

A Policy-Improvement Type Algorithm for Solving Zero-Sum Two-Person Stochastic Games of Perfect Information

T. E. S. Raghavan^{1,2} and Zamir Syed³

Abstract

We give a policy-improvement type algorithm to locate an optimal pure stationary strategy for discounted stochastic games with perfect information. A geometric motivation for our algorithm is presented as well.

Keywords: Stochastic games, MDP, Perfect Information, and Policy Improvement

1. Introduction

Discounted stochastic games were first introduced by Shapley [1953]. In a stochastic game Γ , we have a finite set of states $S = \{1, 2, \dots, s\}$, and for each state $t \in S$ there are two finite sets $A(t) = \{1, 2, \dots, a_t\}$ and $B(t) = \{1, 2, \dots, b_t\}$ called the action sets for players I and II respectively. For each triple (t, a, b) with $a \in A(t)$ and $b \in B(t)$ there is an immediate reward $r(t, a, b)$ as well as a probability distribution $p(t, a, b)$ on the set S . Given an initial starting state $t_0 \in S$, the game is played as follows. The players simultaneously choose actions $a_0 \in A(t_0)$ and $b_0 \in B(t_0)$ resulting in the payment $r(t_0, a_0, b_0)$ to player I by player II. The system moves to a new state t_1 according to $p(t_0, a_0, b_0)$ and the players again choose actions $a_1 \in A(t_1)$ and $b_1 \in B(t_1)$. Accordingly the payment $r(t_1, a_1, b_1)$ is made to player I by player II and the game moves to a new state t_2 according to $p(t_1, a_1, b_1)$ and so on. The game continues infinitely and the rewards $r(t_n, a_n, b_n)$ are recorded. A general strategy for a player would be a function from the

¹Department of Mathematics, Statistics and Computer Science, University of Illinois at Chicago; e-mail: ter@uic.edu

²Partially Funded by NSF Grant DMS 930-1052 and DMS 970-4951

³Department of Mathematics, Statistics and Computer Science, University of Illinois at Chicago; e-mail: ztatti@uic.edu

set of all possible histories into the set of probability distributions over the player's action space. A general strategy can therefore be very complicated but nevertheless, given a pair of strategies (π, ρ) for both players, we can evaluate the expected β -discounted value:

$$\phi_\beta(\pi, \rho)(t_0) = \sum_{n=0}^{\infty} \beta^n r_n(t_0, \pi, \rho)$$

where t_0 is the starting state and $r_n(t_0, \pi, \rho)$ is the expected reward (to player I) at the n th stage when the players are using π and ρ . For any two vectors u and v we write $u \leq v$ to mean $u_i \leq v_i$ for all i . Under this payoff we can define an *equilibrium* pair of strategies to be a pair (π^*, ρ^*) such that:

$$\phi_\beta(\pi, \rho^*) \leq \phi_\beta(\pi^*, \rho^*) \leq \phi_\beta(\pi^*, \rho)$$

for all π and ρ (sometimes we refer to such pairs as *saddlepoints*). A strategy is said to be *stationary* if it only depends on the current state. Shapley[1953] showed that under the discounted payoff criterion, there always exists an equilibrium pair in stationary strategies, and further that the equilibrium payoff vector is unique. This allows us to write $\phi_\beta(\Gamma) = \phi_\beta(\pi^*, \rho^*)$.

Successive approximation methods for finding optimal stationary strategies of discounted stochastic games are well known (Van der Waal[1976]). The question of whether or not for a specific class of stochastic games a finite algorithm exists is generally open. In this paper we consider a special class of stochastic games, those with perfect information, which can be solved via a finite algorithm. The class is so similar to the traditional MDP that our task became to try to apply policy improvement to these games. The existence theorem for perfect information stochastic games imposes a strong combinatorial structure on them. This then serves as a motivation for our algorithm which can be thought of as an extension of Blackwell's methods (Blackwell [1963]).

The algorithm will be presented followed by the geometrical motivation. We exhibit a run of our algorithm on a specific example and close with a conjecture concerning the aforementioned geometric structure of this class of games.

2. Perfect Information Games

Under the same set-up if in each state at least one of the two action sets is a singleton, we say that the game is of perfect information. By deleting any proper subsets of actions from the action spaces the resulting subgame is still one of perfect information. These games are similar to the standard finite Markov decision processes (MDP) except that the goal here is to maximize in some states and minimize in others (rather than maximizing in all states). In fact, if a fixed player has exactly one action in every state then the game reduces to an MDP. For this reason stochastic games with perfect information can be thought of as an immediate generalization of the Markov decision process.

A non-randomized stationary strategy is called a *pure stationary strategy*. For player I a pure stationary strategy is simply a function $f : S \rightarrow \cup_{i=1}^s A_i$ with $f(t) \in A_t$ for all t , that is, in state t player I always chooses the action $f(t)$. Similarly a function $g : S \rightarrow \cup_{i=1}^s B_i$ with $g(t) \in B_t$ for all t is a pure stationary strategy for player II. Shapley[1953] showed that under the discounted payoff criterion, perfect information stochastic games admit equilibria in pure stationary strategies, for both players. For a pair of pure stationary strategies (f, g) we write $\phi_\beta(f, g)$ to be the vector of expected discounted payoffs, resulting from f and g , indexed by the starting state. We also write $Q(f, g)$ for the transition matrix of the system, indexed by the state space, resulting from f and g . The i th row of $Q(f, g)$ will be written as $Q_i(f, g)$, and by the definition of perfect information we can always write either $Q_t(f, g) = Q_t(f)$ or $Q_t(f, g) = Q_t(g)$ for any state $t \in S$. Likewise we write $r(f, g)$ to be the vector indexed by the state space whose i th component is $r(i, f(i), g(i))$. Just like $Q(f, g)$, $r(f, g)$ can be broken into parts which depend on only one of f and g .

For the MDP there is the so-called *policy-improvement* algorithm (Blackwell[1962]) which can be used to determine optimal policies. This algorithm starts at an arbitrary policy f_0 and produces a sequence of improvements f_1, f_2, \dots, f_k until an optimal policy is reached. In the sequence of policies the corresponding values ϕ_β are strictly monotonic and therefore the algorithm must terminate (only a finite number of pure stationary policies.) In this paper we generalize this algorithm to the case of the stochastic game with perfect information. Extending the policy-improvement algorithm of MDP's to stochastic games was initially attempted by Pollatschek and Avi-Itzhak [1969], however, they were only able to prove that their algorithm terminates for games with a stringent condition on the transitions and the discount fac-

tor (see O.J. Vrieze [1983], Van der Waal [1976]). For a general survey on algorithms for stochastic games see Raghavan and Filar [1991]. We have also found Kallenberg[1983] to be an excellent source on policy-improvement in MDPs.

3. Algorithm

Our problem now is as follows: Given a stochastic game Γ with perfect information, we would like to find a pair of pure stationary strategies (f^*, g^*) which are an equilibrium pair for Γ . Due to the existence theorem (Shapley[1953]) we only need

$$\phi_\beta(f, g^*) \leq \phi_\beta(f^*, g^*) \leq \phi_\beta(f^*, g)$$

for all pure stationary strategies f and g . For a pair of pure stationary strategies (f, g) we write $(f^1, g^1), \dots, (f^s, g^s)$ where (f^k, g^k) is the pair of actions chosen in state k under (f, g) . By the definition of perfect information games, we have that for any i at least one of f^i and g^i is 1. An *improvement of type I* is a new pair of pure stationary strategies (h, g) where:

1. $\exists k, 1 \leq k \leq s$, with $h^k \neq f^k$ and $h^i = f^i$ for $i \neq k$
2. $\phi_\beta(h, g) \geq \phi_\beta(f, g)$ and $\phi_\beta(h, g)_i > \phi_\beta(f, g)_i$ for some $1 \leq i \leq s$

The purpose of the second condition is clearly that player I is better off playing h than f against player II's g . The first condition is an adjacency condition required in our algorithm. It states that h differs from f in exactly one state. Of course we have the corresponding definition for *improvement of type II*, namely it is a pair (f, h) where:

1. $\exists k, 1 \leq k \leq s$, with $g^k \neq h^k$ and $g^i = h^i$ for $i \neq k$
2. $\phi_\beta(f, h) < \phi_\beta(f, g)$ and $\phi_\beta(f, h)_i < \phi_\beta(f, g)_i$ for some $1 \leq i \leq s$

Notice that in both improvements we require a *strict* improvement in ϕ_β value. A pair of pure stationary strategies (f', g') will be called an *improvement* of (f, g) if it is an improvement of either type I or type II. Note that in such a case we would have either $f' = f$ or $g' = g$ depending on the type of improvement. In our algorithm we start with a pair of pure stationary strategies

and generate a sequence of improvements via lexicographic search. That is, when looking for an improvement of the current pair, we start in state 1 and proceed as follows: Within state 1 itself we first look for an improvement of type I. If such an improvement doesn't exist then we search for an improvement of type II. Now if neither exist then the search moves to state 2 and we repeat the procedure. After an improvement of either type is found, we move to the new pair and begin searching for improvements back in state 1 again. We will prove that such a procedure must terminate in a saddlepoint pair (f^*, g^*) , i.e. a pair with no improvements..

Algorithm:

1. Choose initial pair of pure stationary strategies (f_0, g_0) arbitrarily (e.g. $f_0^k = g_0^k = 1$ for $k = 1, \dots, s$) and set $\tau = 0$.
2. Search lexicographically for an improvement $(f_{\tau+1}, g_{\tau+1})$ of (f_τ, g_τ) . There are two cases:
 - Case 1: An improvement is found. In this case let $\tau = \tau + 1$ and repeat step 2.
 - Case 2: There are no improvements. Go to step 3.
3. The pair $(f^*, g^*) = (f_\tau, g_\tau)$ is a saddlepoint.

To show that the algorithm terminates we need a few results.

In all that follows, we assume that the underlying stochastic game Γ is of perfect information.

Lemma 1: In a zero-sum, perfect information, stochastic game Γ , a pair of pure stationary strategies (f, g) is an equilibrium pair if and only if $\phi_\beta(f, g) = \phi(\Gamma)$.

Proof: That an equilibrium pair (f^*, g^*) satisfies $\phi_\beta(f^*, g^*) = \phi(\Gamma)$ follows from the uniqueness of equilibrium payoff. Now suppose $\phi_\beta(f, g) = \phi(\Gamma)$ for some pair of strategies (f, g) . Let (f^*, g^*) be an equilibrium pair. Then for any strategy h for player I we have (this is a vector inequality)

$$r(h, g^*) + \beta Q(h, g^*) \phi_\beta(f^*, g^*) \leq \phi_\beta(f, g).$$

On coordinates controlled by player II, we must have equality since h and f^* must choose the same (the only) action there. Therefore, viewing this inequality on those coordinates $k \in S$ which are controlled by player I, and

using the facts that here $r(h, g^*)_k = r(h)_k = r(h, g)_k$, $Q(h, g^*)_k = Q(h)_k = Q(h, g)_k$, and $\phi_\beta(f, g) = \phi(\Gamma) = \phi_\beta(f^*, g^*)$, yields

$$r(h, g) + \beta Q(h, g) \phi_\beta(f, g) \leq \phi_\beta(f, g).$$

Similar is the other side of the equilibrium condition. ◇

For what follows we require some notation. Let $t \in S$ be a fixed state. For any $X \subset A_t$ we write Γ_X^t to the subgame in which only the actions X are allowed in state t . The corresponding pure stationary strategy sets will be denoted by F_X^t and G_X^t . For the original game Γ we write F and G for the pure stationary strategy sets of players I and II respectively.

Lemma 2: For $t \in S$, $X \subset A_t$, $Y \subset A_t$, $X \cap Y = \emptyset$ we have $\phi(\Gamma_{X \cup Y}^t) = \max\{\phi_\beta(\Gamma_X^t), \phi_\beta(\Gamma_Y^t)\}$.

Proof: Clearly $G_X^t = G_Y^t = G_{X \cup Y}^t = G$. From the definition of a saddlepoint and the existence theorem of pure stationary equilibria (Shapley[1953]) we have:

$$\begin{aligned} \phi_\beta(\Gamma_{X \cup Y}^t) &= \max_{f \in F_{X \cup Y}^t} \min_{g \in G_{X \cup Y}^t} \phi_\beta(f, g) \\ &= \max\{\max_{f \in F_X^t} \min_{g \in G_{X \cup Y}^t} \phi_\beta(f, g), \max_{f \in F_Y^t} \min_{g \in G_{X \cup Y}^t} \phi_\beta(f, g)\} \\ &= \max\{\max_{f \in F_X^t} \min_{g \in G_X^t} \phi_\beta(f, g), \max_{f \in F_Y^t} \min_{g \in G_Y^t} \phi_\beta(f, g)\} \\ &= \max\{\phi_\beta(\Gamma_X^t), \phi_\beta(\Gamma_Y^t)\} \end{aligned}$$

This completes the proof. ◇

Remark: From the proof of Lemma 2 we can also conclude that the vectors $\phi_\beta(\Gamma_X^t)$ and $\phi_\beta(\Gamma_Y^t)$ are comparable, that is, either $\phi_\beta(\Gamma_X^t) \leq \phi_\beta(\Gamma_Y^t)$ or $\phi_\beta(\Gamma_X^t) \geq \phi_\beta(\Gamma_Y^t)$ holds.

An obvious player II analog of Lemma 2 exists using B_t instead of A_t . The proof of such is identical except that the min-max form of the saddlepoint is used.

Theorem 1: The algorithm terminates in a finite number of steps and furthermore $(f_i, g_i) = (f_j, g_j)$ if and only if $i = j$, that is, the sequence never passes through the same pair twice.

Proof: The proof is by induction on $n = \sum_{i=1}^s (a_i + b_i)$. The smallest this value can be is 2 in which case the algorithm trivially terminates at (f_0, g_0) . Therefore, suppose the algorithm terminates at a saddlepoint policy whenever $n = 2, \dots, k$ and assume $n = k + 1$. If $a_i = b_i = 1$ for all i then again there is nothing to prove. So without loss of generality suppose there is a

state where one of the players has more than one action. Let j be the largest value of k for which in state k a player has more than one action. Now in state j one of the players has more than one action, and the other has exactly one action. We will prove the theorem for the first case as the proof for the second case is almost identical. So assume that player I has more than one action in state j .

We now split the game at state j . The algorithm will pass through a sequence c_1, c_2, \dots, c_m of actions in state j . The first such action is $c_1 = f_0^j$ whereas c_2, \dots, c_m will be written later on. Let $X_i \subset A_j$ be defined by $X_i = \{c_1, c_2, \dots, c_i\}$. Now suppose the algorithm is initiated at (f_0, g_0) . By the induction hypotheses, the algorithm will reach a policy (f_{n_1}, g_{n_1}) which has no improvements in $\Gamma_{X_1}^j$. That is, by construction, (f_{n_1}, g_{n_1}) is a saddlepoint of $\Gamma_{X_1}^j$. If there are no improvements of (f_{n_1}, g_{n_1}) in Γ as well, then the algorithm terminates and $(f^*, g^*) = (f_{n_1}, g_{n_1})$ is the saddlepoint of Γ with $\phi_\beta(\Gamma) = \phi_\beta(f_{n_1}, g_{n_1}) = \phi_\beta(\Gamma_{X_1}^j)$. Otherwise, the only improvement can be to change f_{n_1} in state j .

Let $c_2 = f_{n_1+1}^j$, the new action chosen in state j . After this improvement, the algorithm continues and again by the induction hypotheses we have that the algorithm will reach a saddlepoint (f_{n_2}, g_{n_2}) of the subgame $\Gamma_{\{c_2\}}^j$. Applying the previous lemma with $X = \{c_1\}$ and $Y = \{c_2\}$ we get $\phi_\beta(f_{n_2}, g_{n_2}) = \phi_\beta(\Gamma_{X_2}^j)$. By Lemma 1 we can conclude that there are no improvements of (f_{n_2}, g_{n_2}) in $\Gamma_{X_2}^j$, and further (by the remark following lemma 2) that $\phi_\beta(f_{n_1}, g_{n_1}) \leq \phi_\beta(f_{n_2}, g_{n_2})$ with strict inequality in some coordinate. If there are no improvements in Γ as well then $(f^*, g^*) = (f_{n_2}, g_{n_2})$ is a saddlepoint of Γ and the algorithm terminates. Otherwise an improvement in state j to an action outside X_2 is available. Just as before, let c_3 be this action. Because there is only a finite number of actions in state j (as in all states), by repeating the same arguments as before we get that the algorithm will pass through the saddlepoints $(f_{n_1}, g_{n_1}), (f_{n_2}, g_{n_2}), \dots, (f_{n_m}, g_{n_m})$ of the respective games $\Gamma_{X_1}^j, \Gamma_{X_2}^j, \dots, \Gamma_{X_m}^j$, and (f_{n_m}, g_{n_m}) a pair with no improvements. That is, $(f^*, g^*) = (f_{n_m}, g_{n_m})$ is a saddlepoint of Γ .

Since improvements in state j travel strictly through the sequence c_1, c_2, c_3, \dots , by the same induction we get non-repetition in the sequence. To prove the theorem in the case of player II having more than one action in state j is no more than using the player II-analog of lemma 2. \diamond

Remark: From the remark following lemma 2 we actually have a partial monotonicity property of the sequence, namely $\phi_\beta(f_{n_1}, g_{n_1}) \leq \phi_\beta(f_{n_2}, g_{n_2}) \leq$

$\dots \leq \phi_\beta(f_{n_m}, g_{n_m})$, of which no two vectors are the same.

4. Hypercubes

In this section we define a class of hypercubes which will eventually model (some combinatorial properties of) perfect information stochastic games. Since there are two players in the games we considered in the previous sections, we spoke of a *pair* of strategies (f, g) . But this pair is essentially a function $\theta : S \rightarrow \cup_{t=1}^s (A_t \cup B_t)$ with $\theta(i) \in A_i$ for those states i belonging to player I, and $\theta(j) \in B_j$ for those states j belonging to player II. In other words, θ just chooses an action in each state. In MDP terminology, θ is just a policy (for the single player). In defining our hypercubes, the idea is to associate each policy (or pair (f, g) of strategies) of Γ with a vertex of a graph. In the next section we connect the two.

Let s be a fixed positive integer. We define a class of digraphs Ω_s as follows. Let a_1, a_2, \dots, a_s be integers with $a_i \geq 2$ for all i and set $A_i = \{1, 2, \dots, a_i\}$. Consider a digraph C with the properties (P1), (P2), and (P3) that follow.

(P1): The vertex set of C is the set $\prod_{i=1}^s A_i$.

Two vertices $\alpha, \theta \in \prod_{i=1}^s A_i$ are called *neighbors* if and only if they differ in exactly one coordinate.

(P2): For any pair of neighboring vertices α and θ , at least one of (α, θ) and (θ, α) is an edge.

(P3): If (α, θ) is an edge of C then α and θ are neighbors.

The number s will be called the dimension of C . The set of all such digraphs C (which result from some s, a_1, a_2, \dots, a_s) will be denoted by Ω_s , and we define $\Omega = \cup_{s=1}^{\infty} \Omega_s$. From here on we write $\alpha \in C$ to mean α is a vertex of C . If (α, θ) and (θ, α) are both edges of C then we write $\alpha \leftrightarrow \theta$. Temporarily we will assume that $a_i = 2$ for all i and refer to this restricted class of digraphs by Ω' ($= \cup_{s=0}^{\infty} \Omega'_s$). In this case, C can be identified with the well-known s -dimensional hypercube. Given any fixed vertex $\alpha \in C$, there are exactly s vertices adjacent to α , call them $\theta_1, \theta_2, \dots, \theta_s$. Without loss of generality, assume that α and θ_i differ in the i th coordinate. If (α, θ_i) is an edge then we say that α is a *max in dimension i* . If (θ_i, α) is an edge then we say that

α is a *min in dimension i* . Note that $\alpha \leftrightarrow \theta_i$ implies that α is both a max and a min in dimension i .

Any vertex α is a max in some dimensions and a min in others. All together there are 2^s such possible max/min configurations for α . If $\alpha \leftrightarrow \theta$ for some vertex θ then α will have multiple configurations. Each such θ will double the number of configurations α has.

Let $S = \{1, 2, \dots, s\}$ and let T be an arbitrary subset of S . For any $b \in \prod_{i \in T} A_i$, we can define the subgraph $C_{b,T}$ of C by restricting all coordinates in T to b . If $|T| = k$ then $C_{b,T}$ can be identified with an $(s - k)$ -dimensional hypercube in its own right, and thus we have $C_{b,T} \in \Omega'_{s-k}$. $C_{b,T}$ will be called a *subcube* of C . Furthermore, given any two vertices $\alpha, \theta \in C$, there is a unique subcube of smallest dimension which contains them. We call this subcube $C(\alpha, \theta)$. A single max/min configuration for a vertex α in a t -dimensional subcube D can be thought of as a binary t -tuple $([\alpha]_{i_1}, \dots, [\alpha]_{i_t})$ where $[\alpha]_{i_q}$ equaling zero(one) means that in this configuration, α is a min(max) in dimension q of D (here we need to write i_1, \dots, i_t to denote the t non-constant coordinates on which D is defined.) We say α and θ *share a max/min configuration* in a subcube D to mean that $([\alpha]_{i_1}, \dots, [\alpha]_{i_t})$ and $([\theta]_{i_1}, \dots, [\theta]_{i_t})$ are max/min configurations of α and θ respectively in the subcube D , and they are equal as binary vectors.

A digraph $C \in \Omega'_s$ will be called *complete* if all 2^s max/min configurations are present in C . C will be called *balanced* if the following properties hold:

- (B1):** If two vertices $\alpha, \theta \in C$ have a max/min configuration in common, then for every pair of vertices $h, k \in C(\alpha, \theta)$ we have $h \leftrightarrow k$.
- (B2):** If $\alpha \leftrightarrow \theta$ for some pair of neighboring vertices $\alpha, \theta \in C$ then α and θ have the same max-min configurations, i.e. for all i , α is a max(min) in dimension i if and only if θ is a max(min) in dimension i .

A digraph C will be said to be *full* if the following two conditions hold:

(F1): The digraph C and all its subcubes are complete.

(F2): The digraph C and all its subcubes are balanced.

The set of full s -dimensional elements of Ω'_s will be denoted by Λ'_s and we write $\Lambda' = \cup_{s=1}^{\infty} \Lambda'_s$. Next we define the maps $W_i : \Omega'_s \rightarrow \Omega'_s$ for $i = 1, 2, \dots, s$. For $C \in \Omega'_s$ we define $W_i(C) \in \Omega'_s$ to have the same vertex set as C . The edges of $W_i(C)$ are the same as those in C except that those edges whose vertices differ in the i th coordinate are reversed, i.e. for all neighboring pairs $\alpha, \theta \in C$ with α and θ differing in the i th coordinate, (α, θ) is an edge of $W_i(C)$ if and only if (θ, α) is an edge of C (See **Figure 2**). It follows directly from the definition that $W_i(W_i(C)) = C$ and that $W_i(W_j(C)) = W_j(W_i(C))$ for all i, j .

The vertex α in **Figure 1** is a min in all three dimensions, whereas θ is a max in dimension 1 and a min in dimensions 2 and 3. The other six vertices possess the rest of the max/min configuration.

Lemma 3: $W_i(\Lambda'_s) = (\Lambda'_s)$.

Proof: Let $C \in \Lambda'_s$. If $s = 1$ then C has exactly two vertices, call them α and θ . If (α, θ) is an edge then (θ, α) is an edge in $W_1(C)$ so that θ is a max and α is a min in dimension 1. Thus all of the $2^s = 2^1$ configurations are present, and therefore $W_1(C)$ is complete. If α and θ share some configuration then they must both be max's and min's in dimension 1. This just means that (α, θ) as well as (θ, α) are edges in C and by definition are also edges of $W_1(C)$. Thus $\alpha \leftrightarrow \theta$ in $W_1(C)$ as well so that F2) is also satisfied.

Suppose the lemma holds for $C \in \Lambda'_s$, $s = 1, 2, \dots, n - 1$. If $s = n$ then we proceed as follows. Fix $T = \{i\}$ and write $D = W_i(C)$. Then $D_{0,T}$ and $D_{1,T}$ are the subcubes of D which correspond to fixing the i th coordinate of all vertices to 0 and 1 respectively. By the induction hypothesis we have that W_i leaves all proper subcubes complete, in particular, $D_{0,T}$ and $D_{1,T}$. If α and θ , viewed as vertices in C , have the same max-min configurations then viewed as vertices in D they will as well. Also, W_i leaves double edges unaltered so that D satisfies F2). Thus, we only need check that D itself is complete.

Let α be any vertex in C . We will show that the max/min configuration of α is present in D . Suppose $\alpha \in C$ is a max in dimension i . As C is complete, there must be a vertex $\theta \in C$ which has the same max/min configuration

as α except that it is a min in dimension i . Then θ viewed as a vertex in D will have the same max/min configuration in all dimensions as in C except that in dimension i it will be a max. Thus, $\theta \in D$ will have the same max/min configuration as $\alpha \in C$. Similar is the case for $\alpha \in C$ being a min in dimension i . As α was arbitrary, all max/min configurations in C appear in D . Therefore the completeness of C implies that of D . This proves that $D \in \Lambda'_s$ whence $W_i(\Lambda') \subset \Lambda'$. Applying W_i to this last containment yields $\Lambda'_s = W_i(W_i(\Lambda'_s)) \subset W_i(\Lambda'_s) \subset \Lambda'_s$ so that $W_i(\Lambda'_s) = \Lambda'_s$. This completes the proof. \diamond

Lemma 4: Let $C \in \Lambda'_s$ for some s and let $\alpha \in C$ be any vertex. Then there exists a sequence of vertices $\alpha = \alpha_0, \alpha_1, \dots, \alpha_k = \alpha^*$ where $(\alpha_{i+1}, \alpha_i), i = 0, 1, \dots, k-1$, are edges in C and α^* is a max indimension i for all i .

Proof: If $s = 1$ then C has exactly two distinct vertices, and it is trivial to check that the lemma holds. Suppose the lemma holds for $s = 1, 2, \dots, n-1$. If $s = n$ then we proceed as follows. Let $T = \{n\}$ and let α be any vertex in C . Without loss of generality, assume that $\alpha \in C_{0,T}$. By the induction hypothesis there exists a sequence $\alpha = \alpha_0, \alpha_1, \dots, \alpha_j = \theta$ of vertices in $C_{0,T}$ such that $(\alpha_{i+1}, \alpha_i), i = 0, 1, \dots, j-1$ are edges in $C_{0,T}$ and θ is a max in dimensions i for $i = 1, 2, \dots, s-1$. If θ is also a max in dimension s then setting $k = j$ and $\alpha^* = \theta$ yields the desired sequence. So assume that θ is not a max in dimension s whence there must be a vertex h in $C_{1,T}$ such that (h, θ) is an edge in C . Set $\alpha_{j+1} = h$. Again by the induction hypothesis there exists a sequence of vertices $\alpha_{j+1}, \alpha_{j+2}, \dots, \alpha_k$ in $C_{1,T}$, $(\alpha_{j+l+1}, \alpha_{j+l})$ and edge of $C_{1,T}$, where α_k is a max in dimension i for $i = 1, 2, \dots, s-1$. We claim that α_k must be a max in dimension s as well. If α_k were a min in dimension s then it would have the same max/min configuration as θ . As C is full, this would imply that for all $u, v \in C(\theta, \alpha_k)$ we have $u \leftrightarrow v$. As θ differs from both h and α_k in the s th coordinate and as $\alpha_k \in C(\theta, \alpha_k)$, we must have that $\theta \leftrightarrow h$. But this contradicts our assumption that θ is not a max in dimension s . Setting $\alpha^* = \alpha_k$ yields the sequence required in the lemma. This completes the proof. \diamond

Any sequence $\alpha_0, \alpha_1, \dots, \alpha_k$ where $(\alpha_{i+1}, \alpha_i), i = 0, 1, \dots, k-1$ are edges in C will be called an *increasing sequence*. An increasing sequence $\alpha_0, \alpha_1, \dots, \alpha_k$ in C is called *strictly increasing* if (α_i, α_{i+1}) is not an edge of C for $i = 1, \dots, k-1$. A vertex which is max in dimension i for all i will be called a *max-vertex*. These definitions will also be used in the general class Ω .

Corollary 1: Let $C \in \Lambda'_s$ for some s . Let $\alpha \in C$ be any vertex and let

$\alpha^* \in C$ be any max-vertex. Then there exists an increasing sequence in C starting at α and terminating at α^* .

Proof: By Lemma 4 there exists an increasing sequence starting at α and terminating at some max-vertex θ . As α^* is also a max-vertex, it has the same max/min configuration as θ . Therefore, F2) guarantees an increasing sequence from θ to α^* . Concatenating the latter sequence with the former gives an increasing sequence from α to α^* . This completes the proof. \diamond

By a closer examination of the proof of Lemma 4 we get the following:
Lemma 5: Let $C \in \Lambda'_s$ for some s . Let $\alpha \in C$ be any vertex. Then there exists a strictly increasing sequence in C starting at α and terminating at some max-vertex.

We next return to the original class of digraphs Ω and refer to the elements of Ω' as *2-cubes*. Let $C \in \Omega_s$ for some $s \geq 1$. If we restrict the sets $A_i, i = 1, \dots, s$, corresponding to the digraph C , to some (non-empty) subsets $A'_i \subset A_i$, we get a subgraph C' with vertex set $\prod_{i=1}^s A'_i$. Dropping those A'_i which contain only one element yields a subgraph which can be considered an element of Ω in its own right. Just as before, we will call C' a *subcube* of C . If we have $|A'_i| \leq 2$ for all i then C' will be a 2-cube. If exactly k of the sets A'_i have only one element, then $C' \in \Omega_{s-k}$. Furthermore, given any two vertices $\alpha, \theta \in C$, there is a unique smallest subcube, again denoted by $C(\alpha, \theta)$, containing both α and θ (i.e. no proper subcube of $C(\alpha, \theta)$ contains both α and θ). It is easy to see that $C(\alpha, \theta)$ must be a 2-cube.

Let $C \in \Omega_s$ and let $\alpha \in C$ be an arbitrary vertex. Consider all s -dimensional 2-cubes which are subcubes of C and which contain the vertex α . If α is a max in dimension i in all these 2-cubes then we say that $\alpha \in C$ is a *max in dimension i* . Identical is the definition of $\alpha \in C$ being a *min in dimension i* . Unlike the case of the 2-cubes, the vertex α can be a max, min, both or neither in dimension i . If the vertex α is either a min or max in every dimension then we can speak of α having a max/min configuration. If all 2^s possible max/min configurations are present in C then C is said to be *complete*. The definitions of *balanced* and *full* for 2-cubes can be taken verbatim for $C \in \Omega$. The set of full s -dimensional elements of Ω will be denoted by Λ_s and we write $\Lambda = \cup_{s=1}^{\infty} \Lambda_s$. We can apply the definition of the maps $W_i, i = 1, \dots, s$ to the general set Ω_s , and again we get:

Lemma 3': $W_i(\Lambda_s) = \Lambda_s$

Proof: The proof is similar to that of Lemma 3 except that the induction is on $n = \sum_{i=1}^s a_i$. It mainly requires noticing the fact that if a vertex α is a

min(max) in dimension i , then W_i changes it to a max(min) in dimension i .
 \diamond

Lemma 4': Let $C \in \Lambda_s$ for some s . Let $\alpha \in C$ be any vertex and let $\alpha^* \in C$ be any max-vertex. Then there exists an increasing sequence in C starting at α and terminating at α^* .

Proof: Given α and α^* consider the subcube $D = C(\alpha, \alpha^*)$. Since D is a 2-cube we can apply Corollary 1 to prove the existence of the desired sequence. This completes the proof. \diamond

Lemma 5': For any $C \in \Lambda$ and any vertex $\alpha \in C$, there exists a strictly increasing sequence starting at α and terminating at some max-vertex $\alpha^* \in C$.

Proof: Let $C \in \Lambda$ and $\theta \in C$ be any max-vertex in C . Given any vertex $\alpha \in C$, consider the 2-cube $C(\alpha, \theta)$. We know from Lemma 5 that in this subcube there is a strictly increasing sequence from α to some max vertex $\theta' \in C(\alpha, \theta)$. As θ is also a max-vertex of $C(\alpha, \theta)$, by property (B1) of $C(\alpha, \theta)$ we must have a sequence $\theta' = \theta_0, \theta_1, \dots, \theta_k = \theta$ in $C(\alpha, \theta)$ with $\theta_i \leftrightarrow \theta_{i+1}$ for $i = 0, \dots, k - 1$. By property (B2) of C , θ' and θ have the same max-min configurations (as well as the intermediate θ_i). As θ is a max-vertex of C , so is θ' . This completes the proof. \diamond

5. The Clear Connection

Given a stochastic game of perfect information Γ , we define a digraph $C_\Gamma \in \Omega$ as follows. Let $S = \{1, 2, \dots, s\}$ be the state space of Γ where we leave out those states which have only one action for the player of that state. As before let a_i be the number of actions in state i . We define a digraph $C_\Gamma^0 \in \Omega_s$ just as before using the data s, a_1, \dots, a_s . Each vertex of C_Γ^0 corresponds to a pair of pure stationary strategies for both players. If α is a vertex, we write (f_α, g_α) for the corresponding pair of pure stationary strategies. There may be a little confusion here concerning ordered pairs. The pair $(\delta, *)$ is an edge of a digraph if δ and $*$ are vertices (as in (α, θ) for example), but refers to a pair of pure stationary strategies in a game if δ and $*$ are strategies for players I and II respectively (for example (f_α, g_α)). A pair of vertices α and θ will comprise an edge (α, θ) of C_Γ^0 if and only if (f_α, g_α) and (f_θ, g_θ) differ in exactly one state and $\phi_\beta(f_\alpha, g_\alpha) \geq \phi_\beta(f_\theta, g_\theta)$. The following observations are direct consequences of Blackwell [1962].

For any two adjacent vertices α and β , the respective payoff vectors $\phi_\beta(f_\alpha, g_\alpha)$ and $\phi_\beta(f_\theta, g_\theta)$ are comparable so that C_Γ^0 satisfies properties P1, P2, and P3. The subgames of Γ correspond to subcubes of C_Γ^0 . Furthermore, from arguments similar to those used in proving Lemma 1 and Lemma 2, it is easy to show that $C_\Gamma^0 \in \Lambda_s$.

We next define a sequence of digraphs $C_\Gamma^i, i = 1, 2, \dots, s$ recursively. Set

$$C_\Gamma^i = \begin{cases} W_i(C_\Gamma^{i-1}) & \text{if } a_i = 1 \\ C_\Gamma^{i-1} & \text{if } b_i = 1 \end{cases}$$

and define $C_\Gamma = C_\Gamma^s$. As $C_\Gamma^0 \in \Lambda_s$ we can conclude, by Lemma 3, that $C_\Gamma \in \Lambda_s$ as well. The application of the functions W_i on C_Γ^0 result in the max-vertex of C_Γ corresponding to the saddlepoint policy of Γ . Let α_0 be an arbitrary vertex corresponding to a fixed pair of pure-stationary policies for both players. By Lemma 4' we know that in C_Γ there exists an increasing sequence of vertices $\alpha_0, \alpha_1, \dots, \alpha_k$ where α_k is a max-vertex whence $(f_{\alpha_k}, g_{\alpha_k})$ is a saddlepoint of Γ .

In the representation of the game Γ as a hypercube C_Γ , we ignored those states for which only one action was present. Including those states would have increased the dimension of C_Γ unnecessarily. After all, consider a game Γ with ten states which has two actions in state 1 and one action in the other nine states. This game, viewed as an MDP, has only two distinct policies so that its corresponding hypercube C_Γ would be two vertices connected by one or two edges. That is, C_Γ is a one-dimensional hypercube (and not a ten dimensional one). However, in computing the required sequence of policies, we cannot ignore these states.

To proceed directly from these results we would start with a policy and generate a sequence of improvements. The natural way to search for improvements is to do so lexicographically. This amounts to restricting the search to the smaller subcubes of the graph C_Γ first and as the saddlepoints of the subgames are determined, the larger subcubes would be entered. This is exactly what our algorithm does. From a computational aspect, computing $\phi_\beta(f, g)$ for each new improvement involves a single column pivot and a matrix multiplication.

6. An Example

Next we present a run of the algorithm on a randomly generated stochastic game¹. The game will have 6 states. Player I controls states 1,2,5 and 6 while Player II controls states 3 and 4. Each box represents a state. The first column of the box gives the immediate reward for each action while the second column gives the respective transition to the next state. The discount factor for the game is set at $\beta = 0.999$.

State 1: Player I

| | | | | | | | | |
|---|-------|-------|-------|-------|-------|-------|-------|---|
| 1 | 5.069 | 0.309 | 0.018 | 0.117 | 0.137 | 0.143 | 0.277 | |
| 2 | 5.093 | 0.143 | 0.204 | 0.159 | 0.050 | 0.271 | 0.173 | ← |

State 2: Player I

| | | | | | | | | |
|---|-------|-------|-------|-------|-------|-------|-------|---|
| 1 | 5.088 | 0.178 | 0.238 | 0.171 | 0.108 | 0.173 | 0.132 | ← |
| 2 | 5.086 | 0.172 | 0.189 | 0.076 | 0.193 | 0.170 | 0.201 | |

State 3: Player II

| | | | | | | | | |
|---|-------|-------|-------|-------|-------|-------|-------|---|
| 1 | 5.041 | 0.230 | 0.184 | 0.213 | 0.192 | 0.084 | 0.096 | |
| 2 | 5.005 | 0.266 | 0.129 | 0.118 | 0.222 | 0.120 | 0.145 | ← |

State 4: Player II

| | | | | | | | | |
|---|-------|-------|-------|-------|-------|-------|-------|---|
| 1 | 5.085 | 0.477 | 0.166 | 0.085 | 0.011 | 0.144 | 0.117 | |
| 2 | 5.010 | 0.196 | 0.296 | 0.262 | 0.012 | 0.054 | 0.180 | ← |
| 3 | 5.015 | 0.222 | 0.131 | 0.131 | 0.211 | 0.142 | 0.184 | |

State 5: Player I

| | | | | | | | | |
|---|-------|-------|-------|-------|-------|-------|-------|---|
| 1 | 5.031 | 0.247 | 0.133 | 0.167 | 0.023 | 0.206 | 0.223 | |
| 2 | 5.058 | 0.019 | 0.133 | 0.059 | 0.363 | 0.063 | 0.362 | |
| 3 | 5.098 | 0.315 | 0.157 | 0.175 | 0.002 | 0.308 | 0.043 | ← |

State 6: Player I

| | | | | | | | | |
|---|-------|-------|-------|-------|-------|-------|-------|---|
| 1 | 5.073 | 0.116 | 0.018 | 0.195 | 0.256 | 0.341 | 0.074 | ← |
| 2 | 5.023 | 0.133 | 0.186 | 0.223 | 0.093 | 0.062 | 0.302 | |

What follows is the sequence of policies f_0, f_1, \dots generated by the algorithm along with their respective values $\phi_\beta(f_i)$. $\phi_\beta(f_i)$ is actually a vector with six elements since it is indexed by the starting state. Thus, for clarity, we will write $\phi'_\beta(f_i) = -30000 + \sum_{t=1}^6 [\phi_\beta(f_i)(t)]$.

¹The original game had 4 actions in each state, but we deleted those actions for which the algorithm never passed through

| | |
|--------------------------|------------------------------|
| $f_0 = 1, 1, 1, 1, 1, 1$ | $\phi'_\beta(f_0) = 373.597$ |
| $f_1 = 2, 1, 1, 1, 1, 1$ | $\phi'_\beta(f_1) = 400.411$ |
| $f_2 = 2, 2, 1, 1, 1, 1$ | $\phi'_\beta(f_2) = 401.506$ |
| $f_3 = 2, 2, 2, 1, 1, 1$ | $\phi'_\beta(f_3) = 371.830$ |
| $f_4 = 2, 2, 2, 2, 1, 1$ | $\phi'_\beta(f_4) = 310.021$ |
| $f_5 = 2, 1, 2, 2, 1, 1$ | $\phi'_\beta(f_5) = 311.945$ |
| $f_6 = 2, 1, 2, 3, 1, 1$ | $\phi'_\beta(f_6) = 308.460$ |
| $f_7 = 2, 1, 2, 3, 2, 1$ | $\phi'_\beta(f_7) = 321.684$ |
| $f_8 = 2, 1, 2, 3, 3, 1$ | $\phi'_\beta(f_8) = 405.999$ |
| $f_9 = 2, 1, 2, 2, 3, 1$ | $\phi'_\beta(f_9) = 404.635$ |

So we find that the policy choosing actions 2,1,2,2,3,1 in respective states 1,2,3,4,5,6 is the saddlepoint policy. That is, Player I must choose actions 2,1,3,1 in the respective state 1,2,5,6 while Player II must choose action 2 in both state 3 and state 4.

7. An Open Question

It is only natural to try and extend this algorithm to the average reward payoff criterion

$$\phi(\pi, \rho)(t_0) = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=0}^N r_n(t_0, \pi, \rho).$$

This payoff was first introduced by Gillette[1957]. Liggett and Lippman[1969] showed that for perfect information games an equilibrium pair in pure stationary strategies exists under the average reward criterion so that one would expect there to be a policy-improvement approach to solving such a game. Using the average reward MDP policy improvement methods of Blackwell[1962] on m any randomly generated stochastic games with perfect information showed finite termination every time, but the authors were unable to prove that cycling will never occur.

In travelling such a path in the average reward case, one can conclude, using the results of Blackwell[1962] that a β -discounted path would be travelled for a β so close to 1 that the functions $\phi_\beta(f, g)$ don't "cross" each other any more for different pure stationary strategy pairs. Thus, if one were to prove the following conjecture of ours, the average reward policy improvement

path would never cycle and hence, would always find an average reward equilibrium pair:

Conjecture: Given a stochastic game Γ and a fixed $\beta \in [0, 1)$, the graph C_Γ contains no strict cycles, i.e. there does not exist a strictly increasing sequence $\alpha_0, \alpha_1, \dots, \alpha_k$ with $\alpha_k = \alpha_0$.

We conclude with a remark that the properties **F1** and **F2** are not sufficient to prove the conjecture. It is easy to find even a 3-dimensional full cube which contains a strict cycle (e.g. **Figure 1**).

References

Blackwell, D. [1962] : *Discrete Dynamic Programming*. Annals of Mathematical Statistics **33**, 719-726.

Gillette, D. [1957] : *Stochastic games with zero stop probabilities*, In: Dresher, M., A.W. Tucker & P. Wolfe (eds.), *Contribution to the theory of games, Vol. III*, Ann. of Math. Stud. 39, Princeton Univ. Press, Princeton.

Kallenberg, L.C. M. [1983]: *Linear Programming and Finite Markovian Control Problems*. Mathematical Centre Tract 148, Centre for Mathematics and Computer Science, Amsterdam.

Liggett, T.M. and S. A. Lippman [1969]: *Stochastic Games with Perfect Information and Time Average Payoff*. SIAM Review **11**, 604-607.

O.J. Vrieze [1983]: *Stochastic Games with Finite State and Action Spaces*, (Ph.D Thesis), Free University, Math. Centrum, Amsterdam.

Pollatschek and Avi-Itzhak [1969]: *Algorithms for Stochastic Games with Geometrical Interpretation* Management Science **15**, 399-415.

Raghavan and Filar [1991]: *Algorithms for Stochastic Games - A Survey* Methods and Models of Operations Research ZOR **35**:437-472.

Shapley, L.S. [1953]: *Stochastic Games* Proceedings of the National Academy of Sciences U.S.A. **39**, 1095-1100.

Van der Waal, J. [1977]: *Discounted Markov Games: Successive Approximations and Stopping Times* International J. Game Theory **6**, 11-22.