Cooperative Q-Learning

Lars Blackmore and Steve Block

Multi-Agent Reinforcement Learning: Independent vs. Cooperative Agents
Tan, M
Proceedings of the 10th International Conference on Machine Learning, 1993

Expertness Based Cooperative Q-learning
Ahmadabadi, M.N.; Asadpour, M
IEEE Transactions on Systems, Man and Cybernetics
Part B, Volume 32, Issue 1, Feb. 2002, Pages 66–76

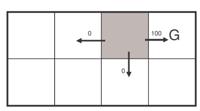
An Extension of Weighted Strategy Sharing in Cooperative Q-Learning for Specialized Agents
Eshgh, S.M.; Ahmadabadi, M.N.
Proceedings of the 9th International Conference on Neural Information Processing, 2002.
Volume 1, Pages 106-110

Overview

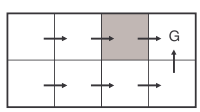
- Single agent reinforcement learning
  - Markov Decision Processes
  - Q-learning
- Cooperative Q-learning
  - Sharing state, sharing experiences and sharing policy
- Sharing policy through Q-values
  - Simple averaging
- Expertness based cooperative Q-learning
  - Expertness measures and weighting strategies
  - Experimental results
- Expertness with specialised agents
  - Scope of specialisation
  - Experimental results

Markov Decision Processes

- Framework
  - States **S**
  - Actions **A**
  - Rewards **R(s,a)**
  - Probabilistic transition Function **T(s,a,s')**



- Goal: find optimal policy **π*(s)** that maximises lifetime reward



Reinforcement Learning

- Want to find π^* through experience
  - Reinforcement Learning
  - Intuitive approach; similar to human and animal learning
  - Use some policy π for motion
  - Converge to the optimal policy π*
- An algorithm for reinforcement learning…

Q-Learning

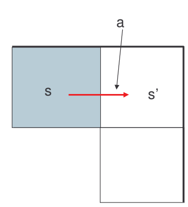
- Define **Q*(s,a):**
  - "Total reward if an agent in state **s** takes action **a**, then acts optimally at all subsequent time steps"
- Optimal policy: **$\pi^*(s)=\mathrm{argmax}_a Q^*(s,a)$**
- **Q(s,a)** is an estimate of **Q*(s,a)**
- Q-learning motion policy: **$\pi(s)=\mathrm{argmax}_a Q(s,a)$**
- Update Q recursively:

$$Q(s,a) = r + \gamma \max_{a'} Q(s',a') \qquad 0 < \gamma < 1$$

Q-learning

- Update Q recursively:

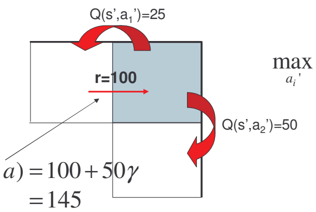
$$Q(s,a) = r + \gamma \max_{a'} Q(s',a')$$



Q-learning

- Update Q recursively:

$$Q(s,a) = r + \gamma \max_{a_i'} Q(s', a_i')$$



$$\max_{a_i'} Q(s', a_i') = 50$$

$$Q(s,a) = 100 + 50\gamma$$
$$= 145$$

Q-Learning

- Define **Q*(s,a):**
  - "Total reward if agent is in state **s**, takes action **a**, then acts optimally at all subsequent time steps"
- Optimal policy: **π*(s)=argmax$_a$Q*(s,a)**
- **Q(s,a)** is an estimate of **Q*(s,a)**
- Q-learning motion policy: **π(s)=argmax$_a$Q(s,a)**
- Update Q recursively:

$$Q(s,a) = r + \gamma \max_{a'} Q(s', a')$$

- Optimality theorem:
  - "If each *(s,a)* pair is updated an infinite number of times, Q converges to Q* with probability 1"

Cooperative Q-Learning

- An example situation:
  - Mobile robots
- Why cooperate?
- Learning framework
  - Individual learning for t_i trials
  - Each trial starts from a random state and ends when robot reaches goal
  - Next, all robots switch to cooperative learning

Cooperative Q-learning

- How should information be shared?
- Three fundamentally different approaches:
  - Expanding state space
  - Sharing experiences
  - Sharing policy through Q-values
- Methods for sharing additional state information and experiences are straightforward
  - These showed some improvement in testing
- Best method for sharing Q-values is not obvious
  - This area offers the greatest challenge and the greatest potential for innovation

Multi-Agent Reinforcement Learning: Independent vs. Cooperative Agents
Tan, M
Proceedings of the 10th International Conference on Machine Learning, 1993

Sharing Q-values

- An obvious approach?
  - Simple Averaging

$$Q_i(s,a) = \frac{1}{n}\sum_{j=1}^{n} Q_j(s,a)$$

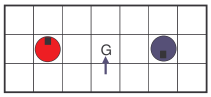
- This was shown to yield some improvement
- What are some of the problems?

Problems with Simple Averaging

- All agents have the same Q table after sharing and hence the same policy:
  - Different policies allow agents to explore the state space differently
- Convergence rate may be reduced

Problems with Simple Averaging

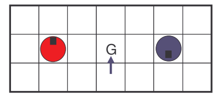
- Convergence rate may be reduced
  - **Without co-operation:**



Trial #	Q(s,a)	
	Agent 1	Agent 2
0	0	0
1	10	0
2	10	10
3	10	10

Problems with Simple Averaging

- Convergence rate may be reduced
  - **With simple averaging:**



Trial #	Q(s,a)	
	Agent 1	Agent 2
0	0	0
1	5	5
2	7.5	7.5
3	8.625	8.625
…	…	…
∞	10	10

Problems with Simple Averaging

- All agents have the same Q table after sharing and hence the same policy:
  - Different policies allow agents to explore the state space differently
- Convergence rate may be reduced
  - Highly problem specific
- Slows adaptation in dynamic environment
- Overall performance is task specific

Expertness

- Idea: value more highly the knowledge of agents who are ‘experts’
  - *Expertness based cooperative Q-learning*
- New Q-sharing equation:

$$Q_i = \sum_{j=1}^{n} W_{ij} \times Q_j$$

- Agent i assigns an importance weight W_{ij} to the Q data held by agent j
- These weights are based on the agents’ relative expertness values e_i and e_j

Expertness Based Cooperative Q-learning
Ahmadabadi, M.N.; Asadpour, M
IEEE Transactions on Systems, Man and Cybernetics
Part B, Volume 32, Issue 1, Feb. 2002, Pages 66–76

Expertness Measures

- Need to define *expertness* of agent i
  - Based on the reinforcement signals agent i has received
- Various definitions:
  - Algebraic Sum $\quad e_i^{NRM} = \sum_t r_i(t)$
  - Absolute Value $\quad e_i^{ABS} = \sum_t |r_i(t)|$
  - Positive $\quad e_i^{P} = \sum_t r_i^{+}$
  - Negative $\quad e_i^{N} = \sum_t r_i^{-}$

Weighting Strategies

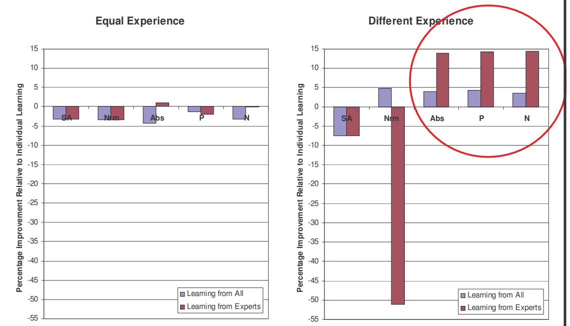
- How do we come up with weights based on the expertnesses?
- Alternative strategies:
  - *‘Learn from all’:* $W_{ij} = \dfrac{e_j}{\sum_{k=1}^{n} e_k}$
  - *‘Learn from experts’:*

$$W_{ij} = \begin{cases} 1-\alpha_i & i = j \\ \alpha_i \dfrac{e_j - e_i}{\sum_{k=1}^{n}(e_k - e_i)} & e_j > e_i \\ 0 & \textit{otherwise} \end{cases}$$

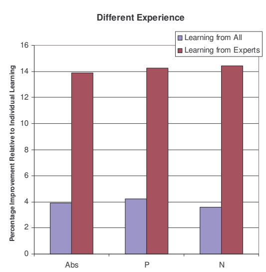
Experimental Setup

- Mobile robots in hunter-prey scenario
- Individual learning phase:
  1. All robots carry out same number of trials
  2. Robots carry out different number of trials
- Followed by cooperative learning
- Parameters to investigate:
  – Cooperative learning vs individual
  – Similar vs different initial expertise levels
  – Different expertness measures
  – Different weight assigning mechanisms
- Performance measured by number of steps

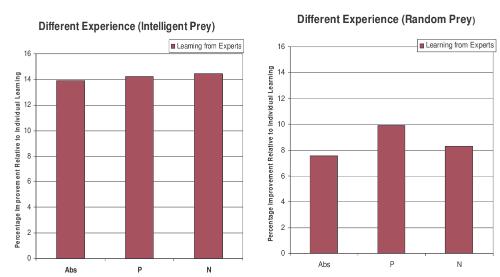
Results



Results



Results

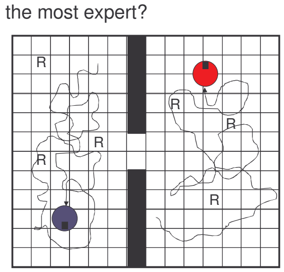


Conclusions

- Without expertness measures, cooperation is detrimental
  – Simple averaging shows decrease in performance
- Expertness based cooperative learning is shown to be superior to individual learning
  – Only true when agents have significantly different expertness values *(necessary but not sufficient)*
- Expertness measures Abs, P and N show best performance
  – Of these three, Abs provides the best compromise
- 'Learning from Experts' weighting strategy shown to be superior to 'Learning from All'

What about this situation?

- Both agents have accumulated the same rewards and punishments
- Which is the most expert?



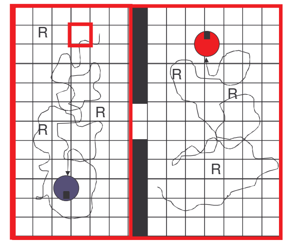
Specialised Agents

- An agent may have explored one area a lot but another area very little
  - The agent is an expert in one area but not in another
- Idea – Specialised agents
  - Agents can be experts in certain areas of the world
  - Learnt policy more valuable if an agent is more expert *in that particular area*

An Extension of Weighted Strategy Sharing in Cooperative Q-Learning for Specialized Agents
Eshgh, S.M.; Ahmadabadi, M.N.
Proceedings of the 9th International Conference on Neural Information Processing, 2002.
Volume 1, Pages 106-110

Specialised Agents

- *Scope of specialisation*
  - *Global*
  - *Local*
  - *State*



Specialised Agents

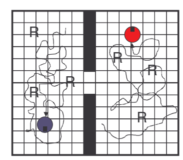
- New Q-sharing equation:

$$Q_{ki} = \sum_{j=1}^{n} W_{ijk} \times Q_{jk}$$

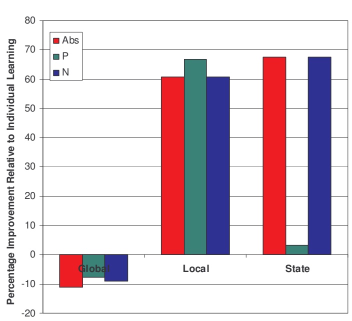
- Agent i assigns an importance weight W_{ijk} to the Q data held by agent j, valid for a region k

Experimental Setup

- Mobile robots in a grid world
- World is approximately segmented into three regions by obstacles
  - One goal per region
- Individual learning followed by cooperative learning as before
- Performance measured by number of steps to reach a goal.



Results



Overall Conclusions

- Expertness based cooperative learning **without** specialised agents can improve performance but can also be detrimental
- Cooperative learning with specialised agents greatly improved performance
- Correct choice of expertness measure is crucial
  - Test case highlights robustness of Abs to problem-specific nature of reinforcement signals