Econometrica, Vol. 71, No. 4 (July, 2003), 1239–1254

NOTES AND COMMENTS

COMPUTING SUPERGAME EQUILIBRIA[1]

BY KENNETH L. JUDD, SEVIN YELTEKIN, AND JAMES CONKLIN[2]

We present a general method for computing the set of supergame equilibria in infinitely repeated games with perfect monitoring and public randomization. We present a three-stage algorithm that constructs a convex set containing the set of equilibrium values, constructs another convex set contained in the set of equilibrium values, and produces strategies that support them. We explore the properties of this algorithm by applying it to familiar games.

KEYWORDS: Supergames, Nash equilibrium, computation, monotone convex set-valued operators.

1. INTRODUCTION

THE NATURE OF DYNAMIC STRATEGIC INTERACTION has been extensively studied in the repeated game literature. Abreu, Pearce, and Stacchetti (APS) (1986, 1990) developed set-valued techniques for solving repeated games with imperfect monitoring, showing that the set of sequential equilibrium payoffs is a fixed point of a monotone operator similar to the Bellman operator in dynamic programming. Cronshaw and Luenberger (1990) extended the APS analysis to games with perfect monitoring. More generally, the APS method can be applied to any problem reducible to finding the maximal fixed point of a monotone set-valued operator. However, these methods seldom produce closed-form solutions and are not directly implementable on a real computer because they require approximation of arbitrary sets. A numerical approach is necessary to quantitatively analyze APS-style models. This paper presents a method for computing subgame perfect equilibrium value sets and equilibrium strategies in infinitely repeated games with perfect monitoring.[3] These methods make it possible to quantitatively analyze a broad range of dynamic strategic problems.

The APS method is a constructive procedure; the equilibrium value set can be obtained as the limit of an iteration of a monotone operator. However, two chief numerical issues must be solved before we can implement this procedure on a computer. The first is the parsimonious numerical representation of a set; the second is the consistency of this representation with the underlying structure of the monotone operator. As a first step in handling these difficulties, we alter the supergame by introducing a public randomization device. We show that convex polytopes provide an efficient and consistent way of approximating the equilibrium value set of the modified supergame.

[1]This work was supported by NSF Grant SES-9012128, SES-9309613, and SES-9708991. This paper is an extension of Conklin and Judd (1993), which discussed only the inner ray approximation method presented in Section 3.4.

[2]We gratefully acknowledge the comments and suggestions of five anonymous referees, and two *Econometrica* editors.

[3]A longer version of this paper, with more explanation, details, figures, examples, and discussion of error analysis and the related literature, is available at http:\\bucky.stanford.edu.

We then produce a tight approximation of the equilibrium value set using iterative procedures that preserve critical monotonicity properties of the APS-style operator. The first step, the *outer approximation* step, finds a convex polytope that contains the equilibrium value set. The second step, the *inner approximation* step, finds a convex polytope contained in the equilibrium value set. These two polytopes provide bounds on the equilibrium value set. Any point contained within the lower bound is certainly an equilibrium payoff. Conversely, any point not contained in the upper bound is certainly not an equilibrium payoff. We then develop a third procedure, called the *ray method*, to compute equilibrium strategies that support arbitrary equilibrium payoffs.

We argue that one needs both an inner and outer approximation since only the difference between the two sets indicates the accuracy of the approximation. The ability to compute both upper and lower bounds on the equilibrium value set (where "upper" and "lower" refers to the partial ordering on sets induced by $\subseteq$) is an unusual feature of our method since few numerical algorithms deliver such bounds. Many numerical algorithms provide convergence theorems, but they generally do not deliver computable error bounds for any particular approximation in actual applications. We present examples that demonstrate the importance of having both inner and outer approximations and show that their difference is very small if one implements a sufficiently flexible version of our algorithm.

The APS method has been applied to a large variety of games studied in industrial organization, contract theory, and dynamic policy analysis.[4] This paper focuses on the case of infinitely repeated static games, but the methods are generalizable to dynamic problems. Conklin and Judd (1996) generalized these methods to dynamic games with a finite number of state variables. Sleet and Yeltekin (2002a) provide an extension to the more difficult case of computing equilibrium payoff correspondences of dynamic games with continuous state variables.

Section 2 presents the characterization of equilibrium we use. Section 3 presents the basic algorithm and its properties. Section 4 examines the algorithm's speed and accuracy with applications to a Prisoner's Dilemma game and a Cournot game. Section 5 concludes.

2. SUPERGAMES AND CHARACTERIZATION OF EQUILIBRIUM PAYOFFS

We examine an N-player infinitely repeated game with perfect monitoring. The actions of player i in the stage game are in A_i, $i = 1, \dots, N$, and each element of $A \equiv A_1 \times A_2 \times \cdots \times A_N$ is an action profile. Player i's payoff in the stage game is $\Pi_i: A \to R$. Let $a_{-i} \equiv (a_1, \dots, a_{i-1}, a_{i+1}, \dots, a_N)$ represent player i's opponents' actions. The payoff of the player i's best reply to a_{-i} is $\Pi_i^*(a_{-i}) \equiv \max_{a_i \in A_i} \Pi_i(a_i, a_{-i})$. We make the following standard assumptions.

ASSUMPTION 1: A_i, $i = 1, \dots, N$, *is a finite set.*

ASSUMPTION 2: *The stage game has a pure strategy Nash equilibrium.*

[4]See Atkeson (1991) for an extension of the APS theory to a game with a state variable in the context of international borrowing and lending.

We construct the supergame S^∞ for discount factor $\delta > 0$. The action space of S^∞ is $A^\infty \equiv \times_{i=1}^{\infty} A$. Let $a(t)$ be the time t action profile. We assume that player i aims to maximize his average discounted payoff

$$(1-\delta)\Pi_i(a(1)) + \delta\left[(1-\delta)\sum_{t=2}^{\infty}\delta^{t-2}\Pi_i(a(t))\right].$$

Note that the average discounted payoff is a convex combination of the first-period payoff, $\Pi_i(a(1))$, and the average discounted payoff of the rest of the game (the term in brackets), which is the *continuation payoff*. The average discounted payoff for agent i is bounded below by $\underline{\Pi}_i \equiv \min_{a\in A}\Pi_i(a)$ and above by $\overline{\Pi}_i \equiv \max_{a\in A}\Pi_i(a)$. Therefore the supergame payoffs are contained in the hypercube $\mathcal{W} = \times_{i=1}^{N}[\underline{\Pi}_i, \overline{\Pi}_i]$. Let $V^P \subset \mathcal{W}$ denote the set of all subgame perfect equilibrium payoffs of S^∞.

We follow the recursive approach developed by APS and applied by Cronshaw and Luenberger (1990) to perfect information games. In the recursive formulation of the problem, each subgame perfect equilibrium payoff vector is supported by a profile of current actions consistent with Nash play in the current period and a vector of continuation payoffs that are themselves payoffs in some subgame perfect equilibrium. The key to finding V^P is the construction of *self-generating* sets. The concept of self-generation can be formulated using the operator, B^P, defined for $W^P \subseteq \mathcal{W}$ by

$$B^P(W^P) = \bigcup_{(a,w)\in A\times W^P}\{(1-\delta)\Pi(a) + \delta w \mid \forall i\ (IC_i \geq 0),\ i=1,N\} \tag{1}$$

where

$$IC_i \equiv ((1-\delta)\Pi_i(a) + \delta w_i) - ((1-\delta)\Pi_i^*(a_{-i}) + \delta \underline{w}_i) \geq 0$$

is the incentive compatibility condition for player i, and $\underline{w}_i \equiv \inf_{w\in W^P} w_i$ is player i's minimum possible continuation value in W^P. A vector $b = (1-\delta)\Pi(a) + \delta w$ is in $B^P(W^P)$ if there is some action profile, $a \in A$, and continuation payoff, $w \in W^P$, that satisfy the players' incentive compatibility constraints, and deliver b as current payoff. A set W^P is self-generating if $W^P \subseteq B^P(W^P)$. Straightforward extensions of the arguments in APS can be used to establish that the operator B^P is monotone in W^P and preserves compactness. Cronshaw and Luenberger (1990)[5] show that V^P is self-generating, repeated application of B^P to $\mathcal{W}$ converges to V^P, and V^P is the largest bounded fixed point of the operator B^P.

Any numerical implementation of $B^P(W^P)$ requires an efficient representation of the set W^P. As a first step in this direction, we alter the supergame by including public randomization. More precisely, we assume that in each repetition of the game there is a lottery depending on the players' current actions that determines which Nash equilibrium will be played in the next period. Strategies in the supergame with public randomization condition player choices on histories of both actions and lottery outcomes. If W^P is the set of possible continuation values at time t of S^∞ without public randomization, then $co(W^P)$ is the set of possible time t ex-ante (before the outcome of the lottery) continuation values for S^∞ with public randomization. Then $B^P(co(W^P))$ is the

[5]Cronshaw and Luenberger (1994) was the published version of this paper, but examined only the strongly symmetric subgame perfect equilibria.

set of equilibrium values at t, and $co(B^P(co(W^P)))$ is the set of ex-ante continuation values available at $t-1$ for S^∞ with public randomization.

Let V be the set of ex-ante continuation values that can occur in equilibrium for the supergame S^∞ with public randomization. Therefore, V is convex and bounded. With minor modifications of the arguments presented by Cronshaw and Luenberger (1990, 1994), it can be shown that repeated application of the APS-style operator with public randomization produces a sequence of convex sets that converge to the equilibrium payoff set. More precisely, if B is

$$B(W) = co(B^P(co(W))), \quad W \subseteq \mathcal{W}, \tag{2}$$

then B is monotone in W (i.e., if $W \subseteq W'$ then $B(W) \subseteq B(W')$), V is the largest fixed point of B, and if $W_0 = \mathcal{W}$ and $W_{i+1} = B(W_i)$, then $V = \bigcap_i W_i$. Whereas standard APS analysis focuses on fixed points of $B^P(\cdot)$, we focus on fixed points of the convex-valued operator $B(\cdot)$.[6]

3. APPROXIMATIONS OF $B(W)$ AND V

We use two kinds of convex polytope approximations of convex sets W: inner and outer approximations. Inner approximations are convex hulls of points on the boundary of W and outer approximations are polytopes defined by supporting hyperplanes of W. These are precisely defined below.

DEFINITION 1: If $Z \subset W \subset R^N$ is a set of m points, then the *inner approximation* to W generated by Z is $co(Z)$.

DEFINITION 2: If Z is a set of m points on the boundary of a convex set $W \subset R^N$ and $G \subset R^N$ a set of m corresponding subgradients oriented such that $(z^\ell - w) \cdot g^\ell > 0$ for $w \in W$, then the *outer approximation* of W generated by (Z, G) is

$$\widehat{W} = \bigcap_{\ell=1}^{m} \{z \in R^N \mid g_\ell \cdot z \leqq g_\ell \cdot z_\ell\}. \tag{3}$$

The critical property of B for our purposes is that it maps convex sets to convex sets and that it is monotone. In particular, $B(W)$ maps the collection of sets $\mathfrak{W} = \{W \subset R^N \mid W \text{ convex}, W \subset \mathcal{W}\}$ into itself. We define inner and outer monotone approximations of the operator $B(\cdot)$ that preserve these critical properties.

DEFINITION 3: A mapping $B^I: \mathfrak{W} \to \mathfrak{W}$ $(B^O: \mathfrak{W} \to \mathfrak{W})$ is an *inner (outer) monotone approximation* of B if:

1. for all $W \in \mathfrak{W}$, $B^I(W) \subseteq B(W)$ $(B^O(W) \supseteq B(W))$, and
2. for all $W, W' \in \mathfrak{W}$, if $W \subseteq W'$ then $B^I(W) \subseteq B^I(W')$ $(B^O(W) \subseteq B^O(W'))$.

The definitions of inner and outer monotone approximations directly imply Proposition 4, which states that B^O inherits some properties of B, and relates the maximal fixed points of B^O to V, the maximal fixed point of B.

[6]Public randomization is now a standard technique to simplify the analysis; see, e.g., Cronshaw (1997) and Phelan and Stacchetti (2001).

PROPOSITION 4: *Suppose $B^O(\cdot)$ is an outer monotone approximation of $B(\cdot)$. Then the maximal fixed point of B^O contains V. More precisely, if $W \supseteq B^O(W) \supseteq V$, then $B^O(W) \supseteq B^O(B^O(W)) \supseteq \cdots \supseteq V$.*

PROOF: The proof is a slight modification of the proof of Cronshaw and Luenberger (1990) and can be summarized as follows. Since B^O is increasing on $\mathfrak{W}$, and $\mathfrak{W}$ is a complete lattice, by Tarski's theorem its maximal fixed point is

$$V^* = \bigcup_{W \subseteq \mathcal{W},\ W \subseteq B^O(W)} W.$$

By definition of B^O, $V = B(V) \subseteq B^O(V)$; therefore, $V \subseteq V^*$. *Q.E.D.*

Lemma 5 tells us that $\mathcal{W}$ is an appropriate initial guess for all games.

LEMMA 5: $\mathcal{W} \supseteq B^O(\mathcal{W}) \supseteq V$.

PROOF: By definition of $\mathcal{W}$ and $B^O(\cdot)$. *Q.E.D.*

3.1. *Inner Hyperplane Approximation Method*

The key to approximating $B(W)$ is to fix some subgradients $H \subset R^N$ (we call them *search subgradients*) and locate boundary points x of $B(W)$ where the subgradient of $B(W)$ at x is in H. We first examine an inner monotone approximation of $B(W)$ since self-generating sets of inner approximations of $B(W)$ are equilibrium values whereas the maximal fixed point of an outer approximation may contain values other than equilibrium payoffs. Algorithm 1 defines the inner monotone approximation $B^I(W; H)$ for a set H of search subgradients. The input for Algorithm 1 is a set of vertices Z such that $W = co(Z)$. The key computation occurs in Step 1. For each search subgradient h_ℓ and action profile a, Step 1a finds a continuation value $w \in W$ that makes a incentive compatible and maximizes a weighted sum of player payoffs where the weights are given by h_ℓ. Furthermore, (4) is a linear programming problem, implying that the approximation error in solving (4) is close to machine epsilon (10^{-16} for double precision arithmetic). Step 1b finds the action profile a_ℓ^* that maximizes the h_ℓ-weighted payoffs, and also records the maximized weighted payoff, z_ℓ^+. Step 2 collects the z_ℓ^+ points into Z^+, which are all in $B(W)$. Therefore, $W^+ \equiv co(Z^+)$ is our inner approximation.

Lemma 6 presents the basic properties of Algorithm 1.

LEMMA 6: *For any set of subgradients $H \subset R^N$,*
1. *$B^I(W; H)$ is an inner monotone approximation of $B(W)$;*
2. *if $H \subseteq H'$ then $B^I(W; H) \subseteq B^I(W; H')$.*

PROOF: The equations defining Algorithm 1 show that it computes points on the boundary of $B(W)$ with subgradients H. Since Step 2 takes the convex hull of these points, $B^I(W; H) \subseteq B(W)$. If $W' \subseteq W$, the only part of (4) that differs for $W' \neq W$ is the list of constraints implicit in (i).[7] Since they are tighter for $W' \subset W$, the solution

[7]In practice, constraint (i) is replaced by a set of linear constraints on w. For an inner approximation, these linear constraints are derived from the piecewise linear approximation of $co(Z)$ in each iteration.

ALGORITHM 1: MONOTONE INNER HYPERPLANE APPROXIMATION $B^I(W;H)$

Input: Vertices $Z=\{z_1,\ldots,z_M\}$ such that $W=co(Z)$.

Step 1: Find extremal points of $B(W)$. For each search subgradient $h_\ell\in H$, $\ell=1,\ldots,L$.

(a) For each $a\in A$, solve

$$c_\ell(a)=\max_w h_\ell\cdot[(1-\delta)\Pi(a)+\delta w],\quad \text{such that} \tag{4}$$

(i) $w\in W$,

(ii) $(1-\delta)\Pi^i(a)+\delta w_i\geq(1-\delta)\Pi_i^*(a_{-i})+\delta\underline{w}_i,\qquad (i=1,\ldots,N)$,

where $c_\ell(a)=-\infty$ if no w satisfies the constraints. Let $w_\ell(a)$ be a w value that solves (4).

(b) Find best action profile $a\in A$ and corresponding continuation value:

$$a_\ell^*=\arg\max\{c_\ell(a)\mid a\in A\},$$

$$z_\ell^+=(1-\delta)\Pi(a_\ell^*)+\delta w_\ell(a_\ell^*).$$

Step 2: Collect set of vertices $Z^+=\{z_\ell^+\mid \ell=1,\ldots,L\}$, and define $W^+=co(Z^+)$.

---

for W' will produce smaller $c_\ell(a)$ results, which in turn produce smaller output sets. Therefore, $B^I(W';H)\subseteq B^I(W;H)$. Part 2 follows from the observation that increasing the number of search gradients in Algorithm 1 increases the number of boundary points from $B(W)$ in Z^+, which enlarges $co(Z^+)$. *Q.E.D.*

Our inner approximation algorithm for approximating V chooses some set of search subgradients H and executes the iteration $W_{i+1}=B^I(W_i;H)$ until $d(W_i,W_{i+1})<\varepsilon$, where ε is the stopping criterion, and $d(W_i,W_{i+1})$ is defined as

$$d(W_i,W_{i+1})=\max\left\{\max_{z\in Z_i}\min_{z^+\in Z_{i+1}}\|z-z^+\|,\ \max_{z^+\in Z_{i+1}}\min_{z\in Z_i}\|z-z^+\|\right\}$$

where $Z_i(Z_{i+1})$ is the set of extreme points of $W_i(W_{i+1})$.[8]

Theorem 7 presents the critical limit theorem for iterations of inner approximations.

THEOREM 7: *Define the iteration $W_{i+1}=B^I(W_i)$. If $W_0\subseteq\mathcal{W}$ contains a static Nash pure strategy equilibrium value, then $\varnothing\neq W_\infty=\limsup W_i\subseteq V$. If $W_0=\mathcal{W}$, then W_i is a monotonically decreasing sequence.*

PROOF: W_∞ is not empty since any pure strategy Nash value in W_0 is also in each W_i. Both B and B^I are monotonic; therefore, $B^I(W)\subseteq B(W)$ implies $B^I(W_0)\subseteq B^I(\mathcal{W})\subseteq B(\mathcal{W})$. By induction, $W_\infty\subseteq\lim_{i\to\infty}B^i(\mathcal{W})=V$. If $W_0=\mathcal{W}$, then $W_1=B^I(W_0)\subseteq B(W_0)\subseteq W_0$, and by induction, $W_{i+1}\subseteq W_i$ for all i. *Q.E.D.*

Theorem 7 provides a sufficient condition for the limit of inner monotone approximation iterates to be contained in V. If $B^I(W_0)\subseteq W_0$ (such as $W_0=\mathcal{W}$), then W_i is a monotonically decreasing sequence. One conventional computational approach would

[8]The Hausdorff distance between W_i and W_{i+1} is bounded above by $d(W_i,W_{i+1})$ because each face of $co(Z_i)$ and $co(Z_{i+1})$ is a convex combination of the vertices in Z_i and Z_{i+1}. Therefore, if the two sets are close in $d(.,.)$ then they are close in the Hausdorff metric.

continue until $d(W_i, W_{i+1}) < \varepsilon$ at, say, $\widehat{W}$. While the limit $W_\infty \subseteq V$, this may not be true for $\widehat{W}$. For some purposes, it may be acceptable to take such a $\widehat{W}$ as an approximate set of equilibrium values.

Since we are interested in error bounds, we pursue another approach to find an approximation to V that is contained in V after a finite number of iterations. Proposition 8 presents a sufficient condition for $W \subseteq V$. In particular, if $W \subseteq B^I(W)$ for some inner monotone operator B^I, then $W \subseteq V$. While there is no guarantee of success, we find that it is often possible to find such a W (the details are given below) and construct an inner approximation $W^I \subseteq V$.

PROPOSITION 8: *Suppose B^I is an inner monotone approximation of B. If $W \subseteq B^I(W)$, then $B^I(W) \subseteq B^I(B^I(W)) \subseteq \cdots \subseteq V$.*

PROOF: Monotonicity of B^I implies that if $W \subseteq B^I(W)$, then $B^I(W) \subseteq B^I(B^I(W))$, etc. By definition of B^I, if $W \subseteq B^I(W)$, then $W \subseteq B(W)$, which implies that $W \subseteq V$. Then, monotonicity implies $B^I(W) \subseteq B(W) \subseteq B(V) = V$, $B^I(B^I(W)) \subseteq B(B^I(W)) \subseteq B(V) = V$, etc. *Q.E.D.*

Propositions 4 and 8 show that the monotonicity properties are critical for computing reliable approximations to V. We now present three examples of approximations to $B(W)$ that satisfy these properties.

3.2. *Outer Hyperplane Approximation*

Once we find a set W^I and subgradients H such that $W^I \subseteq B^I(W^I; H)$ we know that $W^I \subseteq V$. However, W^I may be significantly smaller than V. We next construct an outer monotone approximation of B, $B^O: \mathfrak{W} \to \mathfrak{W}$, and use it to find a set W^O such that $V \subseteq W^O$. For a set of subgradients H, our monotone hyperplane approximation procedure is the same as Algorithm 1 except for Step 2. To construct an outer approximation, we take each point z_ℓ^+, and construct the hyperplane through z_ℓ^+ with normal h_ℓ. The output polytope, $B^O(W; H)$, is the intersection of the half-spaces defined by these hyperplanes.[9]

Lemma 9 presents the basic properties of our outer hyperplane approximation to B, and its proof is analogous to that of Lemma 6.

LEMMA 9: *For any set of search subgradients $H \subset R^N$,*
1. *$B^O(W; H)$ is an outer monotone approximation of $B(W)$;*
2. *if $H \subset H'$, then $B^O(W; H') \subseteq B^O(W; H)$.*

PROOF: The equations defining Algorithm 1 show that it computes points on the boundary of $B(W)$ with subgradients H. Since Step 2 uses an outer approximation construction, $B^O(W; H) \supseteq B(W)$. If $W' \subset W$, the only part of (4) that differs for $W' \neq W$ is the list of constraints implicit in (i). Since they are tighter for $W' \subset W$, the solution

[9]Cronshaw (1997) proposed an outer approximation method alternative to the inner method exposited in Conklin and Judd (1993). He examined games with continuous strategies and closed-form best reply functions. He acknowledged that applying these methods to general continuous strategy games creates global optimization problems that are difficult to solve. Therefore, we stay with discrete strategies in our effort to be general.

for W' will produce smaller $c_\ell(a)$ results, smaller values of c_ℓ^+,[10] which in turn, since the approximation subgradients of W^+ are unaffected, produce smaller output sets. Therefore, $B^O(W'; H) \subseteq B^O(W; H)$. Part 2 is similarly obvious. *Q.E.D.*

To find a W^O that contains V, we first need a W_0 such that $W_0 \supseteq B^O(W_0) \supseteq V$ and then compute $W_{i+1} = B^O(W_i; H)$, until $d^O(W_i, W_{i+1}) < \varepsilon$ where

$$d^O(W_i, W_{i+1}) = \min_\ell |(z_{i,\ell} - z_{i+1,\ell}) \cdot h_\ell|$$

and ε is the stopping criterion.

3.3. *Approximating V and Computing an Error Bound*

In approximating V, we proceed in two stages. First, we compute an outer approximation W^O. Second, we use W^O to construct an initial guess W_0 for the iteration $W_{i+1} = B^I(W_i; H)$. We want $W_0 \subseteq V$ since that, by Proposition 8, guarantees $W_i \subseteq V$ for all i. We construct W_0 by shrinking the Pareto frontier of W^O by a small amount (2–3%); in our examples, this set satisfies $B^I(W_0; H) \supseteq W_0$. Once an appropriate W_0 is constructed by this, or some other method,[11] the algorithm then computes $W_{i+1} = B^I(W_i; H)$ until $d(W_i, W_{i+1}) < \varepsilon$. The result, W^I, is an inner approximation of V.

We then compute an error bound for our approximations of V by computing the distance between W^I and W^O. The most natural norm for computing the distance between two sets, X and Y, is the Hausdorff norm defined as

$$\rho(X, Y) = \max_{x \in X} \min_{y \in Y} \|x - y\|.$$

Since $W^I \subseteq V \subseteq W^O$, we know $\rho(W^I, W^O) \geq \rho(V, W^I), \rho(V, W^O)$. Therefore $\rho(W^I, W^O)$ is a bound on the error, and since W^I and W^O are convex polytopes,

$$\rho(W^I, W^O) = \max_{\tilde{w} \in T^O} \min_{\hat{w} \in W^I} \|\tilde{w} - \hat{w}\|$$

where T^O is the set of vertices of W^O. Thus, $\rho(W^I, W^O)$ can be computed by solving L (number of search gradients and vertices) concave C^∞ minimization problems with linear constraints. Lemma 6 and Lemma 9 imply that the error bound $\rho(W^I, W^O)$ is decreasing in the number of search gradients used. If the error bound is unacceptably large, it can be reduced by using a larger number of search subgradients.

We advocate computing both an outer and inner approximation. An outer approximation $W^O \supseteq V$ would suffice if the objective is to show that a particular equilibrium value is not part of the equilibrium value set. If instead one wants to find values in V, W^O is not an appropriate approximation since it almost surely contains points not in V. In particular, the boundary of W^O will contain few, if any, points on the boundary of V. Since applications will often want to find points that can be supported in equilibrium, it would be generally wrong to treat points on ∂W^O as equilibrium values. Thus an inner approximation $W^I \subseteq V$ is necessary. However, an inner approximation

[10] c_ℓ^+ is defined as $c_\ell^+ = \max_{a \in A}\{c_\ell(a)\}$.

[11] This slicing technique always worked in our examples. An alternative choice for W_0 would be a set of equilibrium payoffs with simple strategies (such as constant actions) supported by Nash reversion.

$W^I \subseteq V$ may miss some equilibrium values. The pair (W^I, W^O) together is useful since they bracket V, $W^I \subseteq V \subseteq W^O$ and they provide an error bound. If W^O/W^I is small, then W^I contains most values in V.

Others have offered partial alternatives to the foregoing. In his extension of Conklin and Judd (1993), Cronshaw (1997) proposed using the outer approximation, coupled with Newton's method to approximate V. Since there may be many fixed points to $B^O(W; H)$ and since Newton's method does not involve a monotonic operator, his solution may not be the maximal fixed point of $B^O(W; H)$ and may miss some points in V. In their alteration of Conklin and Judd (1996), Phelan and Stacchetti (1999, 2001) aimed to find an outer approximation of a correspondence. However, their linear interpolation step for the capital stock k outside of its grid is inconsistent with computing an outer approximation, and their final approximation may therefore be neither an inner nor an outer approximation. Chang (1998) used discretization to produce at best an inner approximation; like any discretization approach, this approach suffers from the curse of dimensionality. None of these alternative methods include a procedure for computing error bounds. See the working paper version of this paper (Judd, Yeltekin, and Conklin (2002)) for more discussion of these comparisons.

3.4. *Computing Actions and Strategies: Inner Ray Approximation Method*

The hyperplane steps produce inner and outer approximations of the equilibrium value set, but they provide limited information about the actions and strategies that support those values. Our inner hyperplane step identifies equilibrium actions only at extreme points of the equilibrium value set, but the construction of strategies requires finding actions at a variety of points on the boundary of W^I. A third step, called the *ray method*, takes W^I and produces an inner approximation of its image under $B(\cdot)$ and action profiles $a \in A$ and continuation values in W^I that support arbitrary values on ∂W^I. The idea of the ray method is to choose a point in $w^0 \in B(W)$, compute points $Z^+ \subset \partial B^P(W)$ that lie on a finite number of rays, Θ, emanating from w^0, and then report $co(Z^+)$ as an inner approximation, denoted $B^R(W; w^0, \Theta)$, to $B(W)$. To compute equilibrium actions and continuation values for $w \in \partial W$, the set Θ must include $(w - w^0)/\|w - w^0\|$. The ray procedure is presented in Algorithm 2.

ALGORITHM 2: INNER RAY APPROXIMATION $B^R(W; w^0, \Theta)$

Input: Vertices $Z = \{z_1, \dots, z_L\}$ such that $W = co(Z)$.
Initialize: Choose origin, $w^0 \in B(W)$, and M points, $\Theta \subset R^N$, on the unit sphere.

Step 1: Find new extremal points of $B(W)$. For each $\theta_m \in \Theta$:
(a) For each action profile $a \in A$ solve

$$\lambda_m(a) = \max_{\lambda > 0,\, w} \lambda, \quad \text{such that} \tag{5}$$

(i) $w = \delta^{-1}[(w^0 + \lambda\theta_m) - (1-\delta)\Pi(a)]$,
(ii) $w \in W$,
(iii) $(1-\delta)\Pi^i(a) + \delta w_i \geq (1-\delta)\Pi_i^*(a_{-i}) + \delta \underline{w}_i, \quad \forall i$,

where $\lambda_m(a) = -\infty$ if there is no w satisfying the constraints.
(b) Set $\lambda_m^* = \max_{a \in A} \lambda(a)$, $a_m^* = \arg\max_{a \in A} \lambda(a)$, and $z_m^+ = w^0 + \lambda_m^* \theta_m$.

Step 2: Collect set of points $Z^+ = \{z_m^+ \mid m = 1, \dots, M\}$ and define $W^+ = co(Z^+)$.

TABLE I
PRISONER'S DILEMMA
PAYOFFS

		Player 2: C	Player 2: D
Player 1:	C	4,4	0,6
	D	6,0	2,2

The point $z_m^+ \in Z^+$ on the ray $\overrightarrow{w^0\theta_m}$ is supported by the action profile a_m^* and (ex-ante, before the outcome of the lottery) the continuation value $w = \delta^{-1}[z'_m - (1-\delta)\Pi(a_m^*)]$. To construct a sample equilibrium path, we need to compute how the continuation value w is supported. If w corresponds to a point on the boundary associated with a particular search gradient, the action profile and continuation values that support w are already determined by the inner ray method. If $w \notin \partial W$ however, w can be represented as a convex combination of vertices of W^+ that are supported by pure strategies. In this case, we use a random number generator to simulate the public randomization device and determine the equilibrium play. Of course, there are multiple solutions: our programs make no attempt to list all equilibria. Below we use the ray method to compute sample action paths and strategies for some familiar games.[12]

Our complete algorithm integrates the outer hyperplane, inner hyperplane, and ray methods into a three-stage convex polytope algorithm. As our initial guess, we use $\mathcal{W}$. Generally, we use a uniformly distributed set of search gradients. Once our three-stage algorithm is completed, we then compute the error measure, $\rho(W^I, W^O)$, as described above.

4. EXAMPLES

We test our algorithms by applying them to well-understood problems with known solutions. Our primary aim is to quantitatively demonstrate some of the properties of the recursive value set algorithm, such as monotonic convergence and the increase in precision from additional search subgradients.

Our first example is the Prisoner's dilemma game. Each player chooses either cooperate (C) or defect (D) and payoffs are in Table I. Figure 1 displays some early iterates $W_{i+1} = B^O(W_i; H)$ and V for the Prisoner's dilemma game. The monotonicity property of B^O is apparent; the iterates are nested and become smaller, converging to V. Figure 2 displays the solutions using 8, 24, and 72 search gradients. The polygon $ABCDEF$ represents both the inner and outer approximations with 72 uniformly distributed subgradients. The outermost lines represent the value set after convergence with 8 subgradients and include many points not in the 24- and 72-gradient approximations, and, hence, not part of the equilibrium. In this example, the difference between the inner and outer approximations with $L = 72$ cannot be visually detected and together, they represent a very good approximation of the equilibrium value set.

We next present solutions to a Cournot duopoly game. We assume q_i is firm i's sales, c_i is firm i's unit cost, p_i is firm i's price, and $\Pi_i(q_1, q_2)$ is firm i's profits.

[12]We could use the ray method as our inner approximation procedure (as was done in Conklin and Judd (1993)) but the hyperplane approach we present here has some technical advantages over the ray method; see the working paper version (Judd, Yeltekin, and Conklin (2002)) for more details.

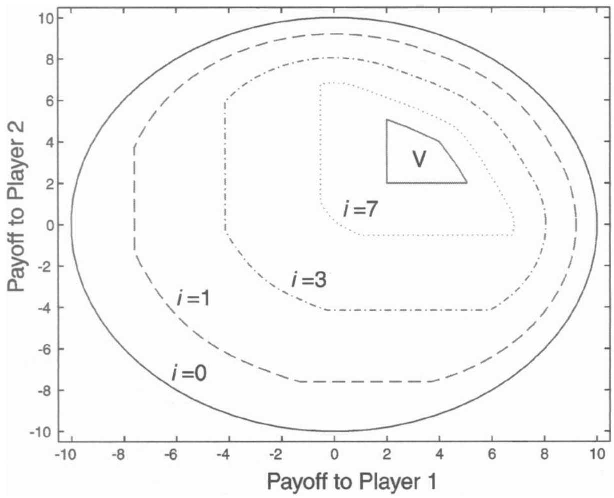

FIGURE 1.—Monotonic convergence of outer approximations.

We assume a linear demand function, $p = \max\{6 - q_1 - q_2, 0\}$; the profit function is $\Pi_i(q_1, q_2) = q_i(p - c_i)$. We discretize the action space; firms can choose one of 15, uniformly distanced actions from the set [0, 6].

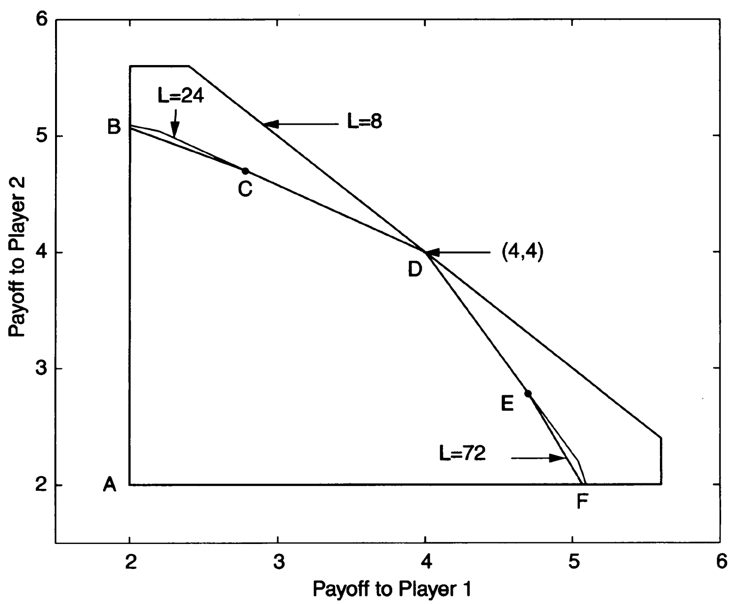

FIGURE 2.—Value sets with different subgradients.

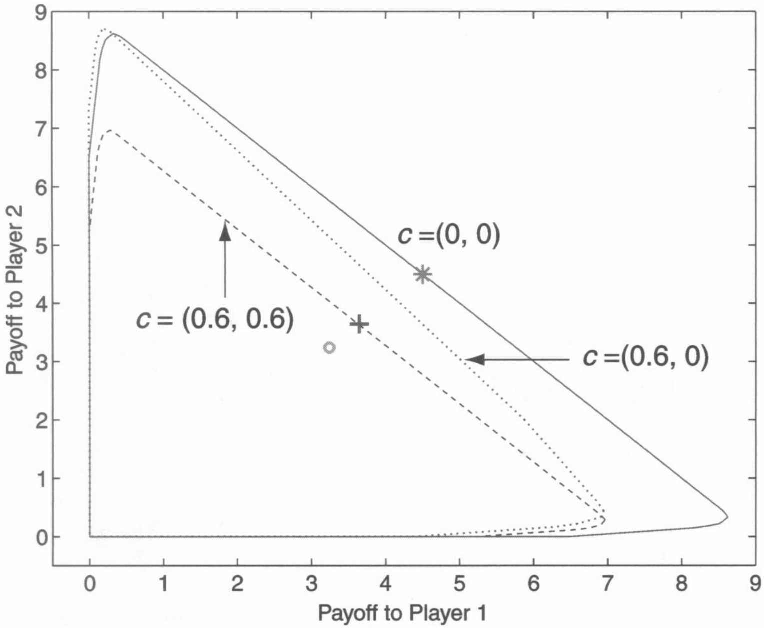

FIGURE 3.—Cournot duopoly with alternative cost assumptions.

Figure 3 displays V for $c_i \in \{0.0, 0.6\}$. In all cases, the inner and outer approximations are indistinguishable at the resolution of the figures. If both firms have zero costs, a Cournot–Nash equilibrium payoff is (4, 4)[13] and the shared monopoly payoff is (4.5, 4.5) ('*' in Figure 3). If $c_i = 0.6$, $i = 1, 2$, then the shared monopoly payoff is (3.64, 3.64) ('+' in Figure 3) and the unique Nash equilibrium payoff is (3.24, 3.24) ("○" in Figure 3). The set of equilibrium values in the supergame is quite large in both cases. When costs are positive, the threats are far worse than Nash–Cournot reversion.

Figure 4 and Table II describe the actions that support equilibrium values of the Cournot game with $c = (0.6, 0.6)$. Figure 4 illustrates the relation between equilibrium values (labelled by "*") and continuation values (labelled by "○") indicated by arrows. Table II displays information for 7 points from Figure 4.[14] In Table II, each ℓ corresponds to the point marked ℓ in Figure 4, $v_i(\ell)$ is player i's equilibrium value at point ℓ, $w_i(\ell)$ is his continuation value, q_i is his current output, and $\Pi_i(q_1, q_I)$ is player i's current profit. The results displayed in Figure 4 and Table II indicate that points on southern (and western) extreme points of V, the punishment points, usually involve "going along with your own punishment" as in Abreu (1988). At these points, at least one player makes losses in the current period. This is rational for each player because of the high continuation values. For example, at point 46, both firms produce $q = 5.1$,

[13]The point (0, 0) is also a Nash equilibrium value in the zero cost case because playing (q_1, q_2), is a Nash equilibrium for $q_1, q_2 > 6$.

[14]The ℓ indices indicate the direction of a ray in our ray approximation. For a total of M rays, ray $\ell \in \{1, \ldots, M\}$ corresponds to the vector $(\cos\theta, \sin\theta)$, $\theta = (\ell - 1)2\pi/M$.

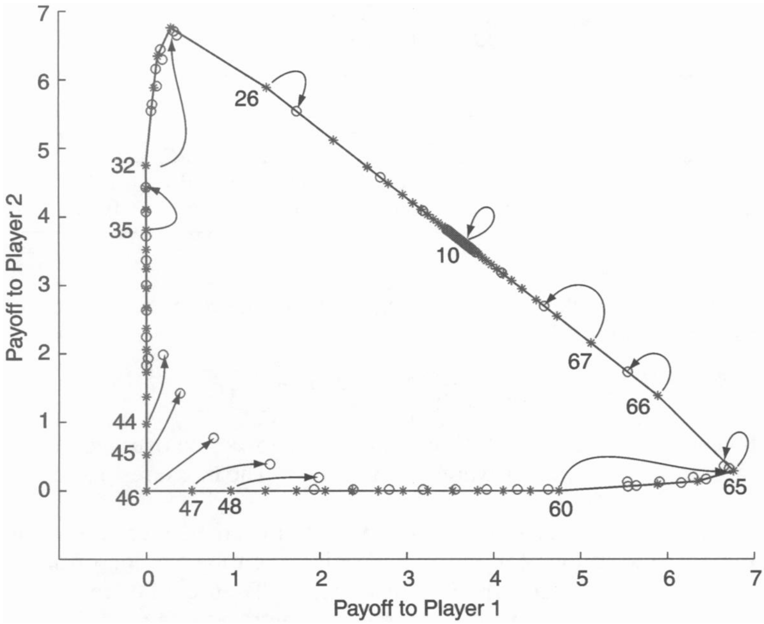

FIGURE 4.—Equilibrium paths in a Cournot game.

incur losses of 3.0 each, but enjoy an expected continuation value of 0.77. Table II also shows that asymmetric harsh punishments yield equally asymmetric continuation values; for example, point 60 corresponds to a positive value for Firm 1, but zero value for Firm 2. This equilibrium payoff pair is implemented by quantities of (5.1, 2.1) in the current period, and large profits in the future for Firm 1 and small future profits for Firm 2. Point 60 also corresponds to the payoffs associated with the punishment phase after Firm 2 deviates from cooperation.

Table III displays values of $\rho(W^I, W^O)$ for our examples with various numbers of search subgradients. The $\rho(W^I, W^O)$ values look large but that is due to the maximin nature of ρ. Specifically, if there is only one vertex of W^O that is r away from W^I, then $\rho(W^I, W^O) \geq r$. In our examples $\rho(W^I, W^O)$ is determined by the southeast and northwest corners of W^I and W^O, but W^O and W^I are always much closer on average.

TABLE II

ACTIONS, PROMISES, AND THREATS ON THE BOUNDARY OF V, $c = 0.6$

ℓ	$(v_1(\ell), v_2(\ell))$		$(w_1(\ell), w_2(\ell))$		(q_1, q_2)		$\Pi(q_1, q_2)$	
2	3.97	3.30	3.75	3.52	1.7	0.9	4.8	2.4
8	3.71	3.57	3.72	3.55	1.3	1.3	3.6	3.6
10	3.64	3.64	3.64	3.64	1.3	1.3	3.6	3.6
27	0.29	6.76	0.36	6.65	0.0	3.0	0.0	7.1
46	0.00	0.00	0.77	0.77	5.1	5.1	−3.0	−3.0
60	4.75	0.00	6.71	0.32	5.1	2.1	−3.0	−1.3

TABLE III
ERROR BOUNDS, $\varepsilon = 10^{-4}$

Game	Prisoner's Dilemma			Cournot		
Search Gradients	16	24	72	16	24	72
Hausdorff norm, ρ	0.2621	0.1246	0.0841	0.1790	0.1718	0.0943
Average distance, $\bar{\rho}$	0.1021	0.0633	0.0146	0.0436	0.0341	0.0176

We make this precise by defining the average distance between the vertices of W^O and the inner approximation W^I in

$$\bar{\rho}(W^I, W^O) = \frac{1}{\#\{\tilde{w} | \tilde{w} \in T^O\}} \sum_{\tilde{w} \in T^O} \min_{\hat{w} \in W^I} \|\tilde{w} - \hat{w}\|.$$

Table III shows that vertices of W^O are quite close to W^I on average, which implies that the faces of W^O are also on average, close to W^I. The measures ρ and $\bar{\rho}$ are global indices.

Table IV reports the run times with different convergence criterion, number of search gradients and actions per player, and reflect the total amount of time for computing all approximations and equilibrium strategies. We use a Fortran program on a 550 MHz Pentium PC. With two players, the running times rise roughly quadratically as the number of discrete actions increase. The running times increase in the number of search subgradients because each iteration checks more directions and each polytope approximation uses more subgradients and therefore more constraints for the $w \in W$ condition. Stricter convergence criteria (smaller ε) slow down the algorithm only slightly; as we move from $\varepsilon = 10^{-5}$ to $\varepsilon = 10^{-7}$ running times are only doubled. Table IV also indicates that as the discount factor δ increases, the rise in running time is moderate: convergence takes at most double the time when δ is increased from 0.8 to 0.9. While the times in Table IV may seem long, they are acceptable considering that we are finding the set of all Nash equilibrium payoffs and doing so with high accuracy. Furthermore, application of standard acceleration methods (such as Gauss-Seidel iteration), utilization of parallel computing methods (which are directly applicable for this problem), and the use of faster computers will all substantially reduce the time cost.

These examples show that our algorithm can be used to approximate the set of all Nash equilibria of nontrivial games, compute an error bound on the approximation

TABLE IV
RUN TIMES FOR COURNOT GAME

Search Gradients	16	32	72	16	16
ε	10^{-5}	10^{-5}	10^{-5}	10^{-7}	10^{-5}
δ	0.8	0.8	0.8	0.8	0.9
Actions per player					
5	8 s[a]	36 s	4 m 57 s	15 s	11 s
10	28 s	1 m 34 s	17 m 7 s	50 s	45 s
15	63 s	4 m 46 s	44 m 53 s	1 m 59 s	1 m 54 s

[a] "x m y s" means "x minutes, y seconds."

error, compute strategies that support specific equilibria, and do so in an acceptable amount of time. Therefore, this approach and its refinements offer robust and practical methods for finding all Nash equilibrium values of finite-action supergames.

5. CONCLUSION

Dynamic strategic theories of Abreu, Pearce, and Stacchetti (1990) have been useful for qualitative purposes, but quantitative applications require reliable numerical methods. This paper presents an algorithm to solve discounted N-player finite-move supergames with perfect monitoring and public randomization. We approximate the set of equilibrium values with convex sets. This efficient approximation allows us to compute both inner and outer approximations of the set of equilibrium payoffs, which together produce an error bound. The ease with which we compute an error bound is unusual for numerical algorithms, and it is particularly valuable in this context where the key set-valued operator can only be shown to be monotone. The examples show that computational demands are reasonable given the task of finding all Nash equilibrium values. The methods outlined in this paper can be used to approximate the equilibrium value sets of many dynamic problems, as shown in Conklin and Judd (1996) and Sleet and Yeltekin (2002a)[15] and do so with high accuracy and moderate computational costs.

Hoover Institution, Stanford University, Stanford, CA 94305; judd@hoover.stanford.edu; http://bucky.stanford.edu,

Northwestern University, KSM-MEDS, 2001 Sheridan Rd., Evanston, IL 60208; s-yeltekin@kellogg.northwestern.edu; http://www.kellogg.nwu.edu/faculty/yeltekin/html/research/index.html,

and

West End Capital Management, 12 Church St., Suite 224, Hamilton HM 11, Bermuda; jconklin@westend.bm

Manuscript received December, 2000; final revision received October, 2002.

REFERENCES

ABREU, DILIP (1988): "On the Theory of Infinitely Repeated Games with Discounting," *Econometrica*, 56, 383–396.

ABREU, DILIP, DAVID PEARCE, AND ENNIO STACCHETTI (1986): "Optimal Cartel Equilibria with Imperfect Monitoring," *Journal of Economic Theory*, 39, 251–269.

——— (1990): "Toward a Theory of Discounted Repeated Games with Imperfect Monitoring," *Econometrica*, 58, 1041–1063.

ATKESON, ANDREW (1991): "International Lending with Moral Hazard and Risk of Repudiation," *Econometrica*, 59, 1069–1089.

CHANG, ROBERTO (1988): "Credible Monetary Policy in an Infinite Horizon Model: Recursive Approaches," *Journal of Economic Theory*, 81, 431–461.

CONKLIN, JAMES, AND KENNETH JUDD (1993): "Computing Supergame Equilibria," Mimeo, Hoover Institution.

——— (1996): "Computing Value Correspondences for Two-Player Games with State Variables," Mimeo, Hoover Institution.

[15]See Tsyvinski and Villaverde (2002), Sleet (2001, 2002), Sleet and Yeltekin (2002b) for successful applications of these methods to dynamic government policy games.

CRONSHAW, MARK (1997): "Algorithms for Finding Repeated Game Equilibria," *Computational Economics*, 10, 139–168.

CRONSHAW, MARK, AND DAVID G. LUENBERGER (1990): "Subgame Perfect Equilibria in Infinitely Repeated Games with Perfect Monitoring and Discounting," University of Colorado Working Paper.

——— (1994): "Strongly Symmetric Subgame Perfect Equilibrium in Infinitely Repeated Games with Perfect Monitoring and Discounting," *Games and Economic Behavior*, 6, 220–237.

JUDD, KENNETH, SEVIN YELTEKIN, AND JAMES CONKLIN (2002): "Computing Supergame Equilibria," Working Paper, Hoover Institution.

PHELAN, CHRISTOPHER, AND ENNIO STACCHETTI (1999): "Subgame-perfect Equilibria in a Ramsey Tax Model," Staff Report 258, Federal Reserve Bank of Minneapolis, http://minneapolisfed.org/research/sr/sr258.html.

——— (2001): "Subgame-perfect Equilibria in a Ramsey Tax Model," *Econometrica*, 69, 1491–1518.

SLEET, CHRISTOPHER (2001): "On Credible Monetary Policy and Private Information," *Journal of Economic Theory*, 99, 338–376.

——— (2002): "Optimal Taxation with Private Government Information," University of Iowa Mimeo.

SLEET, CHRISTOPHER, AND SEVIN YELTEKIN (2002a): "On the Computation of Value Correspondences," Mimeo, KSM-MEDS, Northwestern University.

——— (2002b): "Credible Monetary Policy with Private Government Preferences: U Turn if You Want to," University of Iowa and Northwestern University Mimeo.

TSYVINSKI, ALEH, AND JESUS FERNANDEZ-VILLAVERDE (2002): "Optimal Fiscal Policy in a Business Cycle Model without Commitment," UPENN and Minneapolis Fed. Working Paper.