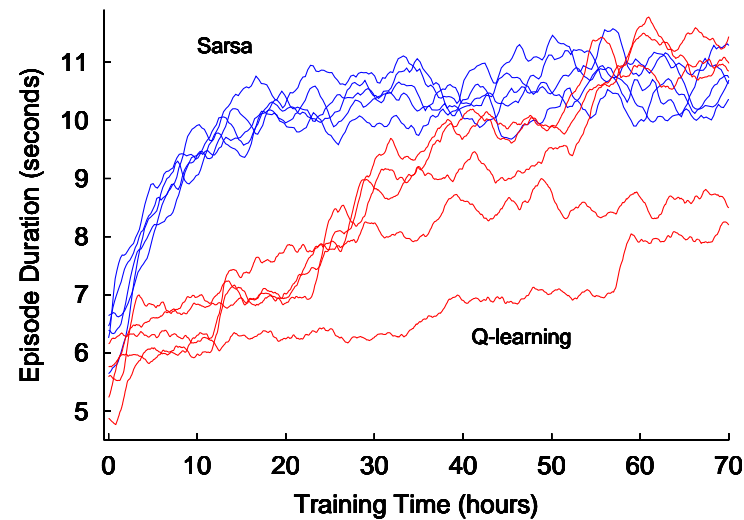
Difficulty of Multiagent Learning



- Multiagent learning is harder!

More Results

- Learns faster than on-policy method: **Q-learning**



Scaling up:

- Solution scales to: **4 vs. 3**, **5 vs. 4**, **6 vs. 5**
- Learning time doubles each step

Conclusion

- SMDP Sarsa(λ) with tile-coding provides a robust multiagent learning solution despite lack of theoretical guarantees.
- Performs as well as a handcoded solution and is more robust.
- Keepaway domain part of official Soccer Server:
<http://sserver.sourceforge.net/>
- **Acknowledgement:** Richard S. Sutton