THE IRISH SOFTWARE
ENGINEERING RESEARCH CENTRE

Multiple Policy Optimization using Collaborative Reinforcement Learning

Ivana Dusparic and Vinny Cahill

DISTRIBUTED SYSTEMS GROUP
TRINITY COLLEGE DUBLIN

May 2007

Outline

- Problem statement
- Reinforcement learning (RL) as a solution?
- Background on RL
  - Collaborative RL (CRL)
  - Multiple policy RL
- MPCRL: Challenges in applying CRL to multiple policy optimization
- Urban traffic control applications
- Future work

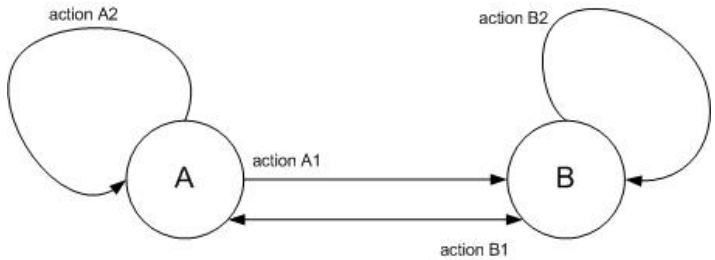
Optimization of Ubiquitous Computing Environments

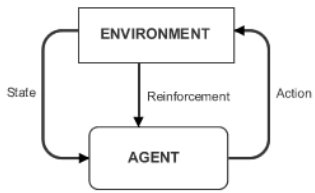
- Ubiquitous computing environments
  - Large-scale, distributed, decentralized, dynamic
  - No global view
  - Multiple goals – conflicting, change over time and space
  - Multiple agents – cooperating to meet various goals

- Self-organizing techniques
  - Global behaviour emerges based on local decisions
  - ACO, evolutionary computing, RL
  - However: typically single goal

Reinforcement Learning Approach

- Originally a single agent, single (implicit) policy, unsupervised learning technique [2]
- Set of states, set of actions, state transition function
- Reinforcement function, Value function – feedback given by the environment
- Learning the best action to perform from each state
- Q-learning – maximize expected long term reward



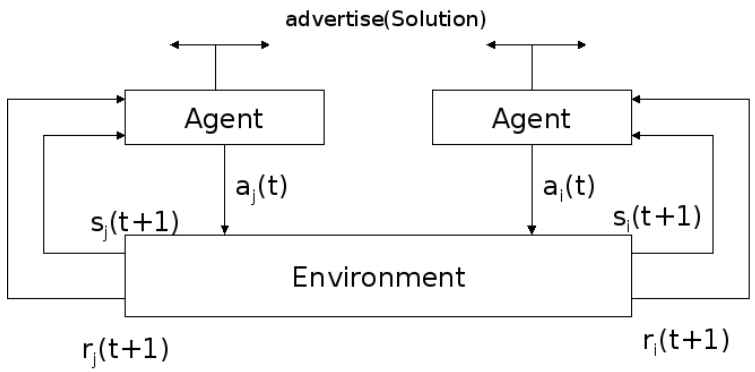


Multiple Policy Reinforcement Learning

- Two approaches:
  - Combined state space:
    - Reduce the joint state space and solve as single policy RL [10]
  - Separate state spaces but common action space:
    - All suggested solutions evaluated and “the best” compromise solution selected (centralized approach) [1]
    - Greatest mass approach (or making sure chosen action is optimal for at least one goal) [8]
    - Choose an action suggested by the agent that will loose the most long term reward if its optimal action is not selected [1]
    - Minimum necessary performance w.r.t. one or more objectives [9]
    - Make sure solution satisfies all objectives, if no such solution then pick a random one [11]

Collaborative Reinforcement Learning (CRL)

- Multi-agent decentralized self-organizing optimization technique [3]
  - Collaborative feedback – exchange of feedback with neighbouring agents
  - Negative feedback – decays local agent's memory of actions/feedback
- Agents learn from the behaviour of other agents
- Agents are homogenous
  - Working together towards a common system goal



Multiple Policy CRL

	Single agent	*Multiple agents (collaboration, decentralisation)*
Single policy	**Reinforcement Learning**	**Collaborative Reinforcement Learning**
Multiple policies (optimise for several goals at a time)	**Multiple policy reinforcement learning**	**MPCRL?** -multiple goals -multiple policy -goals/policies change over time -heterogenous agents -collaborating agents

MPCRL: Research Questions (1 of 2)

- Is RL suitable for multiple policy and multiple agent optimization?
- State space
  - Separate for each policy or combined? How to combine policies of different and of same priorities? Group policies by priorities?
  - State space partitioning – relative or absolute? Does it depend on policy type?
  - Implications for deploying new policies
- Learning
  - Separate for each policy or combined?
  - How to learn for policies that are valid only occasionally?

MPCRL: Research Questions (2 of 2)

- Heterogenous agents
  - Should an agent have knowledge of the policies and priorities of the policies that any agent it exchanges feedback with is implementing?
- Feedback
  - Balancing single agent rewards with feedback from other agents
  - Agents should learn to sacrifice performance w.r.t their local goals if other agents' goals have higher priority
- Implications on the learning process:
  - Time to converge to solution
  - Quality of solution

Application area: Traffic Optimization

- Large-scale decentralized system
- Multiple policy optimization, conflicting goals, several levels, e.g.:
  - Global
    - Maximize global traffic throughput or
    - Minimize global cumulative waiting time for all vehicles in the system
  - Regional
    - Prioritize incoming public transport vehicles
    - Prioritize incoming emergency vehicles
    - Deal with sudden increase of traffic in a given area (public events)
  - Local
    - Each intersection aiming to minimize waiting time for its incoming vehicles

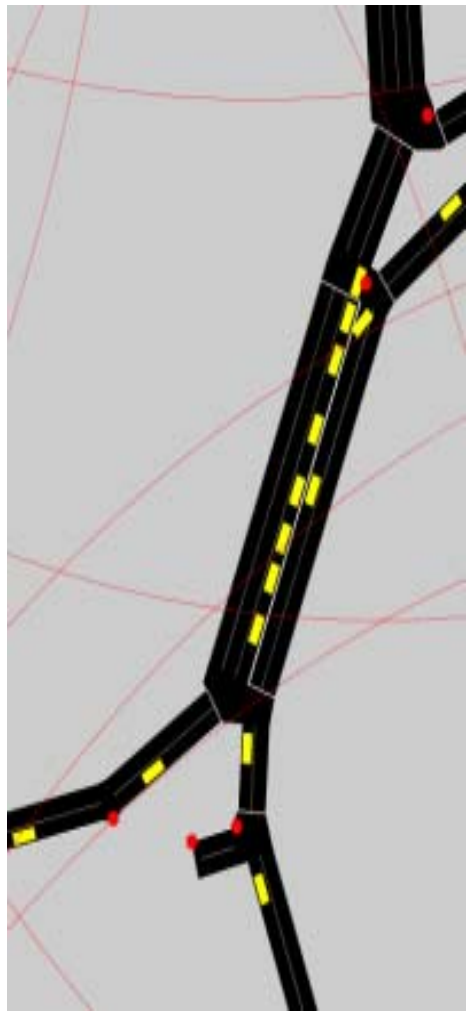
Reinforcement Learning in UTC

- Wiering [5]
  - decision at traffic node based on cars own estimates of time gained by setting the light to green
  - optimisation of travel time – single policy
- Abdulhai [6]
  - single, isolated intersection, decision based on queue length
  - optimisation of travel time – single policy
- Pendrith [4]
  - all the agents updating single shared policy and calculate global reward
  - single policy, centralized
- Cunningham, Cahill, Salkham
  - CRL, single policy based on vehicle arrival rates at approaches

Approach

- Experimental
- Dublin City Traffic Simulator
- Collaborative reinforcement learning framework
- Multiple junctions
- Multiple policies

- Currently:
  - Several junctions
  - Prioritize ambulance
  - Minimize global waiting time
  - Learn to prioritize ambulance



Future work

- Observe how does following influence learning process
  - State space partitioning
  - Combined/separate state spaces
  - Combined/separate learning
  - Actions differing between policies
  - Feedback content exchanged between agents – downstream/upstream junctions
  - Policy priority and scope
  - Feedback between agents implementing different policies
  - Sacrificing performance of the lower priority policies

References

[1] Humphrys, M. Action selection methods using reinforcement learning. In Mataric M. et al, editors, *From Animals to Animats 4: Proceedings of the Fourth International Conference on Simulation of Adaptive Behaviour*, 1996.

[2] Richard S. Suton and Andrew G. Barto. *Reinforcement Learning: An Introduction.* A Bradford Book. The MIT Press, Cambridge, Massachusetts, 2002.

[3] Dowling J., *The Decentralised Coordination of Self-Adaptive Components for Autonomic Distributed Systems*. PhD Thesis, Trinity College Dublin, 2005.

[4] Pendrith M. et al. Distributed reinforcement learning for a traffic engineering application. *AGENTS '00: Proceedings of the fourth international conference on Autonomous agents, 2000.*

[5] Wiering M. Multi-Agent Reinforcement Learning for Traffic Light Control. *Proc. 17th International Conf. on Machine Learning*. 2000

[6] Abdulhai B. et al. Reinforcement Learning for the True Adaptive Traffic Signal Control. *Journal of Transportation Engineering*. 2003

[7] Jim Dowling, Raymond Cunningham, Anthony Harrington, Eoin Curran, and Vinny Cahill. Emergent Consensus in Decentralised Systems Using Collaborative Reinforcement Learning, in *Post-Proceedings of the workshop SELF-STAR: Self-* Properties in Complex Information Systems, Hot Topics in Computer Science*, LNCS, 2005.

[8] N. Sprague and D. Ballard. Multiple-goal reinfrocement learning with modular sarsa. In *International Joint Conference on Artificial Intelligence*, August 2003.

[9] C.R. Shelton. Balancing multiple sources of reward in reinforcement learning. In *NIPS 13*, 2000.

[10] H. Cuayáhuitl, S. Renals, O. Lemon and H. Shimodaira. Learning Multi-Goal Dialogue Strategies Using Reinforcement Learning With Reduced State-Action Spaces. *In Proc. of Interspeech-ICSLP* 2006.

[11] C. E. Mariano and E. F. Morales. Distributed Reinforcement Learning for Multiple Objective Optimization Problems. In *Proceedings of the 2000 Congress on Evolutionary Computation* CEC00.

[12] G. Stevenson, L. Coyle, S. Neely, S. Dobson and P. Nixon. ConStruct – A Decentralised Context Infrastructure for Ubiquitous Computing Environments. In *Proceedings of the IT&T Annual Conference*. 2005.

THANK YOU!

Distributed Systems Group
Trinity College Dublin
Ireland
www.dsg.cs.tcd.ie

Irish Software Engineering Research Centre
www.lero.ie