the case for

Learning Correlated Equilibrium in Markov Games

Amy Greenwald

Brown University

Friday the 13th of December 2002

Why Correlated Equilibrium?

- easily computable via linear programming, unlike Nash equilibri
- players can achieve payoffs outside the convex hull of Nash pay
- players learn correlated equilibrium via no-regret algorithms [Fo
- consistent with the usual AI view of individually rational behav

Why NOT (Nash or) Correlated Equilibrium?

- equilibrium selection problem

Correlated Equilibrium

Chicken

	L	R
T	6,6	2,7
B	7,2	0,0

CE

	L	R
T	1/2	1/4
B	1/4	0

probability constraints

$$\pi_{TL} + \pi_{TR} + \pi_{BL} + \pi_{BR} = 1$$

$$\pi_{TL}, \pi_{TR}, \pi_{BL}, \pi_{BR} \geq 0$$

individual rationality constraints

$$\begin{aligned} 6\pi_{L|T} + 2\pi_{R|T} &\geq 7\pi_{L|T} + 0\pi_{R|T} \\ 7\pi_{L|B} + 0\pi_{R|B} &\geq 6\pi_{L|B} + 2\pi_{R|B} \\ 6\pi_{T|L} + 2\pi_{B|L} &\geq 7\pi_{T|L} + 0\pi_{B|L} \\ 7\pi_{T|R} + 0\pi_{B|R} &\geq 6\pi_{T|R} + 2\pi_{B|R} \end{aligned}$$

Part I

Multiagent Q-Learning

- Correlated-Q Learning
  - converges (empirically) to equilibrium policies
- Nash-Q [Hu and Wellman, 1998]
  - converges (empirically), perhaps not to equilibrium policies
- Minimax-Q [Littman, 1994]
  - converges (analytically), to equilibrium policies in zero-sum

AI Agenda Learn Q-values

Part II

Approximate Q-Learning

- No-regret Q-learning
  - No-external-regret
    - converge to minimax strategies
      in constant-sum games
  - No-internal-regret
    - converge 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

MDP = Decision Process + Markov Property:

$$P[s_{t+1}|s_t, a_t, \ldots, s_0, a_0] = P[s_{t+1}|s_t, a_t]$$

Bellman's Equations

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

$$V(s) = \max_{a \in A(s)} Q(s,a)$$

Theorem

There exist Q^* and V^* that satisfy this system of equati

Q-Learning

Q_LEARNING(MDP, γ, α)

Inputs	discount factor γ
	rate of averaging α
Output	optimal state-value function V^*
	optimal action-value function Q^*
Initialize	arbitrary V, Q, initial state-action pair s, a

REPEAT

simulate action a in state s

observe reward R, 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 α

FOREVER

Theorem [Watkins, 1989] Q-learning converges to V^*

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 ith 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 ith player's reward at state s for action vector $\vec{a}$

Markov Game = Stochastic Game + Markov Property:

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

Bellman's Analogue

$$Q_i(s,\vec{a}) = R_i(s,\vec{a}) + \gamma \sum_{s'} P[s'|s,\vec{a}]V_i(s')$$

Foe-Q $\quad V_1(s) = \max_{\sigma_1 \in \Sigma_1(s)} \min_{a_2 \in A_2(s)} Q_1(s,\sigma_1,a_2) = -V_2(s$

Friend-Q $\quad V_i(s) = \max_{\vec{a} \in A(s)} Q_i(s,\vec{a})$

Nash-Q $\quad V_i(s) \in \text{Nash}_i(Q_1(s),\ldots,Q_n(s))$

CE-Q $\quad V_i(s) \in \text{CE}_i(Q_1(s),\ldots,Q_n(s))$

Theorem [Fink 64, Mertens 02, Greenwald 02]

There exist Q^* and V^* that satisfy each system of equa

Multiagent Q-Learning

MULTIQ(MGame, $\gamma, \alpha, \bigoplus$)

REPEAT

simulate actions $a_1, \ldots, a_n$ in state s

observe rewards $R_1, \ldots, R_n$ and next state s'

for all $i \in I$

$V_i(s') \in \bigoplus(Q_1, \ldots, Q_n)$

$Q_i(s, a_1, \ldots, a_n) = (1 - \alpha)Q_i(s, a_1, \ldots, a_n) + \alpha[R_i(s, a_1, \ldots, a_n)$

choose actions $a'_1, \ldots, a'_n$

$s = s'$, $a_1 = a'_1, \ldots, a_n = a'_n$

decay α

FOREVER

FF-Q converges to equilibrium policies in zero-sum gam

Nash-Q converges empirically, perhaps not to equilibriun

CE-Q converges empirically to equilibrium policies

Correlated Equilibrium Selection

$\mathrm{CE}_i(Q_1(s), \ldots, Q_n(s)) = \{\sum_{\vec{a} \in A} \sigma^*(\vec{a}) Q_i(s, \vec{a}) \; | \sigma^*$ satisfies Eq. 1, 2, 3,

- Utilitarian maximize the sum of values

$$\sigma^* \in \arg\max_{\sigma \in \mathrm{CE}} \sum_{\vec{a} \in A} \sum_{i \in I} \sigma(\vec{a}) Q_i(s, \vec{a})$$

- Egalitarian maximize the minimum value

$$\sigma^* \in \arg\max_{\sigma \in \mathrm{CE}} \sum_{\vec{a} \in A} \min_{i \in I} \sigma(\vec{a}) Q_i(s, \vec{a})$$

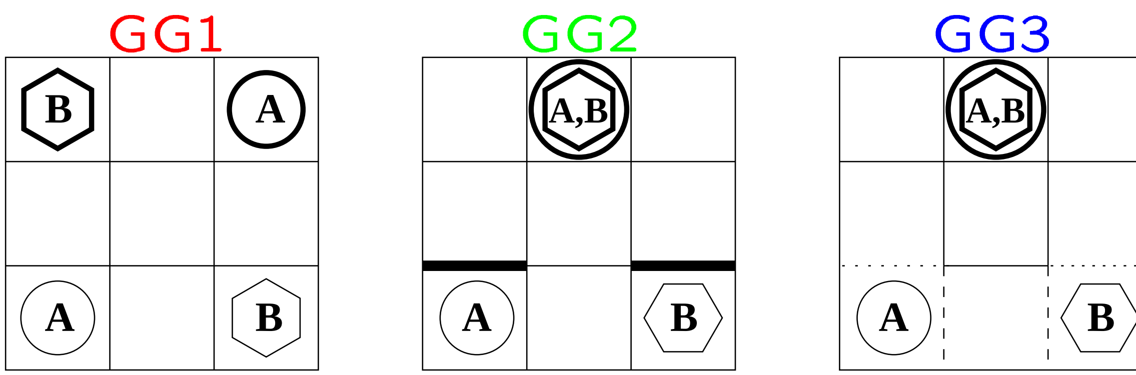
- Republican maximize the maximum value

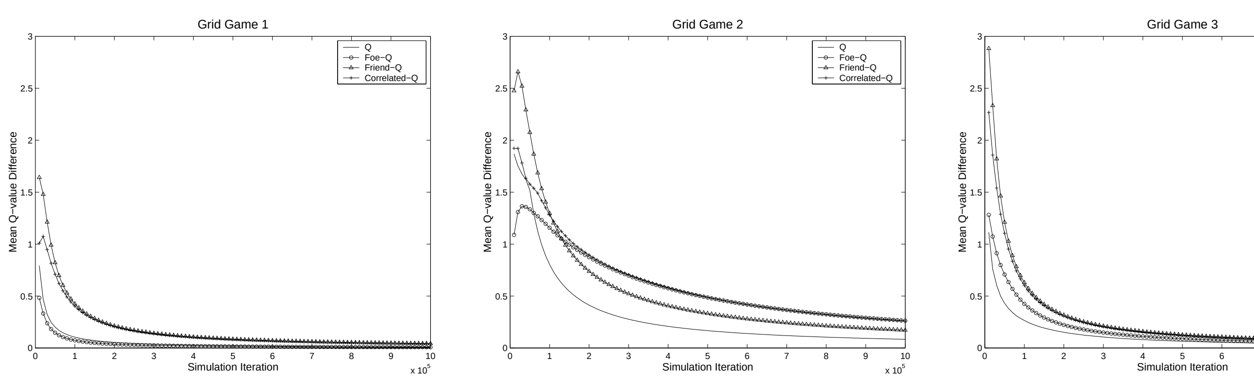
$$\sigma^* \in \arg\max_{\sigma \in \mathrm{CE}} \sum_{\vec{a} \in A} \max_{i \in I} \sigma(\vec{a}) Q_i(s, \vec{a})$$

- Libertarian i maximizes only i's value: $\sigma^* = \prod_i \sigma^i$, where

$$\sigma^i \in \arg\max_{\sigma \in \mathrm{CE}} \sum_{\vec{a} \in A} \sigma(\vec{a}) Q_i(s, \vec{a})$$

Grid Games


GG1
B
A
A
B
GG2
A,B
A
B
GG3
A,B
A
B

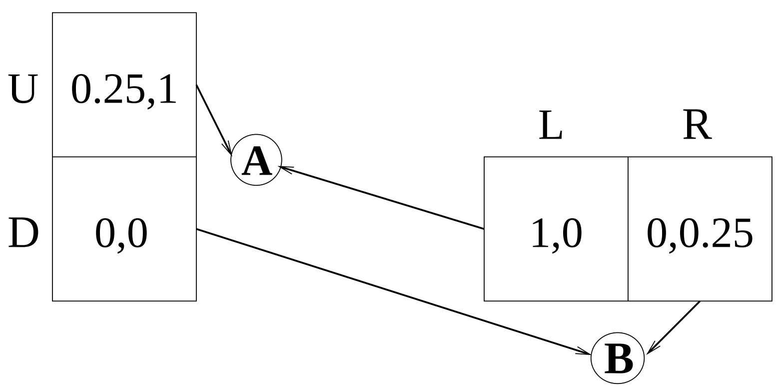

Grid Game 1
Mean Q−value Difference
Simulation Iteration
x 10^5
Q
Foe−Q
Friend−Q
Correlated−Q
Grid Game 2
Mean Q−value Difference
Simulation Iteration
x 10^5
Q
Foe−Q
Friend−Q
Correlated−Q
Grid Game 3
Mean Q−value Difference
Simulation Iteration

Equilibrium Policies

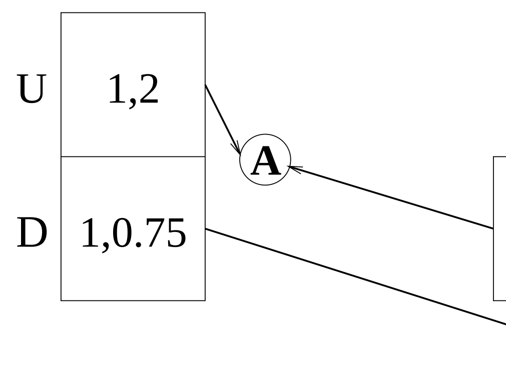
Grid Games	GG1		GG2		
Algorithm	Score	Games	Score	Games	Sc
Q	100,100	2500	49,100	3333	100
Foe-Q	0,0	0	67,68	3003	120
Friend-Q	$-10^4, -10^4$	0	$-10^4, -10^4$	0	-10^4
uCE-Q	100,100	2500	50,100	3333	116
eCE-Q	100,100	2500	51,100	3333	117
rCE-Q	100,100	2500	100,49	3333	125
lCE-Q	100,100	2500	100,51	3333	-10^4

Marty's Game

Rewards



Q-Values



Unique Mixed Strategy Equilibrium

$\pi_1(U) = 7/15$ and $\pi_2(L) = 4/9$

Conjectures

- NER Q-Learning converges to minimax strategies in constant-su
- NIR Q-Learning converges to correlated equilibrium in general-su