Convergent Gradient Ascent in General-Sum Games

Bikramjit Banerjee and Jing Peng

Electrical Engineering and Computer Science Department, Tulane University,
New Orleans, LA 70118, USA
{banerjee, jp}@eecs.tulane.edu
http://www.eecs.tulane.edu/Peng

Abstract. In this work we look at the recent results in policy gradient learning in a general-sum game scenario, in the form of two algorithms, IGA and WoLF-IGA. We address the drawbacks in convergence properties of these algorithms, and propose a more accurate version of WoLF-IGA that is guaranteed to converge to Nash Equilibrium policies in self-play (or against an IGA learner). We also present a control theoretic interpretation of variable learning rate which not only justifies WoLF-IGA, but also shows it to achieve fastest convergence under some constraints. Finally we derive optimal learning rates for fastest convergence in practical simulations.

1 Introduction

Game theory has been a driving impetus for modeling concurrent reinforcement learning problems. With booming e-commerce (notwithstanding the recent slump), the day is not far when automated buyers and sellers will control the electronic economy. There are other potential applications like disaster relief by robots(in potentially hazardous environments, especially after September 11), automated and robotic control of applications (viz. ranging from households to Mars exploration), etc where coordination among multiple agents will hold the key. This makes the focus on multiagent learning research extremely timely and justified. Several algorithms for multiagent learning have been proposed[5],[4],[3],[1], mostly guaranteed to converge to equilibrium in the limit. Bowling and Veloso note in [2], that none of these methods simultaneously satisfies rationality and convergence, two of the desirable criteria for any multiagent learning algorithm.

A recent work [8] demonstrated that policy gradient ascent (which they called "Infinitesimal Gradient Ascent" or IGA) could achieve convergence to either Nash Equilibrium policies or Nash Equilibrium payoffs (when the policies don't converge) in self-play. This algorithm was rational [2] but not convergent to Equilibrium policies in all general-sum games. Subsequently, it was modified with a variable learning rate [2] but the resulting algorithm (WoLF-IGA) might still not converge in some general-sum games. The rationale behind *WoLF* (*W*in *o*r

*L*earn *F*ast) was to allow the opponent to adapt to the learner's policy by learning slowly (i.e. changing its policy slowly) when the learner is "winning", but learn fast when it is not "winning". However, they [2] used an approximate criterion to determine when the learner was "winning", and consequently could fail to converge in some general-sum games. The motivation of the present work is to fill this gap to produce the first multiagent learning algorithm that is both rational and universally convergent. We then proceed to analyse why learning at a variable rate is essential for convergence in policy gradient learning, and show that WoLF-IGA is the best we can do under some constraints. Finally we address the question of how should we choose the learning rates so that we can achieve fastest convergence.

2 Definitions

Here we provide definitions of key concepts for our work. We refer to A_i as the set of possible actions available to the ith agent.

Definition 1 *A bimatrix game is given by a pair of matrices,* (M_1, M_2), *(each of size* $|A_1| \times |A_2|$ *for a two-agent game) where the payoff of the ith agent for the joint action* (a_1, a_2) *is given by the entry* $M_k(a_1, a_2)$, $\forall (a_1, a_2) \in A_1 \times A_2,\ k = 1, 2$.

A *constant-sum game* (also called competitive games) is a special bimatrix game where $M_1(a_1, a_2) + M_2(a_1, a_2) = c,\ \forall (a_1, a_2) \in A_1 \times A_2$, where c is a constant. If $c = 0$, then it is also called a zero-sum game.

Definition 2 *A mixed-strategy Nash Equilibrium for a bimatrix game* (M_1, M_2) *is a pair of probability vectors* (π_1^*, π_2^*) *such that*

$$\pi_1^{*T} M_1 \pi_2^* \geq \pi_1^T M_1 \pi_2^* \quad \forall \pi_1 \in PD(A_1).$$

$$\pi_1^{*T} M_2 \pi_2^* \geq \pi_1^{*T} M_2 \pi_2 \quad \forall \pi_2 \in PD(A_2).$$

where $PD(A_i)$ *is the set of probability-distributions over the ith agent's action space.*

No player in this game has any incentive for unilateral deviation from the Nash equilibrium strategy, given the other's strategy. There always exists at least one such equilibrium profile for an arbitrary finite bimatrix game [6].

3 Policy Gradient Ascent in Bimatrix Games

The basic idea of such an algorithm is to iteratively update an agent's strategy based on the consequent improvement in the agent's expected payoff. When both the agent's fail to improve their strategies any further (which may never happen), they must have converged to some Nash Equilibrium of the game [8]. The simplified domain for studying this problem is a two-agent, two action scenario, with the following payoff matrices

$$R = \begin{bmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{bmatrix} \text{ and } C = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix}.$$

Let α and β denote the probablities of the two agents selecting the first actions from their respective sets of available actions. Then, the expected payoffs of the two agents, (V_r for the row agent, and V_c for the column agent) are given by

$$V_r(\alpha, \beta) = r_{11}(\alpha\beta) + r_{22}((1-\alpha)(1-\beta)) + r_{12}(\alpha(1-\beta)) + r_{21}((1-\alpha)\beta)$$

$$V_c(\alpha, \beta) = c_{11}(\alpha\beta) + c_{22}((1-\alpha)(1-\beta)) + c_{12}(\alpha(1-\beta)) + c_{21}((1-\alpha)\beta)$$

Then given a strategy pair (α, β) (constrained to lie in the unit square), and letting $u = (r_{11}+r_{22})-(r_{21}+r_{12})$ and $u' = (c_{11}+c_{22})-(c_{12}+c_{21})$, the gradients are given by

$$\frac{\partial V_r(\alpha_k, \beta_k)}{\partial \alpha} = \beta u - (r_{22} - r_{12}) \tag{1}$$

$$\frac{\partial V_c(\alpha_k, \beta_k)}{\partial \beta} = \alpha u' - (c_{22} - c_{21}) \tag{2}$$

and the strategy pair can be updated as

$$\alpha_{k+1} = \alpha_k + \eta \frac{\partial V_r(\alpha_k, \beta_k)}{\partial \alpha}$$

$$\beta_{k+1} = \beta_k + \eta \frac{\partial V_c(\alpha_k, \beta_k)}{\partial \beta} \tag{3}$$

The new gradients generated by the above rules are constrained to lie in the unit square by taking their projections on the boundary whenever they cross out. For $\eta \to 0$, the algorithm is called *infinitesimal gradient ascent* or IGA. It is known from game theory [7], that the algorithm may never converge in (α, β), but their expected payoffs have been proved to always converge to that of some Nash pair[8].

4 WoLF-IGA

Using equations 1,2,3 and $\eta \to 0$, we get the unconstrained dynamics of the strategy pair given by the differential equations

$$\begin{bmatrix} \frac{\partial \alpha}{\partial t} \\ \frac{\partial \beta}{\partial t} \end{bmatrix} = \begin{bmatrix} 0 & u \\ u' & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} + \begin{bmatrix} -(r_{22} - r_{12}) \\ -(c_{22} - c_{21}) \end{bmatrix}. \tag{4}$$

It has been proved [8] that the points of zero gradients (projected) are Nash Equilibria. However, the algorithm may not converge to such a point in case the matrix $U = \begin{bmatrix} 0 & u \\ u' & 0 \end{bmatrix}$ has imaginary eigenvalues and the center (point of zero gradient) lies within the unit square. Consequently, the algorithm was modified with a variable learning rate[2] to converge to Nash pair even in this remaining subcase.

The notion of variable learning rate changes the update rules in 3 to

$$\alpha_{k+1} = \alpha_k + \eta l_k^r \frac{\partial V_r(\alpha_k, \beta_k)}{\partial \alpha}$$

$$\beta_{k+1} = \beta_k + \eta l_k^c \frac{\partial V_c(\alpha_k, \beta_k)}{\partial \beta} \tag{5}$$

where $l_k^{r,c} \in \{l_{min}, l_{max}\}$. Since the proof of convergence of IGA for the other subcases depends only on the sign of the gradients, the above learning rules have the same convergence properties as long as $l_{min}, l_{max} > 0$. Moreover, for $l_{min} < l_{max}$, the algorithm can be made to converge to Nash pair in the remaining subcase[2], by choosing

$$l^r(t) = \begin{cases} l_{min} \text{ when } V_r(\alpha_t, \beta_t) \geq V_r(\alpha^e, \beta_t) \\ l_{max} \text{ otherwise} \end{cases}$$

$$l^c(t) = \begin{cases} l_{min} \text{ when } V_c(\alpha_t, \beta_t) \geq V_c(\alpha_t, \beta^e) \\ l_{max} \text{ otherwise} \end{cases} \tag{6}$$

where (α^e, β^e) are some Nash pair. The unconstrained dynamics of the system now follows the differential equations

$$\begin{bmatrix} \frac{\partial \alpha}{\partial t} \\ \frac{\partial \beta}{\partial t} \end{bmatrix} = \begin{bmatrix} 0 & l^r(t)u \\ l^c(t)u' & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} + \begin{bmatrix} -l^r(t)(r_{22} - r_{12}) \\ -l^c(t)(c_{22} - c_{21}) \end{bmatrix}. \tag{7}$$

The algorithm given by equations 5,6 is called the WoLF-IGA. When $V_r(\alpha_t, \beta_t) \geq V_r(\alpha^e, \beta_t)$, the row agent is said to be *winning*, in the sense that it would prefer its current strategy to its Nash strategy against the opponent's current strategy. However, in order to find out whether an agent is *winning*, Bowling et.al.[2] used an approximate criterion for winning, defined by $V_r(\alpha_t, \beta_t) \geq V_r(\alpha^e, \beta^e)$, since $V_r(\alpha^e, \beta_t)$ cannot be directly computed without knowing α^e. As they note, this criterion fails to guarantee convergence in some general-sum games. In the next section we derive an alternative criterion, that can be easily computed and guarantees convergence.

5 Accurate Criterion for *Winning*

We are concerned with the subcase where U has purely imaginary eigenvalues and the center is within the unit square. The differential equation 4 has following solution for $\alpha(t)$ for the unconstrained dynamics[8]

$$\alpha(t) = B\sqrt{u}\cos(\sqrt{uu'}t + \phi) + \alpha^e$$

where B and ϕ are dependent on the intitial α, β. This also describes the constrained motion of the row agent when they have come down to an ellipse fully contained within the unit square. We note that

$$\frac{\partial^2 \alpha(t)}{\partial t^2} = -|uu'|(\alpha - \alpha^e)$$

From [2],

$$\begin{aligned} V_r(\alpha_t, \beta_t) - V_r(\alpha^e, \beta_t) &= (\alpha_t - \alpha^e) \frac{\partial V_r(\alpha_t, \beta_t)}{\partial \alpha} \\ &= -\frac{1}{|uu'|} \frac{\partial^2 \alpha(t)}{\partial t^2} \frac{\partial V_r(\alpha_t, \beta_t)}{\partial \alpha} \end{aligned}$$

Hence we have $V_r(\alpha_t, \beta_t) \geq V_r(\alpha^e, \beta^e)$ when $\frac{\partial^2 \alpha(t)}{\partial t^2} \frac{\partial V_r(\alpha_t, \beta_t)}{\partial \alpha} < 0$. Thus for the iterative updates, if $\Delta_k = \alpha_k - \alpha_{k-1}$ and $\Delta_k^2 = \Delta_k - \Delta_{k-1}$, then the row agent is *winning* if $\Delta_k \Delta_k^2 < 0$. This is not only easy to compute, but because this accurately estimates $V_r(\alpha_t, \beta_t) - V_r(\alpha^e, \beta_t)$, the new method is now guaranteed to converge in all general-sum games. There is a similar criterion of *winning* for the column agent. We note that the same criterion can be extended to the system in equation 7, since $\frac{\partial l^r(t)}{\partial t} = 0$ and $l^r(t)l^c(t) =$ constant within each quadrant, as is evident from figure 1. Figure 2 shows more clearly, how this criterion works in this case.

We demonstrate the working of our criterion on the bimatrix game $R = \begin{bmatrix} 0 & 3 \\ 1 & 2 \end{bmatrix}$, $C = \begin{bmatrix} 3 & 2 \\ 0 & 1 \end{bmatrix}$. This game is directly adopted from [2], where they used it as a counterexample for their approximate criterion. The only Nash Equilibrium is (0.5, 0.5), and the starting point is $(\alpha, \beta) = (0.1, 0.9)$. Figure 3 shows our algorithm converging to the equilibrium while their's failing. The experiment was run for $10,000$ iterations, and the accurate version converged in 1369 iterations. The choices for various parameters in both cases were thus: $l_{max} = 1.0$, $l_{min} = 0.08$, and precision[1] required was $\epsilon = 0.01$. We now turn to looking at learning with a variable rate from a control theory perspective.

6 Variable Learning Rate

Observe that when $uu' < 0$, the equation 4 specifies a conserved energy oscillation system. For the constrained system, the motion may reach an edge of the unit square but sooner or later it will come down to an ellipse contained completely in the unit square. Since energy is conserved, convergence is improbable. The only way to force it to converge is to apply a force of attenuation proportional to its velocity, and in a direction opposing its velocity. For real world free oscillations e.g. in case of a swinging pendulum, friction with air provides such

[1] i.e. if the stopping condition is $|\frac{\partial V_r(\alpha,\beta)}{\partial \alpha}| < \epsilon$ *and* $|\frac{\partial V_c(\alpha,\beta)}{\partial \beta}| < \epsilon$

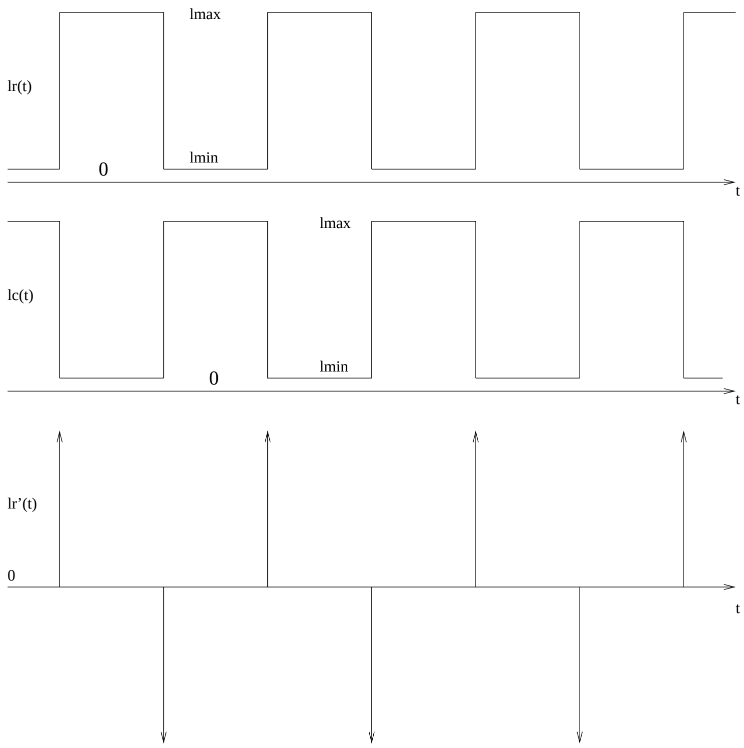

Fig. 1. The learning rate curves and the derivative for l^r

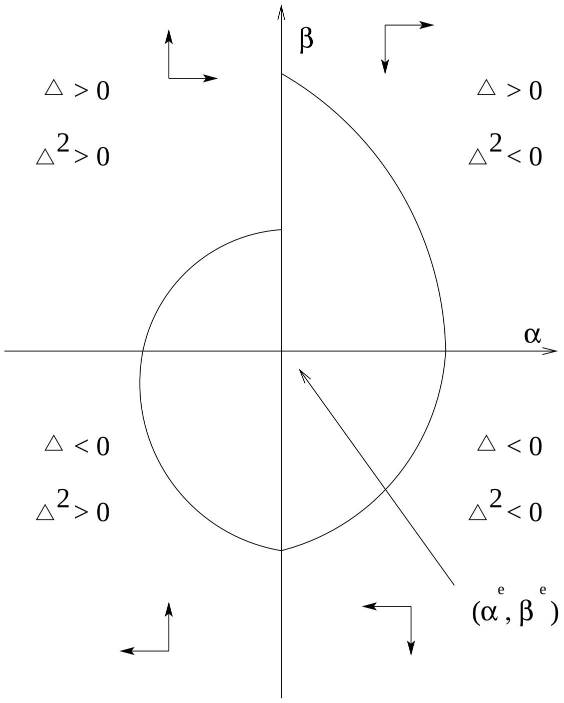

Fig. 2. The Δ and Δ^2 for row agent with WoLF-IGA. The gradient components are also shown.

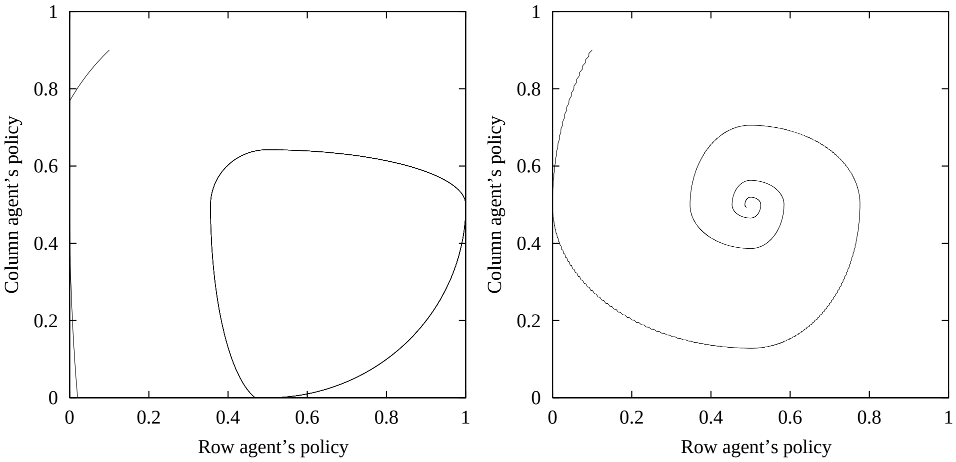

Fig. 3. Left: Failure at convergence with Bowling and Veloso's criterion. Right: Succesful convergence with the accurate criterion.

a force. We can create a second order differential equation from equation 7 for the row agent having the form

$$\frac{\partial^2 \alpha}{\partial t^2} - H(t)\frac{\partial \alpha}{\partial t} - F(t)\alpha = G(t)$$

where $H(t) = \frac{dl^r}{dt}/l^r$, $F(t) = l^r l^c uu'$ (which is incidentally a constant) and $G(t) = (r_{22} - r_{12})H(t) - (c_{22} - c_{21})l^r u$. Notice that in this case, $H(t)\frac{\partial \alpha}{\partial t}$ specifies the force[2] of attenuation, with magnitude proportional to $H(t)$. Hence if the learning rate is a constant at all times, the oscillation will never be damped, and consequently convergence will not be achieved. If $l_{min} > 0$ and l_{max} is finite, then $H(t)$ is also given by the bottom curve in figure 1. This involves a short force of infinite magnitude exactly at the points where (α, β) cross quadrants in the phase plane. At all other points the force is zero.

Given that we are constrained to have l^r and l^c constant within each quadrant (to maintain a piecewise elliptical trajectory for (α, β)), the attenuation within any quadrant is necessarily zero, since $\frac{dl^r}{dt}/l^r = 0$ inside any quadrant. This leaves only the quadrant-crossing points available for damping. An infinite force at such points is clearly the best attenuation achievable. This shows that with the given constraints, WoLF-IGA achieves fastest convergence to the center.

However, theoretically it is still possible to have overdamping (in case of any oscillation) which leads to fastest convergence possible, and which WoLF-IGA does not achieve. We would no longer be constrained to have elliptical orbits, and could design new $l^r(t)$ and $l^c(t)$ functions. This would also necessitate a

[2] This force was absent in the second order differential equation that could be created from equation 4

proof of convergence of all possible cases from scratch. The chief motivation for exploring this avenue is that WoLF is not sound when it comes to competitive games. Allowing the learner to slow down when "winning" assumes that the opponent will do the same when the learner is not "winning", at least in self-play. However, in order to develop a robust algorithm, we cannot discount the possibility that the opponent is an opportunist or that the domain is strictly competitive. We shall look into this prospect in future. We note in passing, that even if the opponent does not slow down when the learner is "loosing", WoLF-IGA is guaranteed to converge, albeit more slowly.

7 Optimal Learning Rate

Another issue that was not addressed in [2] is the choice of values for l_{min} and l_{max}. In order to leave the convergence of the system in other subcases unaffected, it is necessary to have $l_{min}, l_{max} > 0$. We also need $l_{min} < l_{max}$ to guarantee convergence for the remaining subcase. The choice of l_{max} is constrained by the fact that $\eta l_{max} \to 0$ for *infinitesimal* gradient ascent. Hence we cannot choose an l_{max} that is arbitrarily high. Now given l_{max}, we cannot make l_{min} arbitrarily close to zero, since that makes one of the agents stagnant (in that quadrant) and consequently slows down the other (notice from equations 1,2 that the gradient of α depends on β and vice versa). Actually as $l_{min} \to 0$, the convergence may slow down remarkably. We state and prove the theorem for the optimal range of l_{min} given l_{max}, below.

Theorem 1 *Given the choice of l_{max}, the fastest convergence of WoLF-IGA to precision ϵ can be achieved for a value of l_{min} in the range*

$$l_{max}\sqrt{2\epsilon} \geq l_{min} \geq l_{max}2\epsilon$$

Proof : (Outline) As noted in [2], the rate of convergence is proportional to $(\frac{l_{min}}{l_{max}})^2$ per "revolution"[3]. Since the maximum length of half-axis in the unit square is 0.5, in order to converge to a precision ϵ we require $(\frac{l_{min}}{l_{max}})^{2n} \leq 2\epsilon$, where n is the number of revolutions. This gives us the minimum number of revolutions required for convergence to precision ϵ. Since we cannot have convergence in less than a half "revolution" (this is deduced from the directions of the components of gradients in each quadrant, in figure 2), and we should not require more than one "revolution" (provided we choose l_{min} optimally), we require

$$1 \geq \frac{\log(2\epsilon)}{2\log(\frac{l_{min}}{l_{max}})} \geq \frac{1}{2}$$

which gives

$$l_{max}\sqrt{2\epsilon} \geq l_{min} \geq l_{max}2\epsilon$$

[3] By "revolution" we mean a complete rotation around the center, through 360^0. We assume that as $l^{r,c}\eta \to 0$ the number of revolutions is a monotonic function of time.

8 Conclusion

We have presented an accurate version of WoLF-IGA that is guaranteed to converge to Nash Equilibrium policies in all general-sum games. We have also argued from a control theoretic perpective why "Win or Learn Fast" makes sense in concurrent learning. This also revealed the possibility of having more complex learning rate functions with unexplored effects on convergence, that could alleviate the irrational "slow down" in learning in potentially competitive environments. Finally we have derived optimal learning rates for fast convergence of policy gradient learning in practical simulations. In future, we intend to explore the dynamics in situations with more than two actions available to the agents, and experiment on real-world applications.

References

1. Bikramjit Banerjee, Sandip Sen, and Jing Peng. Fast concurrent reinforcement learners. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence*, Seattle, WA, 2001.
2. M. Bowling and M. Veloso. Multiagent learning using a variable learning rate. *Artificial Intelligence*, 2002. In Press.
3. Caroline Claus and Craig Boutilier. The dynamics of reinforcement learning in cooperative multiagent systems. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence*, pages 746–752, Menlo Park, CA, 1998. AAAI Press/MIT Press.
4. J. Hu and M. P. Wellman. Multiagent reinforcement learning: Theoretical framework and an algorithm. In *Proc. of the 15th Int. Conf. on Machine Learning (ML'98)*, pages 242–250, San Francisco, CA, 1998. Morgan Kaufmann.
5. M. L. Littman. Markov games as a framework for multi-agent reinforcement learning. In *Proc. of the 11th Int. Conf. on Machine Learning*, pages 157–163, San Mateo, CA, 1994. Morgan Kaufmann.
6. John F. Nash. Non-cooperative games. *Annals of Mathematics*, 54:286 – 295, 1951.
7. G. Owen. *Game Theory*. Academic Press, UK, 1995.
8. S. Singh, M. Kearns, and Y. Mansour. Nash convergence of gradient dynamics in general-sum games. In *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence*, pages 541–548, 2000.