Correlated $Q$-Learning

&

No-Regret $Q$-Learning

Amy Greenwald

with

David Gondek

Keith Hall

City University of New York

Logic and Games Seminar

February 28, 2003
Part I

Multiagent $Q$-Learning

- Correlated-$Q$ Learning
  - converges (empirically) to equilibrium policies

- Nash-$Q$ [Hu and Wellman, 1998]
  - converges (empirically), but not necessarily to equilibrium policies

- Minimax-$Q$ [Littman, 1994]
  - converges (analytically) to equilibrium policies in constant-sum games

AI Agenda  Learn $Q$-Values
Part II

Approximate $Q$-Learning

- No-regret $Q$-learning
  - No external regret learning
    * converges to minimax strategies in constant-sum games
  - No internal regret learning
    * converges to correlated equilibrium in general-sum games

GT Agenda  Learn Equilibria
Markov Decision Processes (MDPs)

Decision Process

- $S$ is a set of states ($s \in S$)
- $A$ is a set of actions ($a \in A$)
- $R : S \times A \rightarrow \mathbb{R}$ is a reward function
- $P[s_{t+1}|s_t, a_t, \ldots, s_0, a_0]$ is a probabilistic transition function that describes transitions between states, conditioned on past states and actions

\[
\text{MDP = Decision Process } + \text{ Markov Property:} \]
\[
P[s_{t+1}|s_t, a_t, \ldots, s_0, a_0] = P[s_{t+1}|s_t, a_t]
\]
\[
\forall t, \forall s_0, \ldots, s_t \in S, \forall a_0, \ldots, a_t \in A
\]
Bellman’s Equations

\[ Q^*(s, a) = R(s, a) + \gamma \sum_{s'} P[s'|s, a]V^*(s') \]

\[ V^*(s) = \max_{a \in A(s)} Q^*(s, a) \]

Value Iteration

<table>
<thead>
<tr>
<th>VALUE ITERATION (MDP, ( \gamma ))</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inputs</td>
</tr>
<tr>
<td>discount factor ( \gamma )</td>
</tr>
<tr>
<td>Output</td>
</tr>
<tr>
<td>optimal state-value function ( V^* )</td>
</tr>
<tr>
<td>optimal action-value function ( Q^* )</td>
</tr>
<tr>
<td>Initialize</td>
</tr>
<tr>
<td>( V = Q = 0 )</td>
</tr>
</tbody>
</table>

```
REPEAT
  for all \( s \in S \)
    for all \( a \in A \)
      \( Q(s, a) = R(s, a) + \gamma \sum_{s'} P[s'|s, a]V(s') \)
      \( V(s) = \max_a Q(s, a) \)
FOREVER
```
**Q-Learning**

\[
\text{Q-LEARNING}(\text{MDP}, \gamma, \alpha, \epsilon)
\]

**Inputs**
- discount factor \(\gamma\)
- rate of averaging \(\alpha\)
- rate of exploration \(\epsilon\)

**Output**
- optimal state-value function \(V^*\)
- optimal action-value function \(Q^*\)

**Initialize**
\(V = Q = 0\)

**REPEAT**
- initialize \(s, a\)
- WHILE \(s\) is nonabsorbing DO
  - simulate action \(a\) in state \(s\)
  - observe reward \(R\) and next state \(s'\)
  - compute \(V(s') = \max_{a \in A(s)} Q(s, a)\)
  - update \(Q(s, a) = (1 - \alpha)Q(s, a) + \alpha[R + \gamma V(s')]\)
  - choose action \(a'\) (on- or off-policy)
  - \(s = s', a = a'\)
  - decay \(\alpha\)

**FOREVER**

**Theorem** [Watkins, 1989]

\(Q\)-learning converges to \(V^*\) and \(Q^*\)
Markov Games

Stochastic Game

- $I$ is a set of $n$ players ($i \in I$)
- $S$ is a set of states ($s \in S$)
- $A_i(s)$ is the $i$th player’s set of actions at state $s$
  let $A(s) = A_1(s) \times \ldots \times A_n(s)$ ($\vec{a} \in A(s)$)
- $P[s_{t+1}|s_t, \vec{a}_t, \ldots, s_0, \vec{a}_0]$ is a probabilistic transition function that describes transitions between states, conditioned on past states and actions
- $R_i(s, \vec{a})$ is the $i$th player’s reward at state $s$ for action vector $\vec{a}$

Markov Game = Stochastic Game + Markov Property:

$$P[s_{t+1}|s_t, \vec{a}_t, \ldots, s_0, \vec{a}_0] = P[s_{t+1}|s_t, \vec{a}_t]$$

$\forall t, \forall s_0, \ldots, s_t \in S, \forall \vec{a}_0, \ldots, \vec{a}_t \in A$
Bellman’s Analogue

\[
Q^*_i(s, \bar{a}) = R_i(s, \bar{a}) + \gamma \sum_{s'} P[s' | s, \bar{a}] V^*_i(s')
\]

**Foe-Q**
\[
V^*_1(s) = \max_{\sigma_1 \in \Sigma_1(s)} \min_{a_2 \in A_2(s)} Q^*_1(s, \sigma_1, a_2) = -V^*_2(s)
\]

**Friend-Q**
\[
V^*_i(s) = \max_{\bar{a} \in A_i(s)} Q^*_i(s, \bar{a})
\]

**Nash-Q**
\[
V^*_i(s) \in \text{Nash}_i(Q^*_1(s), \ldots, Q^*_n(s))
\]

**CE-Q**
\[
V^*_i(s) \in \text{CE}_i(Q^*_1(s), \ldots, Q^*_n(s))
\]
# Multiagent $Q$-Learning

<table>
<thead>
<tr>
<th>MULTI$Q$(MGame, $\gamma, \alpha, \epsilon$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>REPEAT</td>
</tr>
<tr>
<td>initialize $s, a_1, \ldots, a_n$</td>
</tr>
<tr>
<td>WHILE $s$ is nonabsorbing DO</td>
</tr>
<tr>
<td>simulate actions $a_1, \ldots, a_n$ in state $s$</td>
</tr>
<tr>
<td>observe rewards $R_1, \ldots, R_n$ and next state $s'$</td>
</tr>
<tr>
<td>for all $i \in I$</td>
</tr>
<tr>
<td>compute $V_i(s')$</td>
</tr>
<tr>
<td>update $Q_i(s, a_1, \ldots, a_n)$</td>
</tr>
<tr>
<td>(simultaneously) choose actions $a'_1, \ldots, a'_n$</td>
</tr>
<tr>
<td>$s = s'$, $a_1 = a'_1, \ldots, a_n = a'_n$</td>
</tr>
<tr>
<td>decay $\alpha$</td>
</tr>
<tr>
<td>FOREVER</td>
</tr>
</tbody>
</table>

**Nash-$Q$** converges (empirically)
not necessarily to equilibrium policies

**FF-$Q$** converges (analytically)
to equilibrium policies in restricted classes of games

**CE-$Q$** converges (empirically)
to equilibrium policies
Why CE?

- easily computable via linear programming, unlike Nash equilibrium
- players can achieve payoffs outside the convex hull of Nash payoffs [Aumann, 74]
- players learn correlated equilibrium via no-regret algorithms [Foster & Vohra, 99]
- consistent with the usual AI view of individually rational behavior

Why NOT (Nash or) CE?

- equilibrium selection problem
Correlated Equilibrium

<table>
<thead>
<tr>
<th>Chicken</th>
<th>( L )</th>
<th>( R )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( T )</td>
<td>6,6</td>
<td>2,7</td>
</tr>
<tr>
<td>( B )</td>
<td>7,2</td>
<td>0,0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CE</th>
<th>( L )</th>
<th>( R )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( T )</td>
<td>1/2</td>
<td>1/4</td>
</tr>
<tr>
<td>( B )</td>
<td>1/4</td>
<td>0</td>
</tr>
</tbody>
</table>

\[
\text{max} 12\pi_{TL} + 9\pi_{TR} + 9\pi_{BL} + 0\pi_{BR}
\]

subject to probability constraints

\[
\pi_{TL} + \pi_{TR} + \pi_{BL} + \pi_{BR} = 1
\]

\[
\pi_{TL}, \pi_{TR}, \pi_{BL}, \pi_{BR} \geq 0
\]

& individual rationality constraints

\[
6\pi_{LT} + 2\pi_{RT} \geq 7\pi_{LT} + 0\pi_{RT}
\]
\[
7\pi_{LB} + 0\pi_{RB} \geq 6\pi_{LB} + 2\pi_{RB}
\]
\[
6\pi_{TL} + 2\pi_{BL} \geq 7\pi_{TL} + 0\pi_{BL}
\]
\[
7\pi_{TR} + 0\pi_{BR} \geq 6\pi_{TR} + 2\pi_{BR}
\]
\[
CE_i(Q_1(s), \ldots, Q_n(s)) = \left\{ \sum_{\bar{a} \in A} \sigma^*(\bar{a})Q_i(s, \bar{a}) \mid \sigma^* \text{ satisfies Eq. 1, 2, 3, or 4} \right\}
\]

- **Utilitarian** maximize the sum of rewards

\[
\sigma^* \in \arg \max_{\sigma \in CE} \sum_{i \in I} \sum_{\bar{a} \in A} \sigma(\bar{a})Q_i(s, \bar{a}) \tag{1}
\]

- **Egalitarian** maximize the minimum reward

\[
\sigma^* \in \arg \max_{\sigma \in CE} \min_{i \in I} \sum_{\bar{a} \in A} \sigma(\bar{a})Q_i(s, \bar{a}) \tag{2}
\]

- **Republican** maximize the maximum reward

\[
\sigma^* \in \arg \max_{\sigma \in CE} \max_{i \in I} \sum_{\bar{a} \in A} \sigma(\bar{a})Q_i(s, \bar{a}) \tag{3}
\]

- **Libertarian** \(i\) maximizes only \(i\)'s rewards

let \(\sigma^* = \prod_i \sigma^i\) where

\[
\sigma^i \in \arg \max_{\sigma \in CE} \sum_{\bar{a} \in A} \sigma(\bar{a})Q_i(s, \bar{a}) \tag{4}
\]
Grid Games

GG1

GG2

GG3

Grid Game 1

Grid Game 2

Grid Game 3
Equilibrium Policies

<table>
<thead>
<tr>
<th>Grid Games</th>
<th>Algorithm</th>
<th>GG1</th>
<th>GG2</th>
<th>GG3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Score</td>
<td>Games</td>
<td>Score</td>
<td>Games</td>
</tr>
<tr>
<td>$Q$</td>
<td>100,100</td>
<td>2500</td>
<td>49,100</td>
<td>3333</td>
</tr>
<tr>
<td>Foe-$Q$</td>
<td>0,0</td>
<td>0</td>
<td>67,68</td>
<td>3003</td>
</tr>
<tr>
<td>Friend-$Q$</td>
<td>$-10^4,-10^4$</td>
<td>0</td>
<td>$-10^4,-10^4$</td>
<td>0</td>
</tr>
<tr>
<td>$uCE-Q$</td>
<td>100,100</td>
<td>2500</td>
<td>50,100</td>
<td>3333</td>
</tr>
<tr>
<td>$eCE-Q$</td>
<td>100,100</td>
<td>2500</td>
<td>51,100</td>
<td>3333</td>
</tr>
<tr>
<td>$rCE-Q$</td>
<td>100,100</td>
<td>2500</td>
<td>100,49</td>
<td>3333</td>
</tr>
<tr>
<td>$lCE-Q$</td>
<td>100,100</td>
<td>2500</td>
<td>100,51</td>
<td>3333</td>
</tr>
</tbody>
</table>
Part I

Correlated-\(Q\) Learning

- good news
  - converges (empirically) to an equilibrium policy

- bad news
  - equilibrium policy is path dependent
    \textit{i.e.}, dynamics are nonergodic

Part II

No-regret \(Q\)-Learning

- \textbf{No-external-regret}
  - converge to minimax strategies in constant-sum games

- \textbf{No-internal-regret}
  - converge to correlated equilibrium in general-sum games
Repeated Games

A game is a tuple $\Gamma = (I, (A_i, R_i)_{i \in I})$ where

- $I$ is a set of players ($i \in I$)
- $A_i$ is a set of pure actions ($a_i \in A_i$)
- $R_i : A \to \mathbb{R}$ is a reward function ($a \in A = \prod_i A_i$)

A repeated game is a sequence of tuples $\Gamma^T$ or $\Gamma^\infty$
**No-Regret Definitions**

Regret is the difference in rewards for playing action $a'_i$ rather than $a_i$ at time $t$:

$$\rho_i^t(a_i, a'_i) = \pi_i^t(a'_i) [R_i(a_i, a_{-i}^t) - R_i(a'_i, a_{-i}^t)]$$

A learning algorithm exhibits no-external-regret iff it generates weights $\{\pi_i^t\}$ s.t. for all opposing policies, there exists $T$ s.t. for all $T > T_0$,

$$\max_{a_i \in A_i} \frac{1}{T} \sum_{t=1}^{T} \sum_{a'_i \in A_i} \pi_i^t(a'_i) \rho_i^t(a_i, a'_i) \leq \text{ERR}(T)$$

where $\text{ERR}(T) \to 0$ as $T \to \infty$.

A learning algorithm exhibits no-internal-regret iff it generates weights $\{\pi_i^t\}$ s.t. for all opposing policies, there exists $T$ s.t. for all $T > T_0$,

$$\max_{a_i \in A_i} \frac{1}{T} \sum_{a'_i \in A_i} \left( \sum_{t=1}^{T} \pi_i^t(a'_i) \rho_i^t(a_i, a'_i) \right)^+ \leq \text{ERR}(T)$$

where $\text{ERR}(T) \to 0$ as $T \to \infty$ and $X^+ = \max\{X, 0\}$.  

16
No-Regret Algorithms

\[ R^t_i(a'_i, a_i) = \frac{1}{t} \sum_{x=1}^{t} p^x_i(a'_i, a_i) \]

\[ X^t_i(a_i) = \sum_{a'_i \in A_i} R^t_i(a'_i, a_i) \quad \mathcal{W}^t_i(a_i) = \sum_{a'_i \in A_i} \pi^t_i(a'_i) R^t_i(a'_i, a_i) \]

Hart and Mas-Colell (HMC)

\[ \pi^{t+1}_i(a_i) = \frac{[X^t_i(a_i)]^+}{\sum_{a_i \in A_i} [X^t_i(a_i)]^+} \]

NER learning approximates minimax equilibria
[Freund and Schapire, 1996]

WAR Weighted Average Regret (WAR)

\[ \pi^{t+1}_i(a_i) = \frac{[\mathcal{W}^t_i(a_i)]^+}{\sum_{a_i \in A_i} [\mathcal{W}^t_i(a_i)]^+} \]

NIR learning approximates correlated equilibria
[Foster and Vohra, 1997]
Normal Form Games

Matching Pennies

<table>
<thead>
<tr>
<th></th>
<th>H</th>
<th>T</th>
</tr>
</thead>
<tbody>
<tr>
<td>H</td>
<td>1,0</td>
<td>0,1</td>
</tr>
<tr>
<td>T</td>
<td>0,1</td>
<td>1,0</td>
</tr>
</tbody>
</table>

Shapley Game

<table>
<thead>
<tr>
<th></th>
<th>L</th>
<th>C</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>1,0</td>
<td>0,1</td>
<td>0,0</td>
</tr>
<tr>
<td>M</td>
<td>0,0</td>
<td>1,0</td>
<td>0,1</td>
</tr>
<tr>
<td>B</td>
<td>0,1</td>
<td>0,0</td>
<td>1,0</td>
</tr>
</tbody>
</table>
Matching Pennies

External Regret: HMC

External Regret: WAR
Matching Pennies

Internal Regret: HMC

Internal Regret: WAR
Matching Pennies

Frequencies: HMC

Hart and Mas-Colell: Matching Pennies

Frequencies: WAR

WAR: Matching Pennies
Rochambeau

External Regret: HMC

Hart and Mas-Colell: Rock, Paper, Scissors

External Regret: WAR

WAR: Rock, Paper, Scissors
Rochambeau

Internal Regret: HMC

Internal Regret: WAR
Rochambeau

Frequencies: HMC

Hart and Mas-Colell: Rock, Paper, Scissors

Frequencies: WAR

WAR: Rock, Paper, Scissors
Shapley Game

External Regret: HMC

External Regret: WAR
Shapley Game

Internal Regret: HMC

Hart and Mas–Colell: Shapley Game

Internal Regret: WAR

WAR: Shapley Game
Shapley Game

Frequencies: HMC

Hart and Mas-Colell: Shapley Game

Frequencies: WAR

WAR: Shapley Game
Naive No-Regret Learning

\[
\hat{R}_i(a_i, a^t_{-i}) = \begin{cases} 
\frac{R_i(a_i, a^t_i)}{\hat{\pi}_i(a_i)} & \text{if } a^t_i = a_i \\
0 & \text{otherwise}
\end{cases}
\]

\[
\hat{\pi}_i^t = (1 - \epsilon)\pi_i^t + \frac{\epsilon}{|A_i|}
\]

Theorem

If an informed learning algorithm \(A_i\) exhibits no-regret, then the naive learning algorithm \(\tilde{A}_i\) exhibits \(\epsilon\)-no-regret.
No-Regret $Q$-Learning

NRQ(MGame, $\gamma, \alpha, \epsilon$)

Inputs
- discount factor $\gamma$
- rate of averaging $\alpha$
- rate of exploration $\epsilon$

Output
- equilibrium state-value function $V^*$
- equilibrium action-value function $Q^*$

Initialize $V = Q = 0$

REPEAT
- initialize $s, a_1, \ldots, a_n$
- WHILE $s$ is nonterminal DO
  - simulate actions $a_1, \ldots, a_n$ in state $s$
  - observe rewards $R_1, \ldots, R_n$ and next state $s'$

  for all $i \in I$
  - let $\tilde{\pi}(s', \tilde{a}) = \prod_i \tilde{\pi}_i(s', a_i)$
  - compute $V_i(s') = \sum_{\tilde{a} \in A} \tilde{\pi}(s', \tilde{a})Q_i(s', \tilde{a})$
  - update $Q_i(s, \tilde{a}) = (1 - \alpha)Q_i(s, \tilde{a}) + \alpha P_i(s, a_i)$
    with $P_i(s, a_i) = R_i + \gamma V_i(s')$
  - update policies
    - informed
    - naive

(simultaneously) choose actions $a'_1, \ldots, a'_n$
- $s = s'$, $a_1 = a'_1, \ldots, a_n = a'_n$
- decay $\alpha$

FOREVER
Part I

Correlated-\(Q\) Learning

- good news
  - converges (empirically) to an equilibrium policy

- bad news
  - equilibrium policy is path dependent

Part II

No-Regret-\(Q\) Learning

- conjectures
  - \textit{WAR} and \textit{PEACE} exhibit no-internal regret
  - \textit{NER Q-Learning} converges to minimax strategies in constant-sum Markov games
  - \textit{NIR Q-Learning} converges to correlated equilibrium in general-sum Markov games
Marty’s Game

Rewards

<table>
<thead>
<tr>
<th></th>
<th>U</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.25,1</td>
<td>0,0</td>
</tr>
<tr>
<td></td>
<td>1,0</td>
<td>0,0.25</td>
</tr>
</tbody>
</table>

Q-Values

<table>
<thead>
<tr>
<th></th>
<th>U</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1,2</td>
<td>1,0.75</td>
</tr>
<tr>
<td></td>
<td>1.75,1</td>
<td>1,1</td>
</tr>
</tbody>
</table>

Unique Mixed Strategy Equilibrium

\( \pi_r(U) = \frac{7}{15} \) and \( \pi_c(L) = \frac{4}{9} \)