Working Paper Series
No. 2016-05

A "Pencil Sharpening" Algorithm for Two Player Stochastic Games with Perfect Monitoring

Dilip Abreu, Benjamin Brooks, Yuliy Sannikov

February 11, 2016

Keywords: Stochastic game, perfect monitoring, algorithm, computation

JEL Codes: C63, C72, C73, D90

Becker Friedman Institute
for Research in Economics

Contact:
773.702.5599
bfi@uchicago.edu
bfi.uchicago.edu

A "Pencil Sharpening" Algorithm for Two Player Stochastic Games with Perfect Monitoring*

Dilip Abreu Benjamin Brooks Yuliy Sannikov

February 11, 2016

Abstract

We study the subgame perfect equilibria of two player stochastic games with perfect monitoring and geometric discounting. A novel algorithm is developed for calculating the discounted payoffs that can be attained in equilibrium. This algorithm generates a sequence of tuples of payoffs vectors, one payoff for each state, that move around the equilibrium payoff sets in a clockwise manner. The trajectory of these "pivot" payoffs asymptotically traces the boundary of the equilibrium payoff correspondence. We also provide an implementation of our algorithm, and preliminary simulations indicate that it is more efficient than existing methods. The theoretical results that underlie the algorithm also yield a bound on the number of extremal equilibrium payoffs.

Keywords: Stochastic game, perfect monitoring, algorithm, computation.
JEL classification: C63, C72, C73, D90.

*Abreu: Department of Economics, Princeton University, dabreu@princeton.edu; Brooks: Becker Friedman Institute and Department of Economics, University of Chicago, babrooks@uchicago.edu; Sannikov: Department of Economics, Princeton University, sannikov@princeton.edu. This work has benefitted from the comments of seminar participants at the Hebrew University and UT Austin. We have also benefitted from the superb research assistance of Moshe Katzwer and Mathieu Cloutier. Finally, we would like to acknowledge financial support from the National Science Foundation.

1 Introduction

This paper develops a new algorithm for computing the subgame perfect equilibrium payoffs for two player stochastic games with perfect monitoring. Specifically, we study the pure strategy equilibria of repeated games with a stochastically evolving state variable that determines which actions are feasible for the players, and, together with the chosen actions, induces the players' flow payoffs. The chosen actions in turn influence the future evolution of the state. This classical structure is used to describe a wide range of phenomena in economics and in other disciplines. The range of applications include: dynamic oligopoly with investment (in, e.g., capacity, research and development, advertising), markets for insurance against income shocks, and the dynamics of political bargaining and compromise (cf. Ericson and Pakes, 1995; Kocherlakota, 1996; Dixit, Grossman, and Gul, 2000).

Our work has three inter-related components: (i) we uncover new theoretical properties of the equilibria that generate extreme payoffs for a fixed discount factor, (ii) we use these properties to develop a new algorithm for calculating the set of all equilibrium payoffs, and (iii) we provide a user-friendly implementation that other researchers can use to specify, solve, and analyze their games of interest. Preliminary results indicate that our algorithm is significantly more efficient than previously known computational procedures.

The standard methodology for characterizing subgame perfect equilibrium payoffs for infinitely repeated games comes from Abreu, Pearce, and Stacchetti (1986, 1990), hereafter APS. They showed that the set of discounted payoffs that can arise in subgame perfect equilibria satisfies a recursive relationship, which is analogous to the Bellman equation from dynamic programming. This recursion stems from the fact that any equilibrium payoff can be decomposed as the flow payoff from one period of play plus the expected discounted payoff from the next period onwards, which, by subgame perfection, is also an equilibrium payoff. Just as the value function is the fixed point of a Bellman operator, so the equilibrium payoff set is the largest fixed point of a certain set operator, which maps a set of payoffs which can be promised as continuation utilities into a set of new payoffs which they generate. In addition, APS show that iterating this operator on a sufficiently large initial estimate will yield a sequence of approximations that asymptotically converges to the true equilibrium payoff set. Although APS wrote explicitly about games with imperfect monitoring and without a state variable, their results extend in an obvious way to the case of perfect monitoring and a stochastic state whose evolution is influenced by the players' actions.[1]

[1]For early extensions involving a state variable see Atkeson (1991) and Phelan and Stacchetti (2001). A more recent application is Hörner et al. (2011). For a more complete description of the self-generation methodology for stochastic games, see Mailath and Samuelson (2006).

The APS algorithm does not exploit the detailed structure of equilibria nor does it focus attention on equilibria that generate extreme payoffs.[2] In contrast, Abreu and Sannikov (2014), hereafter AS, provide an algorithm that does this for two-player *repeated* games perfect monitoring, that is, the same environment studied here but without the stochastic state. The algorithm of AS exploits the simple structure of the equilibria that attain extreme payoffs. Some extreme payoffs are generated with both players strictly preferring their first-period action over any deviations, while for other payoffs, at least one player is indifferent to deviating to another action. AS show that in the former case, the corresponding equilibrium involves the repetition of the same action profile in every period. In the latter case, it turns out that there are at most four payoffs that can be used as continuation values.[3] Thus, associated with each action profile there is at most one extremal equilibrium payoff when incentive constraints bind in the first period and at most one when incentive constraints are slack. This leads to an obvious bound on the number of extreme points when the action sets are finite as AS assume (and as we assume in this paper as well).

With the generalization to a stochastic game, there is not one set of equilibrium payoffs, but rather a set of such payoffs for each possible initial state. In this richer setting, simultaneously considering the generation of a particular *tuple*[4] of payoff vectors, one for each state, leads to significant and computationally useful insights. More specifically, we consider tuples of equilibrium payoff vectors that maximize the same weighted sum of players' utilities in every state, e.g., the tuple of equilibrium payoffs that maximize player 1's payoffs in each state. For a generic choice of weights, these maximal payoffs are unique and are in fact extreme points of their respective equilibrium payoff sets. We show that the equilibria that generate such maximal payoff tuples have significant structure. (Note that corresponding to any action pair a played in the first round of an equilibrium in some state s, the continuation state is random, and the incentive constraints now need to be expressed in terms of expected continuation values.) In particular, we show that the equilibria that generate these payoffs have a stationary structure until the first history at which an incentive constraint binds for some player, and the stationary choice of action profiles is common across all of the equilibria that generate maximal payoffs in the same direction.

[2]APS do not assume the existence of a public randomization device, so the set of equilibrium payoffs need not even be convex.

[3]Due to there being perfect monitoring and two players, the locus of continuation payoffs that make a given player indifferent to deviating is a line, and the intersection of that binding incentive constraint with the (convex) set of equilibrium payoffs that are incentive compatible for both players has at most two extreme points. There are therefore at most four extreme binding continuation values between the two players' incentive constraints, and it is one of these payoffs which must be generated by continuation play.

[4]Throughout our exposition, we will use the term *tuple* to denote a function whose domain is the set of states and the term payoff will usually refer to a vector specifying a payoff for each player.

To illustrate, suppose that in the initial period, the state is s and in the extremal equilibrium all players *strictly* prefer their equilibrium actions a_i over any deviation. Then it must be the case that if the state in the second period turns out (by chance) to also be s, exactly the same actions a_i must be played.[5] Moreover, suppose that the state switches to some s' at which incentive constraints are again slack, and then returns to s. Still, the players must use the original actions a_i. It is only after the state visits some s'' at which at least one player is indifferent to deviating that the stationarity property may break, and subsequent visits to s or s' may be accompanied by different equilibrium actions. This stationarity is reminiscent of the classical observation that Markov decision problems admit an optimal policy that is stationary (Blackwell, 1965). Furthermore, as in AS, there are still at most four payoffs that may be generated by continuation play when the actions a are played in the first period and some player is indifferent to deviating.

The tuples of payoffs that are generated by equilibria with this structure, i.e., stationarity until constraints bind and extreme binding continuation values, can be decomposed into what we refer to as a *basic pair*, which consists of (i) a tuple of pairs of actions that are played in each state in the first period and (ii) a tuple of *continuation regimes* that describe how play proceeds from period two onwards. The continuation regime for a given state either indicates (iia) that an incentive constraint binds for some player and which extreme binding continuation value is used, or (iib) that incentive constraints are slack, the continuation values are implicitly taken to be the generated tuple of equilibrium payoffs themselves. The basic pair is in a sense a generalization of the familiar decomposition of equilibrium payoffs into a discount-weighted sum of flow utilities and continuation values, except that it also incorporates the exceptional recursive structure that arises in extremal equilibria when incentive constraints are slack. Since there are only finitely many extreme binding continuation values associated with each action, there are only finitely many ways to configure the basic pair. Hence, there is a finite set of *basic equilibrium payoffs* that are sufficient to maximize payoffs in any direction (that is for some set of weights over players' utilities).

The second central feature of the theory we develop is a novel algorithm for computing the tuple of equilibrium payoff sets $\mathbf{V}(s)$. This algorithm adopts a methodology that is quite different from the earlier results of APS and AS. We construct an infinite sequence of payoff tuples that move around the equilibrium payoff correspondence in a clockwise direction. As the algorithm progresses, these payoffs move closer and closer to the true equilibrium payoffs, and asymptotically trace the boundary of $\mathbf{V}$. To be more specific, our algorithm constructs

[5] In exceptional cases, it could be that there is an equivalent action $a_i' \neq a_i$ that could also be played, but even in such cases, a_i may be reused without loss.

a sequence of tuples of payoffs $\mathbf{v}^k$, with one payoff $\mathbf{v}^k(s)$ for each state s, which are estimates of the equilibrium payoffs that all maximize in a given common direction. We refer to the payoff tuples along the sequence as *pivots.* It will always be the case that these pivots are generous estimates, in the sense that they are higher in their respective directions than the highest equilibrium payoff. In addition, we keep track of an estimate of the basic pair that generates each pivot. In a sense, this equilibrium structure is analogous to a hybrid of AS and Blackwell, in that we attempt to "solve out" the stationary features of the equilibrium, but when non-stationarity occurs, we use a coarse approximation of the extreme binding payoffs that can be inductively generated. Our algorithm generates the sequence of pivot payoffs by gradually modifying the approximate basic pair.

This "pivoting" idea is especially fruitful in combination with another insight, that allows us to make changes only in one state at a time as we wrap around and compute successive pivots. This is possible because of a remarkable property of the equilibrium sets: it is possible to "walk" around the boundary of $\mathbf{V}$ while stepping only on payoffs that are generated by basic pairs, where each basic pair differs from its successor in at most one state. We remark that this property is far from obvious, given the complex synergies that can arise between actions in different states through their effect on transitions. For example, switching actions in state s may lead to a higher probability of state s', which is an unfavorable change unless actions are also changed in s' to facilitate transition to a third state s'' that has desirable flow payoffs. It turns out, however, that one does not need to exploit these synergies when starting from maximal payoffs and moving incrementally along the frontier. Moreover, we show that there is a simple procedure for identifying the direction in which a particular modification will cause payoffs to move. Thus, starting from some incumbent basic pair, there is a straightforward series of calculations that identify the modification that moves payoffs in as "shallow" a direction as possible, so that they move along the frontier of the equilibrium payoff correspondence. Indeed, by iteratively computing shallowest directions and making single-state substitutions, one can construct a sequence of basic pairs whose corresponding payoffs demarcate the extent of the equilibrium payoff sets.[6]

This structure is the second pillar of our approach and underlies the algorithm we propose. In particular, our algorithm constructs an analogous sequence of pivots, except that instead of using true basic pairs (which require precise knowledge of equilibrium payoffs), the algorithm constructs approximate basic pairs that use an approximation of the equilibrium payoff correspondence to compute incentive constraints and binding continuation values. At each

[6]Strictly speaking, at each step, our algorithms identify an initial substitution in a new action pair and/or continuation regime in single state. A Bellman-like procedure is used to obtain the next pivot. This updating procedure entails no further changes to actions, but the regimes in some states may change from non-binding to binding, in order to preserve incentive compatibility.

iteration, the algorithm generates "test directions" that indicate how payoffs would move for every possible modification of the action pair or continuation regime in one of the states. The algorithm identifies the shallowest of these test directions and introduces the corresponding modification into the basic pair. We show that this shallowest substitution will cause the pivot to move around the equilibrium payoff correspondence in a clockwise direction but always stay weakly outside. This is the algorithmic analogue of the equilibrium property. As the algorithm progresses, the pivot revolves around and around $\mathbf{V}$. The most recent revolution is used as an estimate of the payoffs that can be used as binding continuation values, so that as the revolutions get closer to $\mathbf{V}$, the estimate improves, and the pivots move even closer. In the limit, the pivot traces the frontier of the true equilibrium payoff correspondence.

At a high level, the progression of these approximations towards the equilibrium payoff correspondence resembles the process by which a prism pencil sharpener gradually shaves material from a new pencil in order achieve a conical shape capped by a graphite tip. In this analogy, the final cone represents the equilibrium payoffs and the initial wooden casing represents the extra non-equilibrium payoffs contained in the initial approximation. Every time the pivot moves, it "shaves" off a slice of the excess material. The rotations continue until the ideal conical shape is attained.

As with the APS algorithm, the set of available continuation values starts out large and is progressively refined, as we require the available continuation values to be inductively generated. A key difference is that APS implicitly generates new payoffs using all kinds of equilibrium structures. Here we show that only certain very particular configurations can generate extreme payoffs. Moreover, those configurations change incrementally as we move around the equilibrium payoff sets. Our algorithm attempts to discover the optimal configuration for maximizing payoffs in each possible *relevant* direction in welfare space, and uses only those configurations to generate new payoffs. By cutting down on the number of equilibria the algorithm tries to generate, by only searching in *endogenously* generated directions (as opposed to some exogenously given set), and by exploiting common structure across states, the algorithm saves time and computational power.

Our procedure is quite different from previous methods for computing equilibrium payoffs, and the complete description requires the introduction of a number of new concepts. We will therefore build slowly towards a general algorithm by first considering a simpler problem. Computing equilibrium behavior in a stochastic game is a complex task, both because behavior is constrained by incentives and because behavior is constrained by the transitions between states. In other words, even if it were incentive compatible, it is generally not feasible to play the same actions in every period, since a given action cannot be

played until its corresponding state is reached. Thus, aside from the question of which payoffs can arise in equilibrium, it is not even obvious which payoffs can arise from any feasible sequence of actions. It turns out, however, that our methodology yields a simple "pencil sharpening" algorithm for calculating this feasible payoff correspondence. The exposition of this algorithm, in Section 3, allows us to develop intuition and ideas that will be used in the computation of equilibrium payoffs.

In addition to our theoretical results, we have also implemented our algorithm as a software package that is freely available through the authors' website.[7] This package consists of a set of routines that compute the equilibrium payoff correspondence, as well as a graphical interface that can be used to specify games and visualize their solutions. The implementation is standalone, and does not require any third party software to use. We have used this program to explore a number of numerical examples, and we will report computations of the equilibria of risk-sharing games à la Kocherlakota (1996).

A classic paper on computation of equilibria in repeated games is Judd, Yeltekin, and Conklin (2003), hereafter JYC. They provide an approximate implementation of the APS algorithm that is based on linear programming. Rather than represent sets of payoffs exactly, they propose to approximate these sets by their supporting hyperplanes in a *fixed* grid of directions. From a given set of bounding hyperplanes, they compute a new set of hyperplanes that bound the set that would be generated by APS, which computation can be reduced to solving a family of linear programming problems. JYC's theoretical results and implementation were written for the non-stochastic case, but the idea extends readily to stochastic games. For comparison, we have written our own implementation of a stochastic version of the JYC algorithm using the commercial linear programming software Gurobi.[8] Preliminary simulations indicate that our algorithm is substantially faster than generalized JYC.

The rest of this paper is organized as follows. Section 2 describes the basic model and background material on subgame perfect equilibria of stochastic games. Section 3 provides a simple algorithm for calculating the feasible payoff correspondence, to build intuition for the subsequent equilibrium analysis. Section 4 gives our characterization of the equilibria that generate extreme payoffs and explains how one might trace the frontier of the equilibrium payoff correspondence. These insights are used in Section 5 to construct the algorithm for computing equilibrium payoffs. Section 6 presents the risk sharing example, and Section 7 concludes. All omitted proofs are in the Appendix.

[7] www.benjaminbrooks.net/software.shtml

[8] Gurobi is available under a free license for academic use.

2 Setting and background

We study stochastic games in which two players $i = 1, 2$ interact over infinitely many periods. The nature of the interaction is governed by a state variable s which lies in a finite set S and evolves stochastically over time. In each period, player i takes an action a_i in a finite set of feasible actions $\mathbf{A}_i(s)$ that are available when the state is s. We denote by $\mathbf{A}(s) = \mathbf{A}_1(s) \times \mathbf{A}_2(s)$ the set of all action pairs that are feasible in state s. Players receive flow utilities as functions of the current state and action pair denoted $g_i(a|s)$. In addition, the next period's state s' is draw from the probability distribution $\pi(s'|a, s)$. Players discount future payoffs at the common rate $\delta \in (0, 1)$. The players' actions and the state of the world are all perfectly observable.

Throughout the following exposition, we will take the pair of actions a to be a sufficient statistic for the state, and simply write $g_i(a)$ and $\pi(s'|a)$. In addition, we will use bold-face to denote functions whose domain is the set of states. Correspondences that map states into sets are denoted by bold upper-case, e.g., $\mathbf{A}$ or $\mathbf{X}$, and functions that map states into actions, scalars, or vectors will generally be denoted by bold lower case, e.g., $\mathbf{a}$ or $\mathbf{x}$. We will abuse notation slightly by writing $\mathbf{x} \in \mathbf{X}$ when $\mathbf{x}(s) \in \mathbf{X}(s)$ for all s.

We will study the equilibrium payoff correspondence $\mathbf{V}$, which associates to each state of the world a compact and convex set of equilibrium payoffs $\mathbf{V}(s) \subset \mathbb{R}^2$ that can be achieved in some pure strategy subgame perfect equilibrium with public randomization, when the initial state of the world is s. For a formal definition of an equilibrium in this setting, see Mailath and Samuelson (2006).[9] The techniques of APS can be used to show that $\mathbf{V}$ is the largest bounded self-generating correspondence (cf. Atkeson, 1991; Phelan and Stacchetti, 2001; Mailath and Samuelson, 2006; Hörner et al., 2011). This recursive characterization says that any subgame perfect equilibrium payoff can be decomposed into the sum of (i) a flow payoff which is obtained in the first period and (ii) expected discounted continuation equilibrium payoffs from period two onwards. Specifically, let $v \in \mathbf{V}(s)$ be generated by a pure strategy a in the first period, and let $\mathbf{w}(s')$ denote the payoff generated by the continuation equilibrium if the state is s' in the second period. Since these continuation values are equilibrium payoffs, we must have $\mathbf{w} \in \mathbf{V}$. Moreover, v and $\mathbf{w}$ must satisfy the *promise keeping* relationship:

$$v = (1 - \delta)g(a) + \delta \sum_{s' \in S} \pi(s'|a)\mathbf{w}(s'). \tag{PK}$$

[9] Strictly speaking, the definition of an equilibrium in Mailath and Samuelson (2006) differs slightly from the one which we are implicitly using. They assume that there is a probability distribution over the state in the initial period, while we are implicitly assuming that an equilibrium is defined *conditional* on a given initial state.

In addition, since v is an equilibrium payoff, neither player must have an incentive to deviate in the first period (incentive constraints after the first period are implicitly satisfied, since $\mathbf{w}(s')$ is an equilibrium payoff for all s'). Since actions are perfectly observable, the continuation payoffs after deviations do not affect the equilibrium payoff, and we may assume without loss of generality that a deviator receives their lowest possible equilibrium payoff in the continuation game. This *equilibrium threat point* is defined by

$$\underline{\mathbf{v}}_i(s) = \min\{w_i | (w_1, w_2) \in \mathbf{V}(s) \text{ for some } w_j\}.$$

That is, $\underline{\mathbf{v}}(s)$ is a vector of the worst equilibrium payoffs for each player in state s. The incentive constraint is therefore that

$$v_i \geq (1-\delta) g(a_i', a_{-i}) + \delta \sum_{s' \in S} \pi(s' | a_i', a_j) \underline{\mathbf{v}}_i(s')$$

for all $a_i' \in \mathbf{A}_i(s)$. Rearranging terms, we can write this condition as

$$\sum_{s' \in S} \pi(s'|a) \mathbf{w}_i(s') \geq h_i(a) \tag{IC}$$

where

$$h_i(a) = \max_{a_i'} \left[\frac{1-\delta}{\delta} (g_i(a_i', a_j) - g_i(a)) + \sum_{s' \in S} \pi(s' | a_i', a_j) \underline{\mathbf{v}}_i(s') \right].$$

Thus, the function $h(a)$ gives the vector of minimum incentive compatible continuation values that are sufficient to deter deviations from the action pair a.

Since $\mathbf{V}$ is the correspondence of all equilibrium payoffs, every payoff $v \in \text{ext}\mathbf{V}(s)$ for each s must be generated in this manner, using some action pair a in the first period and continuation values drawn from $\mathbf{V}$ itself. The technique of APS is to generalize this recursive relationship in a manner that is analogous to how the Bellman operator generalizes the recursive characterization of the value function in dynamic programming. Explicitly, fix a compact-valued payoff correspondence $\mathbf{W}$. Note that the assumption of compactness of $\mathbf{W}$ is maintained throughout. The associated *threat tuple* is $\underline{\mathbf{w}}(\mathbf{W})(s) \in \mathbb{R}^2$, where

$$\underline{\mathbf{w}}_i(\mathbf{W})(s) = \min\{w_i | (w_1, w_2) \in \mathbf{W}(s) \text{ for some } w_j,\ j \neq i\}.$$

For a given action pair $a \in \mathbf{A}(s)$, let

$$h_i(a, \mathbf{W}) = \max_{a_i'} \left[\frac{1-\delta}{\delta} \left(g_i\left(a_i', a_j\right) - g_i(a) \right) + \sum_{s' \in S} \pi\left(s' | a_i', a_j\right) \underline{\mathbf{w}}_i(\mathbf{W})(s') \right].$$

We say that a point v is *generated in state s by the correspondence* $\mathbf{W}$ if there exist $a \in \mathbf{A}(s)$ and $\mathbf{w} \in \mathbf{W}$ such that

$$v = (1-\delta) g(a) + \delta \sum_{s' \in S} \pi(s'|a) \mathbf{w}(s'); \tag{PK$'$}$$

$$\sum_{s' \in S} \pi(s'|a) \mathbf{w}_i(s') \geq h_i(a, \mathbf{W}) \ \forall i = 1, 2. \tag{IC$'$}$$

The correspondence $\mathbf{W}$ is *self-generating* if every $v \in \mathbf{W}(s)$ is a convex combination of payoffs that can be generated in state s by $\mathbf{W}$. In particular, define the operator B by

$$B(\mathbf{W})(s) = \operatorname{co} \{ v | v \text{ is generated in state } s \text{ by } \mathbf{W} \},$$

where co denotes the convex hull. $\mathbf{W}$ is then self-generating if $\mathbf{W} \subseteq B(\mathbf{W})$ (i.e., $\mathbf{W}(s) \subseteq B(\mathbf{W})(s)$ for all $s \in S$). Note from the definition that this operator is monotonic. Tarski's theorem therefore implies that B has a largest fixed point $\mathbf{V}$, which is in fact the equilibrium payoff correspondence.

Throughout the following, we will assume that a pure-strategy subgame perfect equilibrium exists for each possible initial state, so that the sets $\mathbf{V}(s)$ are all non-empty. In fact, we will maintain the even stronger hypothesis that each $\mathbf{V}(s)$ has a non-empty interior. The possibility that some of the $\mathbf{V}(s)$ have less than full dimension adds a profusion of cases for the statement of our algorithm and the proof that the algorithm converges. We therefore regard the full dimension assumption as a cheap way to reduce the complexity of our exposition. We wish to stress, however, that (i) our characterizations of extremal equilibria do not rely on existence of an equilibrium or the full dimensionality of $\mathbf{V}$ and (ii) all of our algorithms have suitable generalizations to the case where $\mathbf{V}$ does not have full dimension.

In the context of repeated (i.e., non-stochastic) games, APS also propose an iterative procedure for calculating $\mathbf{V}$. This procedure extends naturally to the case of stochastic games as follows. Start with any correspondence $\mathbf{W}^0$ that contains $\mathbf{V}$, and generate the infinite sequence $\mathbf{W}^k = B\left(\mathbf{W}^{k-1}\right)$ for $k \geq 1$. One can show that this sequence converges to $\mathbf{V}$ in the sense that $\cap_{k \geq 0} \mathbf{W}^k = \mathbf{V}$. Moreover, if $\mathbf{W}^0$ is chosen so that $B(\mathbf{W_0}) \subseteq \mathbf{W}_0$, then the correspondences will be monotonically decreasing: $\mathbf{W}^k \subseteq \mathbf{W}^{k-1}$ for all $k > 0$.

The operator B uses the worst equilibrium payoffs in $\mathbf{W}$ as threat points for calculating incentive compatibility. In the following, we will make use of a related operator that uses a fixed payoff tuple $\underline{\mathbf{w}}$ as the threat point, which need not coincide with the worst payoffs in $\mathbf{W}$. In particular, for a fixed $\underline{\mathbf{w}}$, we can define

$$h_i(a, \underline{\mathbf{w}}) = \max_{a_i'} \left[(1-\delta) g_i\left(a_i', a_j\right) + \delta \sum_{s' \in S} \pi\left(s' | a_i', a_j\right) \underline{\mathbf{w}}_i\left(s'\right) \right].$$

We say that the payoff v is *generated in state s by $\mathbf{W}$ with threats $\underline{\mathbf{w}}$* if there exist $a \in \mathbf{A}(s)$ and $\mathbf{w} \in \mathbf{W}$ such that (PK′) and

$$v_i \geq h_i(a, \underline{\mathbf{w}}) \ \forall i = 1, 2. \tag{IC″}$$

are satisfied. Then

$$B(\mathbf{W}, \underline{\mathbf{w}}) = \operatorname{co}\left\{v | v \text{ is generated in state } s \text{ by } \mathbf{W} \text{ with threats } \underline{\mathbf{w}}\right\}.$$

We denote by $\mathbf{V}(\underline{\mathbf{w}})$ the largest bounded fixed point of $B(\mathbf{W}, \underline{\mathbf{w}})$. This represents a kind of partial equilibrium correspondence, where continuation values on the equilibrium path must be drawn from $\mathbf{V}(\underline{\mathbf{w}})$, but in the event of a deviation, the continuation values will be $\underline{\mathbf{w}}$. Since $B(\mathbf{W}, \underline{\mathbf{w}})$ is monotonic in $\mathbf{W}$, $\mathbf{V}(\underline{\mathbf{w}})$ is well defined. Moreover, $\mathbf{V}(\underline{\mathbf{w}})$ is monotonic in $\underline{\mathbf{w}}$, in the sense that if $\underline{\mathbf{w}}_i(s) \geq \underline{\mathbf{w}}_i'(s)$ for all i and s, then $\mathbf{V}(\underline{\mathbf{w}}') \subseteq \mathbf{V}(\underline{\mathbf{w}})$. Thus, if $\underline{\mathbf{w}} \leq \underline{\mathbf{v}}$, then $\mathbf{V} \subseteq \mathbf{V}(\underline{\mathbf{w}})$.

3 Intuition: The feasible payoff correspondence

Before describing our approach to computing $\mathbf{V}$, we will first provide some intuition for our methods by solving a simpler problem: finding the *feasible* payoff correspondence $\mathbf{F}$. $\mathbf{F}(s)$ is defined to be the set of discounted present values generated by all possible distributions over action sequences starting from state s, without regard to incentive constraints, but respecting the transition probabilities between states induced by the actions that are played. In a repeated game, the set of feasible payoffs is just the convex hull of stage-game payoffs, since each action can be played every period. In stochastic games, however, the state variable is changing over time, and a given action cannot be played until its corresponding state is reached. Moreover, the distribution over the sequence of states is determined by the sequence of actions. This simultaneity makes calculating $\mathbf{F}$ a non-trivial task.

State 1

	C	D
C	$1,1\ ^{1/3}$	$-1,2\ ^{1/2}$
D	$2,-1\ ^{1/2}$	$0,0\ ^{1/3}$

State 2

	C	D
C	$3,3\ ^{1/3}$	$1,4\ ^{1/2}$
D	$4,1\ ^{1/2}$	$2,2\ ^{1/3}$

Table 1: A simple stochastic game

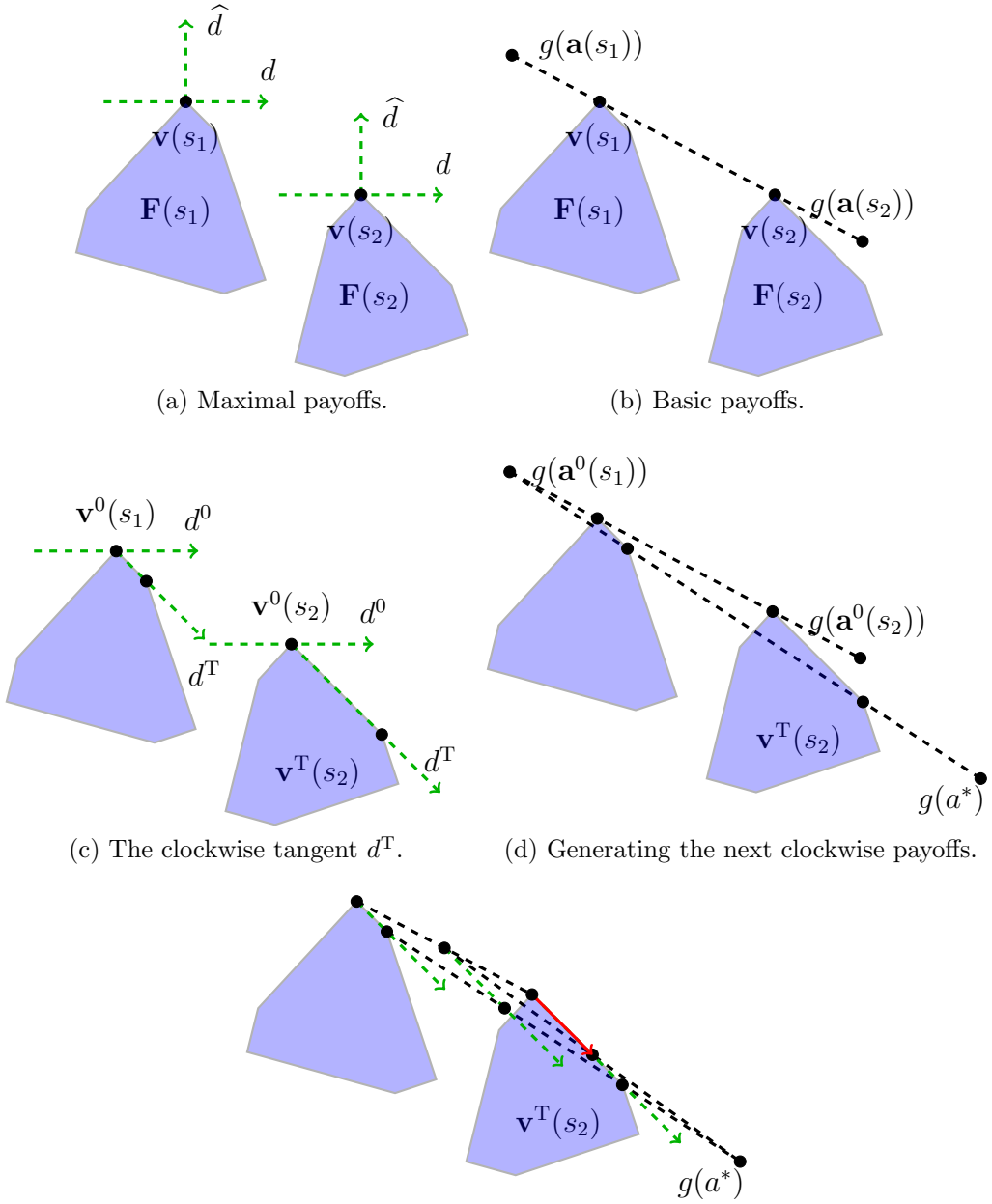
For example, consider the simple stochastic game depicted in Table 1. There are two states in which the stage game takes the form of a prisoner's dilemma. For each state, we have written the players' payoffs, followed by the probability of remaining in the same state after playing that action profile. Note that the payoffs in state 2 are equal to the payoffs in state 1, shifted up by the vector $(2, 2)$. In addition, the level of persistence is the same for corresponding action pairs. While it is easy to represent the stage-game payoffs, transition probabilities complicate the choice of best actions overall. For example, if the goal were to maximize the sum of players payoffs, should (C, C) be played in state 2, even though it leads to a lower probability of remaining in the more favorable state 2 than do (C, D) and (D, C)?

We approach the problem as follows. A payoff (vector) $v \in \mathbb{R}^2$ that is an extreme point of $\mathbf{F}(s)$ must maximize some linear objective over all elements of $\mathbf{F}(s)$. In particular, there must exist some vector $d = (d_1, d_2)$ such that the line $\{v + xd \mid x \in \mathbb{R}\}$ is a supporting hyperplane of $\mathbf{F}(s)$ at v. Let us further denote by $\widehat{d} = (-d_2, d_1)$ the counter-clockwise normal to that d, i.e., d rotated 90 degrees counter-clockwise. Supposing that d points clockwise around $\mathbf{F}$, then it must be the case that v is a solution to

$$\max_{w \in \mathbf{F}(s)} w \cdot \widehat{d}. \tag{1}$$

We denote this relationship by saying that the payoff v is *d-maximal*. We may extend this notion to a *tuple* of payoffs $\mathbf{v}$, where $\mathbf{v}(s) \in \mathbf{F}(s)$ for all $s \in S$. If $\mathbf{v}(s)$ is d-maximal for each s, then we will say that the entire tuple of payoffs $\mathbf{v}$ is d-maximal. Finally, payoffs and payoff tuples are maximal if they are d-maximal for some direction d. These definitions are illustrated in Figure 1a,[10] which depicts a game with two states $S = \{s_1, s_2\}$, with payoffs in state s_1 on the left and payoffs in state s_2 on the right. Highlighted in green are the directions d and its counter-clockwise normal $\widehat{d}$, for which $\mathbf{v}$ is the maximal tuple.

[10]This picture has been drawn for the case in which $\pi(s'|a) > 0$ for all s' and for all actions which generate extreme payoffs. It is for this reason that all of the edges of $\mathbf{F}(s_1)$ are parallel to edges of $\mathbf{F}(s_2)$. More generally, if transition probabilities are degenerate, there does not have to be any particular relationship between the shapes of the feasible payoff sets for states which are not mutually reachable.



(a) Maximal payoffs.

(b) Basic payoffs.

(c) The clockwise tangent d^{T}.

(d) Generating the next clockwise payoffs.

(e) A test direction.

Figure 1: The structure of feasible payoffs.

For a fixed vector d, the problem of maximizing $\mathbf{v} \cdot \widehat{d}$ over all feasible distributions over action sequences has the structure of a Markov decision problem, and it is a well-known result in dynamic programming that there exists an optimal solution which is stationary (Blackwell, 1965). In particular, for any direction d there is a d-maximal payoff tuple which is generated by stationary strategies in which the same actions $\mathbf{a}(s)$ are played whenever the

state is s. The payoff tuple so generated is the unique solution of the system of equations

$$\mathbf{v}(s) = (1-\delta)g\left(\mathbf{a}(s)\right) + \delta \sum_{s' \in S} \pi\left(s'|\mathbf{a}(s)\right) \mathbf{v}(s') \ \forall s \in S. \tag{2}$$

We refer to payoff tuples that can be generated as the solution to (2) for some choice of action tuple as *basic*. Indeed, every extreme point of $\mathbf{F}(s)$ is part of some basic payoff tuple, and hence can be described as a solution of (2) for some choice of action tuple $\mathbf{a}$. We can see in Figure 1b how basic payoff tuples can be generated by a tuple of stationary strategies defined by an action tuple $\mathbf{a}$.

One could in principle use this observation to characterize the feasible payoff correspondence by solving (2) for *every* possible action tuple to obtain all of the basic payoff tuples and taking the convex hull. An immediate implication is that each $\mathbf{F}(s)$ has a finite number of extreme points. There is, however, a large number of action tuples, and this number grows exponentially in the number of states. One might hope to find an algorithm for characterizing $\mathbf{F}(s)$ where the computational burden depends not on the number of action tuples but rather on the number of actual extreme points, which may be much smaller in practice.

We will argue that such a procedure exists. The naïve algorithm described in the previous paragraph uses the recursive structure of d-maximal payoffs, but there is even more structure to be exploited in how the maximal action tuples change with the direction of maximization. For example, suppose that one had found a maximal action tuple $\mathbf{a}$ with corresponding basic payoff tuple $\mathbf{v}$. A conjecture is that that the action tuples that generate maximal payoff tuples for nearby directions will be similar in structure to $\mathbf{a}$. If this conjecture were correct, we could use the known maximal structure to find other maximal structures by making small modifications to $\mathbf{a}$, and thereby extend our knowledge of the frontier of $\mathbf{F}$.

This intution is indeed correct and is a fundamental building block of the algorithm we propose. Consider a direction d^0, and suppose that we know the actions $\mathbf{a}^0$ which generate a basic payoff tuple $\mathbf{v}^0$ which is d^0-maximal. Let us further suppose that $\mathbf{F}(s)$ is not a singleton in every state. Thus, if the direction of maximization were to rotate clockwise from d^0, $\mathbf{v}^0$ would eventually cease to be maximal in the rotated direction, and there is some critical d^{T} such that if the direction were to rotate any further clockwise, some other extremal tuple would become maximal. The direction d^{T} is in fact the clockwise tangent from $\mathbf{v}^0$ to $\mathbf{V}$. Indeed, if the direction rotated any further, a particular tuple $\mathbf{v}^{\mathrm{T}}$ would become uniquely maximal. These payoffs are *clockwise d^T-maximal*, meaning that of all of the tuples that are d^{T}-maximal, they are the ones that are furthest in the direction d^{T}. Note that at the critical direction d^{T}, both $\mathbf{v}^0$ and $\mathbf{v}^{\mathrm{T}}$ are maximal. Thus, it must be possible to find non-negative

scalars $\mathbf{x}(s)$ that are not all zero such that

$$\mathbf{v}^{\mathrm{T}}(s) = \mathbf{v}^{0}(s) + \mathbf{x}(s)d^{\mathrm{T}}.$$

Figure 1c illustrates the directions d^0 and d^{T} in our running example, and Figure 1d shows how the clockwise basic payoffs are generated.

We argue that it must be possible to modify $\mathbf{a}^0$ by changing the action pair in a single state, so that the resulting basic tuples move towards $\mathbf{v}^{\mathrm{T}}$ from $\mathbf{v}^0$. In particular, let s^* be any state in which $\mathbf{x}(s)$ is maximized. In Figure 1, this occurs in state s_2. Since $\mathbf{v}^{\mathrm{T}}(s^*)$ is uniquely maximal, it must also be basic, and thus it is generated by some pure actions a^* in the first period and continuation values $\mathbf{v}^{\mathrm{T}}(s')$, i.e.

$$\mathbf{v}^{\mathrm{T}}(s^*) = (1-\delta)\, g(a^*) + \delta \sum_{s' \in S} \pi(s'|a^*)\, \mathbf{v}^{\mathrm{T}}(s').$$

Now consider the strategy of playing a^* for one period, followed by a return to the original stationary strategies associated with $\mathbf{a}^0$ forever after. The payoff thus generated must be

$$v = (1-\delta) g(a^*) + \delta \sum_{s' \in S} \pi(s'|a^*)\, \mathbf{v}^{0}(s').$$

As a result, the direction from $\mathbf{v}^0(s)$ to v must be

$$\begin{aligned} v - \mathbf{v}^0(s^*) &= \mathbf{v}^{\mathrm{T}}(s^*) - \mathbf{v}^0(s^*) - \delta \sum_{s' \in S} \pi(s'|a^*) \left(\mathbf{v}^{\mathrm{T}}(s') - \mathbf{v}^0(s')\right) \\ &= \left(\mathbf{x}(s^*) - \delta \sum_{s' \in S} \pi(s'|a^*)\, \mathbf{x}(s')\right) d^{\mathrm{T}}, \end{aligned}$$

and since $\mathbf{x}(s^*) \geq \mathbf{x}(s')$ for all s' and $\delta < 1$, it must be that the coefficient on d^{T} is strictly positive. As a result, v must lie in the direction d^{T} relative to $\mathbf{v}^0(s^*)$.

In the simple example in Figure 1d, s_2 is the state in which $\mathbf{x}(s)$ is maximized. Figure 1e shows that playing a^* for one period, followed by returning to the stationary strategies defined by $\mathbf{a}^0$, generates a payoff that lies between $\mathbf{v}^0(s_2)$ and $\mathbf{v}^{\mathrm{T}}(s_2)$.

Now, let us consider what would happen if we were to modify $\mathbf{a}^0$ by substituting a^* in place of $\mathbf{a}^0(s^*)$, to obtain a new action tuple $\mathbf{a}^1$, which in turn generate a new basic payoff tuple $\mathbf{v}^1$. We claim that this new payoff tuple $\mathbf{v}^1$ has to lie between $\mathbf{v}^0$ and $\mathbf{v}^{\mathrm{T}}$. To see this,

first observe that $\mathbf{v}^1$ must be the fixed point of the following Bellman operator $\mu(\cdot)$:

$$\mu(\mathbf{w})(s) = (1-\delta)g\left(\mathbf{a}^1(s)\right) + \delta \sum_{s' \in S} \pi\left(s'|\mathbf{a}^1(s)\right)\mathbf{w}(s).$$

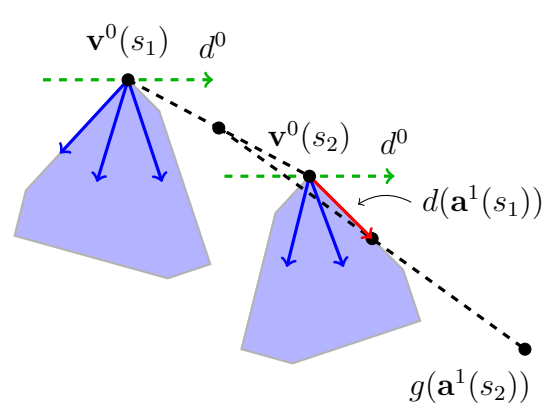
This operator is a contraction mapping, and so starting from any initial guess of $\mathbf{v}^1$, the iterates will eventually converge to the unique fixed point. Moreover, we have already proven that starting from $\mathbf{v}^0$, $\mu(\mathbf{v}^0)$ moves in the direction d^{T} relative to $\mathbf{v}^0$ (in state s^* only). The linearity of μ in $\mathbf{w}$ implies that subsequent applications of μ will only move payoffs further in the direction d^{T}, as the movement in state s^* is propagated through to the other states. Thus, we conclude that there must be at least one action which, when substituted into the initial stationary action sequence, must result in a clockwise movement of the basic payoffs around the feasible correspondence. We should note that generically $\mathbf{v}^1$ is in fact equal to $\mathbf{v}^{\mathrm{T}}$, and only a single substitution will be required to cross each edge of $\mathbf{F}$. In the case of equilibrium studied in Section 4, however, it is a generic possibility that multiple substitutions may be required to move across an edge of $\mathbf{V}$.

Thus far, we have presumed that we omnisciently knew the action a^* that generated the payoffs that were clockwise relative to $\mathbf{v}^0$. This was unnecessary: First, the argument of the previous paragraph shows that we can identify the direction in which a substitution moves the payoff tuple by just looking at the first application of μ in the substituted state. In other words, if a new action tuple differs from $\mathbf{a}^0$ only in state s in which the action is $a \neq \mathbf{a}^0(s)$, then the direction that the substitution moves payoffs will be

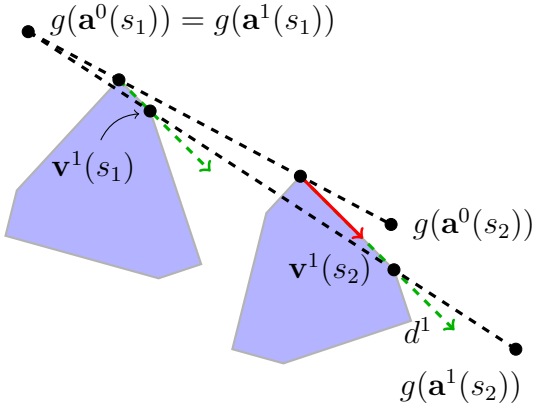
$$d(a) = (1-\delta)g(a) + \delta \sum_{s' \in S} \pi(s'|a)\mathbf{v}^0(s') - \mathbf{v}^0(s).$$

As such, we can easily project where a given substitution will send the payoffs by computing the *test direction* $d\,(a)$. Second, we know that there is *some* substitution that will move us in the direction that points along the frontier. We could therefore consider all substitutions in all states and compare the corresponding test directions. Note that it is impossible for any of the test directions to point above the frontier, since this would imply the existence of a feasible payoff tuple that is outside of $\mathbf{F}$. As a result, the test direction with the smallest clockwise angle of rotation relative to d^0 must point along the frontier, and by implementing the substitution associated with this direction, payoffs are guaranteed to move clockwise along the boundary of $\mathbf{F}$.

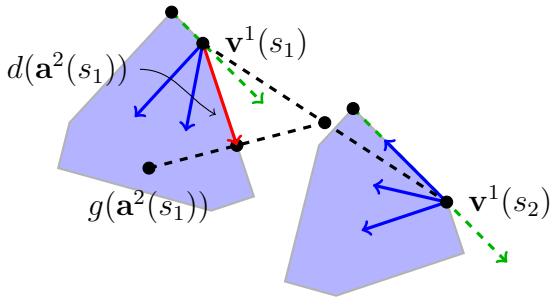
From these observations, it follows that there is a simple way, starting from a known $\mathbf{v}^0$, $\mathbf{a}^0$, and d^0, to trace the entire frontier of $\mathbf{F}$ using only basic payoff tuples. We first generate all of the possible test directions $d(a)$ for all possible substitutions. One of these



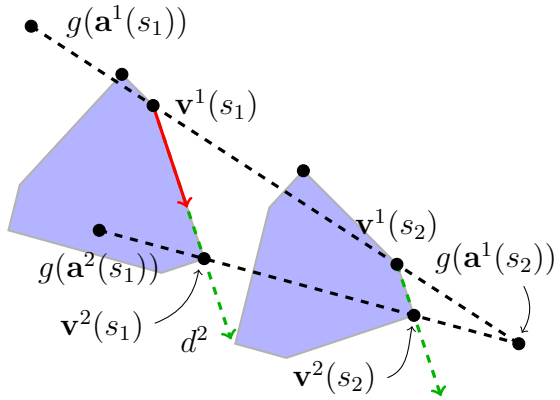
(a) Generate test directions.



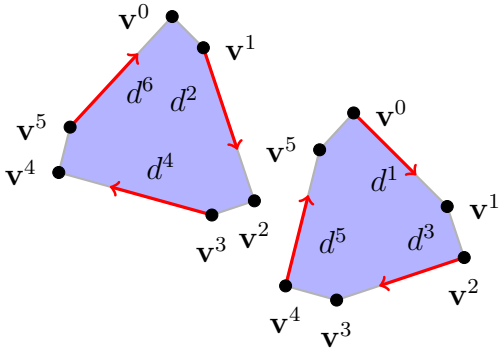
(b) Substitute and advance the pivot.



(c) Generate more test directions.



(d) Substitute again.



(e) Iterate.

Figure 2: Tracing the frontier.

directions, denoted by d^1, is *shallowest*, in the sense of having the smallest clockwise angle of rotation from d^0. This shallowest direction must in fact coincide with the tangent direction d^{T}. We then form a new action tuple $\mathbf{a}^1$ by substituting in the action a^* that generated the shallowest test direction and leaving the rest of the actions unchanged, and the new

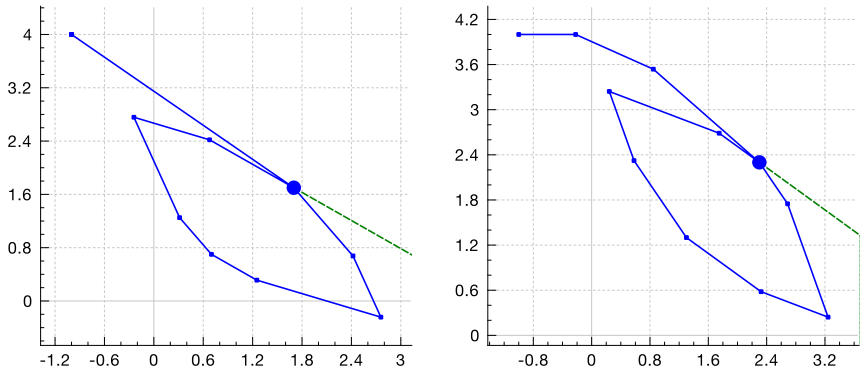


Figure 3: Tracing the boundary of the feasible set for the prisoners' dilemmas example.

tuple $\mathbf{a}^1$ in turn generates new basic payoffs $\mathbf{v}^1$. We then repeat this process, inductively finding shallowest directions and substituting to generate a sequence of action tuples $\mathbf{a}^k$. These action tuples generate a sequence of payoff tuples $\mathbf{v}^k$ that we refer to as *pivots*, since the direction of movement pivots around these points while tracing the frontier. For generic payoffs, the pivot moves in a new direction each time we make a substitution, though it could in principle move more than once in the same direction. These pivots progress clockwise around the frontier of $\mathbf{F}$, and eventually, the pivot will visit all of the basic payoff tuples which are uniquely maximal in some direction, at which point we will have traced the frontier of $\mathbf{F}$.

Figure 2 demonstrates this "pencil sharpening" algorithm for feasible payoffs. Figure 2a shows how, starting from the initial strategies $\mathbf{a}^0$ and basic payoffs $\mathbf{v}^0$, we send out test directions that correspond to playing different actions for one period, followed by a return to the stationary strategies $\mathbf{a}^0$. Most of these test directions are depicted in blue, but one direction in state s_2, painted in red, points clockwise along the frontier of $\mathbf{F}$. In Figure 2b, we substitute this action into the stationary strategy to obtain new basic payoff tuples $\mathbf{v}^1$. Figure 2c shows how we again send out test directions, this time identifying a shallowest direction in state s_1, which is substituted in Figure 2d. This process iterates, and Figure 2e shows how we traverse the entire frontier, eventually returning to $\mathbf{v}^0$ after six steps.

The one remaining difficulty with our proposed algorithm is that it presumes an initial $\mathbf{a}^0$ that generates basic payoffs $\mathbf{v}^0$ that are d^0-maximal. How do we find such a starting point? It turns out that we do not have to! We can augment the game in a simple way that automatically gives us an initial condition. Pick any tuple $\mathbf{v}^0$ that can be weakly separated from $\mathbf{F}$ by some hyperplane. For example, $\mathbf{v}^0(s)$ could be the pointwise maximum of each player's payoffs across all actions in state s, for each s. We use these payoffs to create a tuple of "synthetic" action pairs $\mathbf{a}^0(s) \notin \mathbf{A}(s)$, and define $g\left(\mathbf{a}^0(s)\right) = \mathbf{v}^0(s)$ and $\pi\left(s|\mathbf{a}^0(s)\right) = 1$, so

that $\mathbf{a}^0$ generates the initial pivot. Starting from this initial condition, we trace the boundary of the feasible payoffs of the augmented game which consists of all of the original actions together with the synthetic actions $\mathbf{a}^0$. It is not hard to see that once we pivot around to the opposite side of $\mathbf{F}$ from $\mathbf{v}^0$, it will not be optimal to use any of the the synthetic actions $\mathbf{a}^0$. In fact, these actions will be dominated by *any* choice of actions in the original game. Thus, the synthetic actions will eventually be driven out of the solution, at which point we must be generating payoffs on the boundary of $\mathbf{F}$ for the original game. From that point on, we disallow the synthetic actions $\mathbf{a}^0$ from being substituted back in, and all further pivoting operations will remain on the frontier of $\mathbf{F}$. After one more full revolution, we will have traced the entire boundary of feasible payoffs for the original game.

We used this procedure to calculate $\mathbf{F}$ for the game in Table 1. The results of the computation are depicted in Figure 3. The initial pivot consisted of payoffs of $(-1, 4)$ in both states, which corresponds to the lowest payoff for player 1 and the maximum payoff for player 2, across all states and action pairs. We then searched for the shallowest direction from the initial pivot, which was generated by using (C, D) in state 2 and staying with the synthetic $\mathbf{a}^0$ in state 1. The next substitution is (C, C) in state 2, and then (C, C) in state 1. At this point, both synthetic actions have been driven out of the system, and we are generating payoffs on the frontier, in particular the symmetric surplus maximizing tuple of payoffs.

Note that from the symmetric point, it is somewhat ambiguous what action should be introduced to generate the next edge. Pivoting from (C, C) to (D, C) in either state shifts the stage payoffs in the direction $(1, -2)$. However, introducing (D, C) in state 1 entails a higher probability of being in state 2 tomorrow (2/3 versus 1/2), which results in efficiency gains that partially offset the loss to player 2. In contrast, introducing (D, C) in state 2 entails a higher probability of state 1, which causes further efficiency losses. Thus, the correct choice must be to pivot to (D, C) in state 1, and this is indeed the substitution that is identified by our algorithm.

4 The structure of equilibrium payoffs

We now return to the primary objective of this paper, which is the characterization and computation of equilibrium payoffs. The equilibrium requirement adds significant complexity relative to the analysis of the feasible payoffs in the previous section. Equilibrium payoffs are generated by distributions over action sequences that not only obey the transition probabilities between states but also satisfy the players' forward-looking incentive constraints. In spite of this additional complexity, we shall see that there are high-level similarities between

the structure of feasible payoffs and that of equilibrium payoffs. Characterizing the equilibria that generate extremal equilibrium payoffs is the subject of this section, and in Section 5, we will use these results to develop a general algorithm for computing $\mathbf{V}$.

4.1 Basic pairs

The central tool for characterizing equilibrium payoffs will be what we refer to as a *basic pair*, which in a sense generalizes the stationary strategies of Section 3. This object consists of a tuple of action pairs $\mathbf{a}$ and a tuple of *continuation regimes* $\mathbf{r}$, which we shall explain presently. In Section 2, we reviewed the standard analytical technique of decomposing an equilibrium payoff as the discount-weighted average of a flow payoff and an expected equilibrium continuation value. The basic pair gives this decomposition for an entire *tuple* of equilibrium payoffs $\mathbf{v} \in \mathbf{V}$ simultaneously. In particular, each $\mathbf{v}(s)$ can be decomposed as a weighted average of a flow payoff, $g(\mathbf{a}(s))$, and an expected continuation value w which is determined by $\mathbf{r}(s)$.

The continuation regime, and consequently the expected continuation value, falls into one of two categories depending on whether or not the incentive constraints (IC) hold strictly or with equality. In the *non-binding case*, $\mathbf{r}(s) = \text{NB}$, and the continuation value is simply the expectation of the payoff tuple $\mathbf{v}$ itself. In the *binding case*, at least one of the players is indifferent to deviating from their prescribed action $\mathbf{a}_i(s)$, so that the expected continuation value lies along a binding incentive constraint. Moreover, this continuation value must be an extreme point of the set of feasible and incentive compatible expected continuation values. In the binding case, the continuation regime $\mathbf{r}(s)$ is the expected continuation value itself.

To be more precise, let

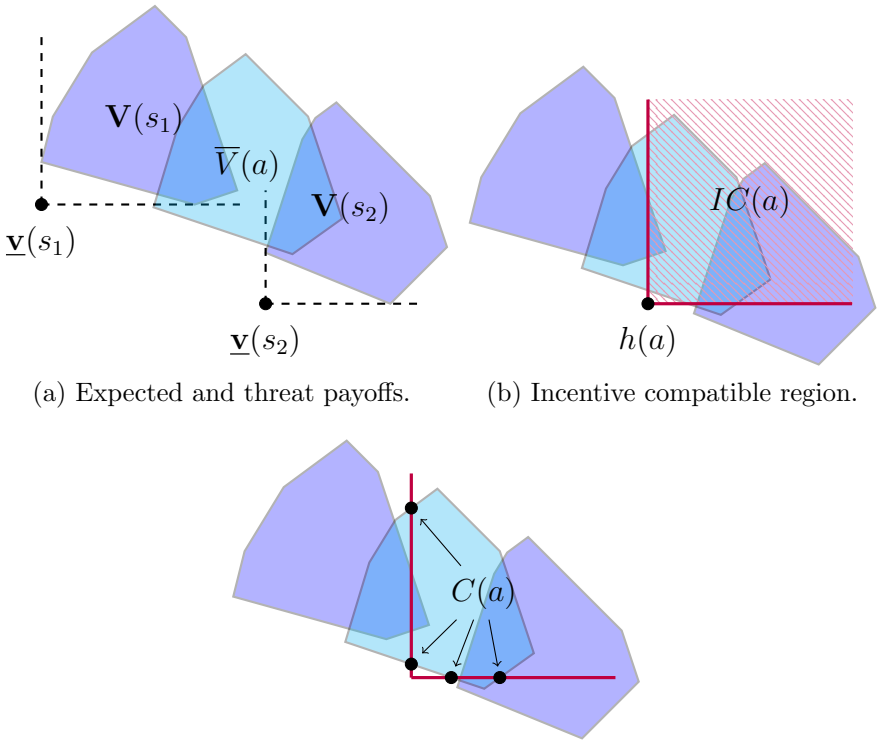
$$\overline{V}(a) = \sum_{s' \in S} \pi(s'|a)\mathbf{V}(s')$$

denote the set of expected equilibrium continuation values when the action pair a is played, and let

$$IC(a) = \left\{w \in \mathbb{R}^2 | w \geq h(a)\right\}$$

denote the set of continuation value pairs that would deter players from deviating from the action pair a (where the minimal expected continuation value $h(a)$ is defined in Section 2). The set of extreme binding continuation values is

$$C(a) = \text{ext}\left(\overline{V}(a) \cap \text{bd}IC(a)\right),$$



(a) Expected and threat payoffs.

(b) Incentive compatible region.

(c) Extreme binding continuations.

Figure 4: The geometry of feasible and incentive compatible continuation values.

where bd denotes the topological boundary, so that bd$IC(a)$ is the set of continuation value vectors at which at least one player is indifferent to deviating.

These sets are depicted for our two-state example in Figure 4. A key observation is that $C(a)$ can have at most four elements. The reason is that bd$IC(a)$ is the union of two rays, so that the intersection of bd$IC(a)$ with $\overline{V}(a)$ is the union of two line segments. Each of these line segments can have at most two extreme points, so that between the two players' constraints there are at most four extreme binding continuation values. Figure 4c illustrates the case where $C(a)$ has the maximal number of elements.

Thus, returning to the definition of the continuation regime, either $\mathbf{r}(s) = \text{NB}$ in the non-binding case, or $\mathbf{r}(s) \in C(\mathbf{a}(s))$ if a constraint binds. As a result, $\mathbf{r}(s)$ can take on at most five values once we have fixed $\mathbf{a}(s)$, so that we can bound the number of basic pairs:

Lemma 1 (Basic pairs)**.** *The number of basic pairs is at most* $5^{|S|}\prod_{s\in S}|\mathbf{A}(s)|$.

We say that the basic pair $(\mathbf{a}, \mathbf{r})$ *generates the payoff tuple* $\mathbf{v}$ if

$$\mathbf{v}(s) = (1-\delta)\, g(\mathbf{a}(s)) + \delta \begin{cases} \sum_{s'\in S} \pi(s'|\mathbf{a}(s))\mathbf{v}(s') \text{ if } \mathbf{r}(s) = \text{NB}; \\ \mathbf{r}(s) \in C(\mathbf{a}(s)) \text{ otherwise}, \end{cases} \tag{3}$$

and if

$$\sum_{s' \in S} \pi\left(s' | \mathbf{a}\left(s\right)\right) \mathbf{v}\left(s'\right) \in IC\left(\mathbf{a}\left(s\right)\right) \tag{4}$$

whenever $\mathbf{r}\left(s\right) = \text{NB}$. In this case, we say that $\mathbf{v}$ is a tuple of *basic equilibrium payoffs*. Equation (3) is a promise keeping condition, analogous to (PK$'$). Incentive constraints are satisfied by definition when $\mathbf{r}\left(s\right) \in C\left(\mathbf{a}\left(s\right)\right)$, and (4) ensures that the expected payoffs themselves are incentive compatible whenever $\mathbf{r}\left(s\right) = \text{NB}$. Note that the tuple of payoffs $\mathbf{v}$ that solves (3) is the unique fixed point of a certain Bellman operator

$$\mu(\mathbf{w}; \mathbf{a}, \mathbf{r})(s) = (1-\delta) g(\mathbf{a}(s)) + \delta \begin{cases} \sum_{s' \in S} \pi(s'|\mathbf{a}(s)) \mathbf{w}(s') \text{ if } \mathbf{r}(s) = \text{NB}; \\ \mathbf{r}(s) \text{ otherwise.} \end{cases} \tag{5}$$

Thus, μ maps a tuple of payoffs $\mathbf{w}$ into a new tuple of payoffs $\mu\left(\mathbf{w}; \mathbf{a}, \mathbf{r}\right)$, where $\mu\left(\mathbf{w}; \mathbf{a}, \mathbf{r}\right)\left(s\right)$ is the discount-weighted average of $g\left(\mathbf{a}\left(s\right)\right)$ and $\mathbf{w}$ when $\mathbf{r}\left(s\right) = \text{NB}$.

Lemma 2 (Basic equilibrium payoffs)**.** *Suppose that the basic pair* $\left(\mathbf{a}, \mathbf{r}\right)$ *generates* $\mathbf{v}$*. Then* $\mathbf{v} \in \mathbf{V}$*.*

Proof of Lemma 2. Consider the correspondence $\mathbf{V}'$ defined by $\mathbf{V}'\left(s\right) = \mathbf{V}\left(s\right) \cup \left\{\mathbf{v}\left(s\right)\right\}$. It is immediate from the definitions that $\mathbf{V}'$ is self-generating, so that $\mathbf{V}' \subseteq \mathbf{V}$, and hence $\mathbf{v} \in \mathbf{V}$. □

Thus, a basic pair describes the decomposition of an entire tuple of equilibrium payoffs into flows and continuations. Such a tuple is generated by a *system of equilibria*, one equilibrium for each possible initial state. Basic pairs correspond to what we might call *basic equilibrium systems*, that exhibit exceptional recursive structure. In particular, each equilibrium in this system can be decomposed into a flow payoff in the first period and a continuation equilibrium system, which describes the continuation equilibria that are played for each possible state in the second period. The system described by the basic pair has the feature that whenever incentive constraints are slack in the first period, the continuation system is simply a reboot of the original equilibrium system, with the prior history erased. This corresponds to the case where $\mathbf{r}\left(s\right) = \text{NB}$. When $\mathbf{r}\left(s\right) \neq \text{NB}$, the continuation equilibrium system generates an extreme binding expected continuation value. This perspective is very analogous to how action tuples were used to describe a corresponding tuple of stationary strategies in Section (3). A stationary strategy tuple defines a strategy for each possible initial state, and after the first period, the strategy tuple simpy restarts. This remains true of the basic equilibrium system when incentive constraints are slack, although when incentive constraints bind, the equilibrium may evolve in a non-stationary manner.

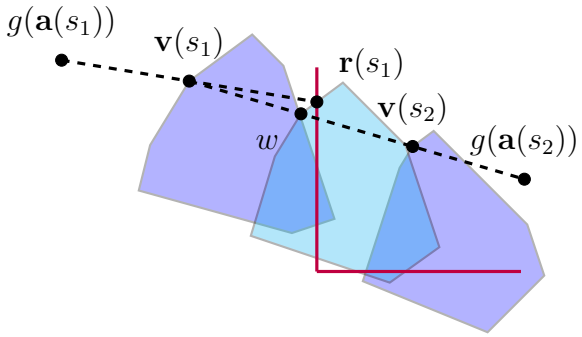


Figure 5: Basic equilibrium payoffs.

Figure 5 gives an example of how basic equilibrium payoffs are generated. In state s_1, on the left, the equilibrium that maximizes player 2' s payoff involves a binding continuation value (at which player 1's incentive constraint is binding), whereas in state s_2, on the right, the equilibrium that maximizes player 2's payoffs has slack constraints. For simplicity, this picture is drawn for the special case where the transition probabilities $\pi(\cdot|\mathbf{a}(s_1))$ and $\pi(\cdot|\mathbf{a}(s_2))$ coincide, so that $\overline{V}(\mathbf{a}(s_1)) = \overline{V}(\mathbf{a}(s_2))$. We suppose, however, that $h(\mathbf{a}(s_1)) > h(\mathbf{a}(s_2))$, so that while the expected pivot w is incentive compatible in state s_2, it is not incentive compatible in s_1.

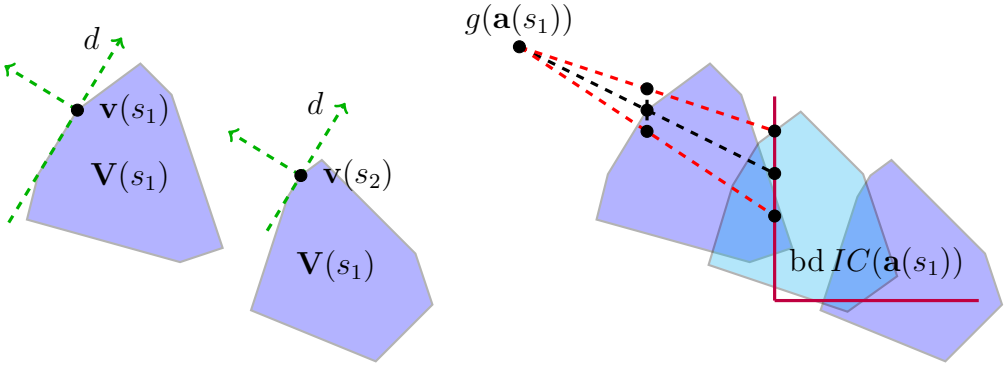
4.2 The maximality of basic equilibrium payoffs

As we have already indicated, basic pairs turn out to have properties which make them extremely useful for the characterization and computation of equilibrium payoffs. First, it turns out that basic pairs are sufficient to maximize equilibrium payoffs in any given direction. This is in a sense a generalization of the sufficiency of stationary strategies for maximizing feasible payoffs in a given direction.

It will be convenient in the sequel to use a refined notion of maximality that selects for a unique maximal payoff tuple. Recall that a payoff tuple $\mathbf{v}$ is clockwise d-maximal in $\mathbf{V}$ if it is d-maximal and if there is no other d-maximal tuple $\mathbf{v}'$ such that $\mathbf{v}'(s) \cdot d > \mathbf{v}(s) \cdot d$ for some s. In other words, among all d-maximal tuples, $\mathbf{v}$ is the one that is furthest in the direction d. Note that while there may be many d-maximal equilibrium payoff tuples, the clockwise d-maximal tuple is unique. Moreover, all of the $\mathbf{v}(s)$ must be extreme points of their respective $\mathbf{V}(s)$. This relationship is depicted in Figure 6a. In both states s_i , there is a continuum of d-maximal payoffs but a unique clockwise d-maximal payoff, which is $\mathbf{v}(s_i)$.

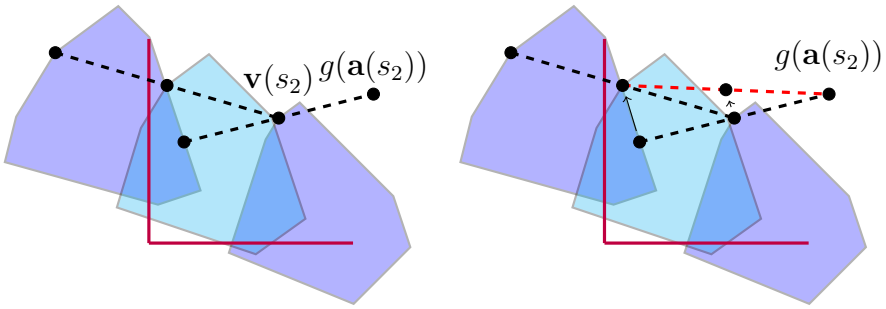
We have the following result:

Proposition 1 (Maximality of basic equilibrium payoffs)**.** *For every direction d, the clockwise d-maximal equilibrium payoffs are basic.*



(a) Clockwise maximal payoffs.

(b) Binding perturbation.



(c) Sub-optimal continuation payoffs.

(d) Non-binding perturbation.

Figure 6: Maximal basic equilibrium payoffs.

Proof of Proposition 1. Suppose $\mathbf{v}$ is clockwise d-maximal. Since $\mathbf{v}(s) \in \text{ext}\mathbf{V}(s)$, $\mathbf{v}(s)$ must be generated by some action $\mathbf{a}(s)$ and expected continuation value $\mathbf{w}(s) \in \overline{V}(\mathbf{a}(s))$. Note that an arbitrary $v \in \mathbf{V}(s)$ may require public randomization over actions in the first period, but this is not true of the extreme points of $\mathbf{V}(s)$.

If (IC) is binding, then it must be that the expected continuation value $\mathbf{w}(s)$ is in $C(a)$. If not, then there must exist a direction $\widetilde{d}$ such that $\mathbf{w}(s)+\widetilde{d}$ and $\mathbf{w}(s)-\widetilde{d}$ are both feasible and incentive compatible continuation values in $\overline{V}(a) \cap \text{bd}IC(a)$, so that we can generate the payoffs $\mathbf{v}(s)+\delta\widetilde{d}$ and $\mathbf{v}(s)-\delta\widetilde{d}$, thus contradicting the extremeness of $\mathbf{v}(s)$. This is depicted in Figure 6b. Thus, $\mathbf{w}(s) \in C(a)$, and we can set $\mathbf{r}(s) = \mathbf{w}(s)$.

On the other hand, suppose that (IC) is slack for both $i = 1, 2$, and

$$\mathbf{w}(s) \neq \sum_{s' \in S} \pi(s'|\mathbf{a}(s))\mathbf{v}(s') = w.$$

This configuration is depicted in Figure 6c. Note that we must be able to find a $\widetilde{\mathbf{v}}$ such that

$$\mathbf{w}(s) = \sum_{s' \in S} \pi(s'|\mathbf{a}(s))\widetilde{\mathbf{v}}(s'),$$

and since $\mathbf{w}(s) \neq w$, there must be at least one state s'' with $\pi(s''|\mathbf{a}(s)) > 0$ such that $\widetilde{\mathbf{v}}(s'') \neq \mathbf{v}(s'')$. Since the clockwise d-maximal payoff is unique, $\mathbf{v}(s'')$ is either higher in the $\widehat{d}$ direction or in the d direction relative to $\widetilde{\mathbf{v}}(s'')$.

Since $\mathbf{V}$ is convex (because of public randomization), the payoff

$$\widetilde{w} = \sum_{s' \neq s''} \pi(s'|\mathbf{a}(s))\widetilde{\mathbf{v}}(s) + \pi(s''|\mathbf{a}(s))((1-\epsilon)\widetilde{\mathbf{v}}(s'') + \epsilon \mathbf{v}(s''))$$

is in $\overline{V}(\mathbf{a}(s))$ for every $\epsilon \in (0,1)$, and since (IC) is slack, there must be a sufficiently small but positive ϵ so that constraints will still be statisfied, i.e., $\widetilde{w} \geq h(a)$ (the constraint is satisfied strictly at $\epsilon = 0$). Thus, it is possible to generate the payoff

$$\begin{aligned}\widetilde{v} &= (1-\delta)g(\mathbf{a}(s)) + \delta\widetilde{w} \\ &= \mathbf{v}(s) + \delta\pi(s''|\mathbf{a}(s))\epsilon(\mathbf{v}(s'') - \widetilde{\mathbf{v}}(s'')).\end{aligned}$$

We can see the construction of this payoff in Figure 6d. Thus, $\widetilde{v}$ must be higher in the $\widehat{d}$ or d direction, thus contradicting clockwise d-maximality of $\mathbf{v}(s)$.

As a result, it must be that $\widetilde{\mathbf{v}}(s') = \mathbf{v}(s')$ for states in which $\pi(s'|\mathbf{a}(s)) > 0$, and it is obviously without loss of generality to take this to be true when $\pi(s'|\mathbf{a}(s)) = 0$. We can then set $\mathbf{r}(s) = \text{NB}$, so that $\mathbf{v}(s)$ is a solution to (3) and must therefore be a basic equilibrium payoff. □

Intuitively, if incentive constraints are slack, then it is possible to move the continuation payoffs in the direction of $\mathbf{v}$ without violating incentive constraints or feasibility. Since $\mathbf{v}$ is already presumed to be clockwise d-maximal, the continuation values move weakly in the directions $\widehat{d}$ and d, and strictly in at least one of these directions. This means that the payoffs that we generate with these continuation values have also moved in the direction $\widehat{d}$ or d relative to $\mathbf{v}$, which would violate our hypothesis that $\mathbf{v}$ is already clockwise d-maximal.[11]

Since every extremal equilibrium payoff is clockwise maximal for some direction, Proposition 1 implies that it must be generated by a basic pair. Combining this observation with Lemma 1, we have the following result:

[11]This result has some antecedents in the literature. For example, Kocherlakota (1996) studies the Pareto efficient equilibria of a model of informal insurance, and he shows that ratios of marginal utilities of agents should be held constant over time when incentive constraints are slack. Ligon, Thomas, and Worrall (2000, 2002) make a similar point in the context of a more general class of insurance games. Dixit, Grossman, and Gul (2000) make similar observations in the context of a model of political power-sharing. These results are implied by the sufficiency of basic pairs for generating extremal payoffs, as basic pairs build in the property that as long as incentive constraints do not bind, continuation payoffs must maximize a fixed weighted sum of players' utilities.

Corollary 1 (Number of extremal equilibrium payoffs)**.** *For each s, the number of extreme points of $\mathbf{V}(s)$ is at most $5^{|S|}\prod_{s'\in S}|\mathbf{A}(s')|$.*

This result, like a similar one in AS for the non-stochastic case, is of independent theoretical interest. In the prior literature, it was not even known that the number of extreme points is finite.

4.3 Identifying nearby maximal payoffs

Thus, the extremal equilibrium payoffs are generated by basic pairs, and if we wish to describe the equilibrium payoff correspondence, we may do so by identifying those basic pairs that generate extremal payoffs. At a high level, this approach is analogous to the one used by AS in the context of non-stochastic games. AS showed that there were at most five possibile ways in which a given action pair could generate extreme payoffs: (i) when an incentive constraint binds, the extreme payoff must be generated by one of the four extreme points of the set of binding and feasible continuation values, and (ii) when incentive constraints are slack, the extreme payoff must be generated by infinite repetition of the given action pair, so that the discounted payoff is equal to the flow utility. Indeed, our characterization reduces to this description when the state is perfectly persistent.

Whereas in repeated games the non-binding case can lead to only one payoff per action pair, in stochastic games there can be a multitude of payoffs generated by the same actions when constraints are slack. The reason is that the payoffs that are generated by that action pair depend on behavior in other states. Even if we restrict attention to payoffs that are generated by basic pairs, Corollary 1 indicates that the number of such pairs could be quite large. We would therefore like to have a more efficient procedure for searching for extremal payoffs and the basic pairs that generate them. We shall see that this task of describing the extremal basic pairs is greatly simplified by remarkable properties of basic equilibrium payoffs that mirror those of feasible payoffs as described in Section 3. These are also the features of basic equilibria that will be exploited by our pencil sharpening algorithm in Section 5. For now, we shall exposit this structure in the context of the equilibrium payoff correspondence $\mathbf{V}$.

Let us motivate the analysis as follows. Suppose we have found payoffs $\mathbf{v}^0$ which are maximal for some direction d^0. We may then ask how the maximal payoffs change as the direction of maximality changes. In particular, as the direction rotates clockwise from d^0, there is a first direction d^{T} for which the clockwise maximal payoffs are a tuple $\mathbf{v}^{\mathrm{T}} \neq \mathbf{v}^0$. This d^{T} is in fact the tangent vector from $\mathbf{v}^0(s)$ to $\mathbf{V}(s)$ for the state s in which this tangent has the smallest clockwise angle of rotation from d^0.

The tangent direction d^{T} and the next-clockwise maximal payoffs $\mathbf{v}^{\mathrm{T}}$ undeniably exist. But suppose that all we know is that $\mathbf{v}^{0}$ is maximal. What we would like to have is a way to *discover* the next clockwise payoffs through a computationally tractable procedure. Preliminary to this more ambitious goal, we may ask: is there even a simple procedure that will tell us the *direction* d^{T} in which those payoffs lie? It turns out that there is such a procedure, which we will now explain.

Observe that the directions $\mathbf{d}(s) = \mathbf{v}^{\mathrm{T}}(s) - \mathbf{v}^{0}(s)$ all point (weakly) in the direction d^{T}, in that $\mathbf{d}(s) = \mathbf{x}(s) d^{\mathrm{T}}$ for non-negative scalars $\mathbf{x}(s) \geq 0$ that are not all zero. Let state s^{*} be a state for which this movement attains its maximum value of $\mathbf{x}(s^{*}) > 0$, so that the direction $\mathbf{v}^{\mathrm{T}}(s^{*}) - \mathbf{v}(s^{*}) = \mathbf{x}(s^{*}) d^{\mathrm{T}}$ points in the direction d^{1}. Thus, if we can identify how $\mathbf{v}^{\mathrm{T}}(s^{*})$ is generated, then we would know the tangent direction as well.

From existing theory, we know that $\mathbf{v}^{\mathrm{T}}(s^{*})$ can be decomposed as

$$\mathbf{v}^{\mathrm{T}}(s^{*}) = (1-\delta) g(a^{*}) + \delta w,$$

where a^{*} is the action pair and $w \in \overline{V}(a^{*})$ is the expected discounted continuation value. Moreover, because $\mathbf{v}^{\mathrm{T}}(s^{*})$ is clockwise maximal in $\mathbf{V}(s^{*})$, we know that w falls into one of two categories which can be represented by a regime r^{*}. In one case, incentive constraints are slack, so $r^{*} = \mathrm{NB}$ and w is simply the expectation of $\mathbf{v}^{\mathrm{T}}$:

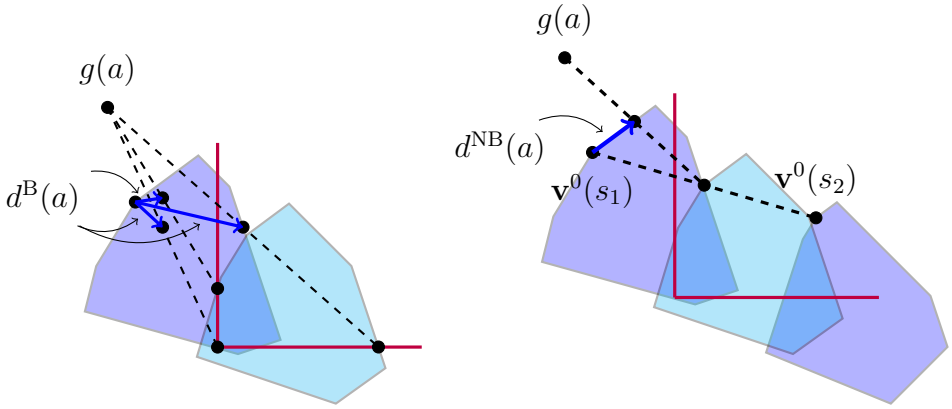
$$w = \sum_{s' \in S} \pi(s'|a^{*}) \mathbf{v}^{\mathrm{T}}(s).$$

In the other case, an incentive constraint binds and $w = r^{*}$ is simply an extreme binding continuation value in $C(a^{*})$.

This suggests that in the latter binding case, there is a simple series of calculations that would reveal $\mathbf{v}^{\mathrm{T}}(s^{*})$ and the direction d^{T}. In particular, for each state $s \in S$, action pair $a \in \mathbf{A}(s)$, and for each $w \in C(a)$, we can construct a *binding test direction*

$$d^{\mathrm{B}}(a, w) = (1-\delta) g(a) + \delta w - \mathbf{v}^{0}(s),$$

which points from the current payoffs in state s towards the payoff that is generated by (a, w). The geometric construction of the binding test directions is depicted in Figure 7a. It is apparent that if $\mathbf{v}^{\mathrm{T}}(s^{*})$ is generated with binding incentive constraints, then there will be a binding test direction that points to $\mathbf{v}^{\mathrm{T}}(s^{*})$, and therefore in the direction d^{T}. Moreover, for any binding test direction, the payoff $\mathbf{v}^{0}(s) + d^{\mathrm{B}}(a, w)$ is generated using feasible and incentive compatible continuation values, and must therefore lie in $\mathbf{V}(s)$. As a result, every binding test direction must point weakly below d^{T}. In effect, the computation of binding



(a) Binding test directions. (b) Non-binding test directions.

Figure 7: Equilibrium test directions.

test directions corresponds to the computation of binding payoffs in AS, adapted to the task of identifying the slope of the boundary $\mathbf{V}$ locally around $\mathbf{v}^0$. In sum, if v^* is generated with binding incentive constraints, then the shallowest binding test direction must be proportional to d^{T}.

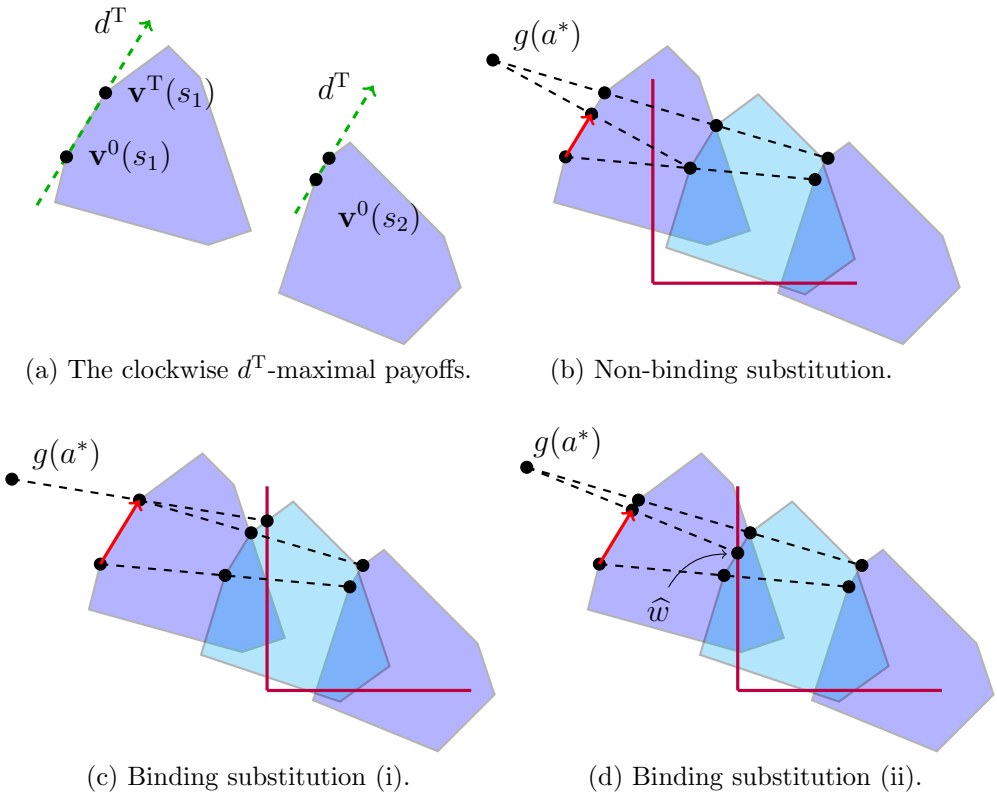
But what about the non-binding case? When incentive constraints are slack, we know that the continuation payoffs are in fact the clockwise maximal payoffs $\mathbf{v}^{\mathrm{T}}$, which are the very objects that we would like to discover. One could adopt a brute force search for $\mathbf{v}^{\mathrm{T}}$, but as we previously observed, the computational complexity of this task is exponential in the number of states. It turns out, however, that the *direction* to $\mathbf{v}^{\mathrm{T}}(s^*)$ can be easily computed by replacing $\mathbf{v}^{\mathrm{T}}$ as continuation values with a payoff tuple that we already know: $\mathbf{v}^0$.

In particular, for actions $a \in \mathbf{A}(s)$, the *non-binding test direction* is defined by

$$d^{\mathrm{NB}}(a) = (1-\delta)\, g(a) + \delta \sum_{s' \in S} \pi(s'|a)\, \mathbf{v}^0(s') - \mathbf{v}^0(s).$$

For our two-state example, we have depicted the non-binding direction in Figure 7b. Let us verify that in the event that $r^* = \mathrm{NB}$, $d^{\mathrm{NB}}(a^*)$ does indeed point in the direction d^{T}. Note that

$$\begin{aligned} d^{\mathrm{NB}}(a^*) &= (1-\delta)\, g(a^*) + \delta \sum_{s' \in S} \pi(s'|a^*)\, \mathbf{v}^0(s') - \mathbf{v}^0(s^*) \\ &= \mathbf{v}^{\mathrm{T}}(s^*) - \mathbf{v}^0(s^*) - \delta \sum_{s' \in S} \pi(s'|a^*) \left(\mathbf{v}^{\mathrm{T}}(s') - \mathbf{v}^0(s')\right) \\ &= \left(\mathbf{x}(s^*) - \delta \sum_{s' \in S} \pi(s'|a^*)\, \mathbf{x}(s')\right) d^{\mathrm{T}}, \end{aligned}$$



(a) The clockwise d^{T}-maximal payoffs.

(b) Non-binding substitution.

(c) Binding substitution (i).

(d) Binding substitution (ii).

Figure 8: Cases for the shallowest test direction. $s^* = s_1$.

and since $\mathbf{x}(s^*) \geq \mathbf{x}(s')$ for all s' and $\delta < 1$, the coefficient on d^{T} is strictly positive.

Thus, if v^* is generated without binding incentive constraints, there will exist a non-binding test direction that is proportional to d^{T}. In contrast to the binding test directions, however, there may well exist non-binding directions that point above d^{T}. The reason is that whereas $d^{\mathrm{NB}}(a)$ always points to a payoff that is feasible (since $\mathbf{v}^0 \in \mathbf{V}$), the associated continuation values need not be incentive compatible. As a result, non-binding test directions may point to non-equilibrium payoffs.

We will say that a test direction d is *incentive compatible* (IC) if the payoff $\mathbf{v}^0(s) + d$ can be generated using feasible and incentive compatible continuation values. Note that all binding test directions are IC by definition. Also, it must be the case that all of the IC non-binding test directions point to equilibrium payoffs, and therefore point weakly below d^{T}. As a result, if $r^* = \mathrm{NB}$ and if $d^{\mathrm{NB}}(a^*)$ is IC, then it must point towards $\mathbf{v}^{\mathrm{T}}(s^*)$. The remaining case to consider is when $r^* = \mathrm{NB}$ but $d^{\mathrm{NB}}(a^*)$ is not IC. The proof of the following proposition deals with this remaining case, in which there must exist a binding test direction that is proportional to d^{T}. Figure 8 illustrates the various possibilities.

Proposition 2 (Test directions)**.** *Let* $\mathbf{v}^0$ *and* d^T *be as given in the preceding discussion. All of the IC test directions* d *point below the direction* d^T*, i.e.,* $d \cdot \widehat{d^T} \leq 0$*. Moreover, there exists an IC test direction* d *which is equal to* xd^T *for some* $x > 0$*.*

Proof of Proposition 2. As $\mathbf{v}^0$ is d^{T}-maximal, the line through d^{T} is a supporting hyperplane of $\mathbf{V}(s)$ for each s. As a result,

$$v \cdot \widehat{d^{\mathrm{T}}} \leq \mathbf{v}^0(s) \cdot \widehat{d^{\mathrm{T}}}$$

for all $v \in \mathbf{V}(s)$. Since IC test directions point to equilibrium payoffs, we conclude that they must all point below d^{T}.

For the second part of the proposition, the remaining case to consider is when $r^* = \mathrm{NB}$ and $d^{\mathrm{NB}}(a^*)$ is not IC. Let

$$w^0 = \sum_{s' \in S} \pi(s'|a^*)\, \mathbf{v}^0(s');$$
$$w^{\mathrm{T}} = \sum_{s' \in S} \pi(s'|a^*)\, \mathbf{v}^{\mathrm{T}}(s').$$

Note that $IC(a^*)$ is closed and convex, $w^0 \notin IC(a^*)$, and $w^{\mathrm{T}} \in IC(a^*)$. As a result, there exists a unique convex combination $w = \alpha w^0 + (1-\alpha) w^{\mathrm{T}}$ that lies on the boundary of $IC(a^*)$. As w^0 and w^{T} are d^{T}-maximal in $\overline{V}(a^*)$, w must also be d^{T}-maximal in $\overline{V}(a^*)$. Moreover, since incentive constraints are slack at w^{T}, d^{T} must point into the interior of $IC(a^*)$ from w. This implies that if w were written as a convex combination of other payoffs in $\overline{V}(a^*)$, those payoffs must also be d^{T}-maximal, and at least one of those payoffs would have to be on the interior of $IC(a^*)$. As a result, w must be an extreme binding payoff, so that $d^{\mathrm{B}}(a^*, w)$ is a binding test direction, and

$$\begin{aligned} d^{\mathrm{B}}(a^*, w) &= \alpha d^{\mathrm{NB}}(a^*) + (1-\alpha)\left(\mathbf{v}^{\mathrm{T}}(s^*) - \mathbf{v}^0(s^*)\right) \\ &= \left(\mathbf{x}(s^*) - \alpha\delta \sum_{s' \in S} \pi(s'|a^*)\, \mathbf{x}(s')\right) d^{\mathrm{T}}, \end{aligned}$$

so there exists a binding test direction that is proportional to d^{T}. □

Thus, even though a given action pair may generate a large number of equilibrium payoffs, when we adopt a local perspective around a given maximal payoff tuple $\mathbf{v}^0$, there is only a sparse set of possibilities that may generate an incremental movement along the boundary. In particular, each action is associated with at most five (easily computable!) test directions, and the shallowest of these test directions must point clockwise along the frontier of $\mathbf{V}$.

4.4 Finding a new basic pair

Proposition 2 shows that there is a shallowest IC test direction d^* that points in the direction d^{T}. This tells us a great deal about the shape of the frontier of $\mathbf{V}$ locally around $\mathbf{v}^0$. In particular, we know that there exist equilibrium payoffs $\mathbf{v}^0(s^*) + d^* \in \mathbf{V}(s^*)$, so that we could generate a new maximal payoff tuple by moving payoffs just in state s^*. Ultimately, though, we would like to go even further and identify the payoffs $\mathbf{v}^{\mathrm{T}}$ that are clockwise d^{T}-maximal. In order to accomplish this task, we will need to make use of more than just knowledge of the slope of the frontier, but also knowledge of which changes to the equilibrium system will lead to the generation of the next clockwise maximal payoffs. It turns out that the correct modifications are implicit in how the shallowest test direction was generated.

In particular, let us further suppose that $\mathbf{v}^0$ are maximal basic equilibrium payoffs generated by a basic pair $(\mathbf{a}^0, \mathbf{r}^0)$. There is a shallowest IC test direction associated with $\mathbf{v}^0$ which points in the direction d^{T}. Moreover, this best test direction d^* is associated with a state s^*, an action a^*, and a continuation regime r^*, where $r^* = \mathrm{NB}$ if $d^* = d^{\mathrm{NB}}(a^*)$ and $r^* \in C(a^*)$ if $d^* = d^{\mathrm{B}}(a^*, r^*)$.

We claim that there is a simple procedure for substituting (a^*, r^*) into $(\mathbf{a}^0, \mathbf{r}^0)$ to create a new basic pair $(\mathbf{a}^1, \mathbf{r}^1)$, so that the new basic pair generates basic equilibrium payoffs $\mathbf{v}^1$ that move in the direction d^{T} relative to $\mathbf{v}^0$. First, we define

$$\mathbf{a}^1(s) = \begin{cases} a^* \text{ if } s = s^*; \\ \\ \mathbf{a}^0(s) \text{ otherwise.} \end{cases}$$

Note that the new action tuple $\mathbf{a}^1$ is identical to $\mathbf{a}^0$ except possibly in state s^*. The new regime tuple will be the limit of a sequence $\left\{\mathbf{r}^{1,k}\right\}_{k=0}^{\infty}$ which is jointly constructed with a sequence of payoff vectors $\left\{\mathbf{v}^{1,k}\right\}_{k=0}^{\infty}$. These sequences begin at

$$\mathbf{r}^{1,0}(s) = \begin{cases} r^* \text{ if } s = s^*; \\ \\ \mathbf{r}^0(s) \text{ otherwise,} \end{cases}$$

and $\mathbf{v}^{1,0} = \mathbf{v}^0$.

Recall that there are unique payoffs that solve the system of equations (3) for the basic pair $(\mathbf{a}^1, \mathbf{r}^{1,0})$. Now, it may be that these payoffs are not equilibrium payoffs because the incentive constraint (4) is violated in some state. The purpose of the iterative procedure is to discover if any incentive constraints will be violated and, if so, modify the basic pair so that it generates equilibrium payoffs by changing non-binding regimes to suitable binding

continuation values. In addition, we will select these binding continuation regimes in order to move the generated payoffs as far as possible in the direction d^{T}.

More specifically, at iteration $k \geq 1$, if $\mathbf{r}^{1,k-1}(s) \neq \mathrm{NB}$, then we simply set $\mathbf{r}^{1,k}(s) = \mathbf{r}^{1,k-1}(s)$ and

$$\mathbf{v}^{1,k}(s) = (1-\delta)\, g\left(\mathbf{a}^1(s)\right) + \delta \mathbf{r}^{1,k}(s).$$

Only if $\mathbf{r}^{1,k-1}(s) = \mathrm{NB}$ will we update payoffs and potentially update regimes. This is accomplished as follows. Suppose $\mathbf{r}^{1,k-1}(s) = \mathrm{NB}$. If the payoff

$$w^{k-1} = \sum_{s' \in S} \pi\left(s' | \mathbf{a}^1(s)\right) \mathbf{v}^{1,k-1}(s') \in IC\left(\mathbf{a}^1(s)\right),$$

so that $\mathbf{v}^{1,k-1}$ are incentive compatible continuation values, then we set $\mathbf{r}^{1,k}(s) = \mathrm{NB}$ and

$$\mathbf{v}^{1,k}(s) = (1-\delta)\, g\left(\mathbf{a}^1(s)\right) + \delta w^{k-1}.$$

If $w^{k-1} \notin IC(\mathbf{a}^1(s))$, then assuming inductively that

$$w^{k-2} = \sum_{s' \in S} \pi\left(s' | \mathbf{a}^1(s)\right) \mathbf{v}^{1,k-2}(s') \in IC\left(\mathbf{a}^1(s)\right),$$

then we identify the unique convex combination $\widehat{w} = \alpha w^{k-2} + (1-\alpha) w^{k-1}$ such that $\widehat{w} \in \mathrm{bd} IC(\mathbf{a}^1(s))$, and we set $\mathbf{r}^{1,k}(s) = \widehat{w}$ and

$$\mathbf{v}^{1,k}(s) = (1-\delta)\, g\left(\mathbf{a}^1(s)\right) + \delta \widehat{w}.$$

We are implicitly assuming that $\widehat{w} \in C(\mathbf{a}^1(s))$, which shall be verified presently.

This completes the specification of the algorithm. Let us define

$$\mathbf{r}^1 = \lim_{k \to \infty} r^{1,k}.$$

The following proposition characterizes our iterative procedure.

Proposition 3 (New basic pair)**.** *The basic pair* $(\mathbf{a}^1, \mathbf{r}^1)$ *in the preceding discussion is well defined. Moreover,* $(\mathbf{a}^1, \mathbf{r}^1)$ *generates basic equilibrium payoffs* $\mathbf{v}^1$*. Finally, there exist non-negative scalars* $\mathbf{x}(s)$ *that are not all zero such that*

$$\mathbf{v}^1(s) = \mathbf{v}^0(s) + \mathbf{x}(s)\, d^T.$$

Proof of Proposition 3. Let us first argue that our procedure is well defined. For any state such that $\mathbf{r}^{1,0} = \text{NB}$, the payoffs

$$w^0 = \sum_{s' \in S} \pi\left(s' | \mathbf{a}^1(s)\right) \mathbf{v}^0(s)$$

must be incentive compatible for $\mathbf{a}^1(s)$. If $s = s^*$, then $r^* = \text{NB}$, so the non-binding test direction must have been IC. On the other hand, if $s \neq s^*$, then it must have been that $\mathbf{r}^0(s) = \text{NB}$ as well, and incentive compatibility of w^0 follows from the hypothesis that $\mathbf{v}^0(s)$ are equilibrium payoffs. As a result, w^{k-1} is incentive compatible for the base step at $k = 1$.

Now, notice that $\mathbf{v}^{1,1}(s) = \mathbf{v}^0(s)$ when $s \neq s^*$, and $\mathbf{v}^{1,1}(s^*) = \mathbf{v}^0(s^*) + d^*$. Thus, at the first iteration, payoffs in all states move (weakly) in the direction d^*. Let us inductively suppose that on previous iterations $l < k$, $\mathbf{v}^{1,l} - \mathbf{v}^{1,l-1} = \mathbf{x}^l d^*$ for a tuple of non-negative scalars $\mathbf{x}^l$. As a result, all of the payoffs $\mathbf{v}^{1,l}$ with $l < k$ are d^{T}-maximal (since d^* is proportional to d^{T}). Clearly, if $\mathbf{r}^{1,k-1}(s) = \text{NB}$ and $\mathbf{v}^{1,k-1}$ is incentive compatible for $\mathbf{a}^1(s)$, then

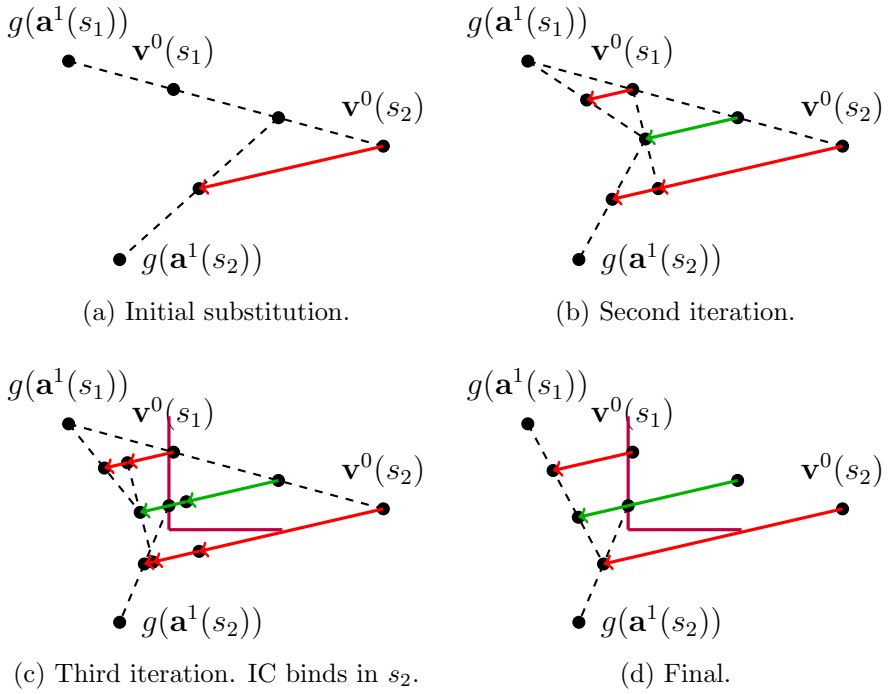
$$\begin{aligned}
\mathbf{v}^{1,k}(s) - \mathbf{v}^{1,k-1}(s) &= \delta \sum_{s' \in S} \pi\left(s' | \mathbf{a}^1(s)\right) \left(\mathbf{v}^{1,k-1}(s') - \mathbf{v}^{1,k-2}(s')\right) \\
&= \delta \left(\sum_{s' \in S} \pi\left(s' | \mathbf{a}^1(s)\right) \mathbf{x}^{l-1}(s') \right) d^*.
\end{aligned}$$

On the other hand, if an incentive constraint is violated at $\mathbf{v}^{1,k-1}$, then since $\mathbf{v}^{1,k-1}$ and $\mathbf{v}^{1,k-2}$ are both d^1-maximal in $\mathbf{V}$, the payoff $\widehat{w}$ must be d^1-maximal in $\overline{V}(\mathbf{a}^1(s))$. Since it also lies on the boundary of $IC(\mathbf{a}^1(s))$, it must be an extreme binding continuation value. In this case,

$$\begin{aligned}
\mathbf{v}^{1,k}(s) - \mathbf{v}^{1,k-1}(s) &= \delta \sum_{s' \in S} \pi\left(s' | \mathbf{a}^1(s)\right) \left(\alpha \mathbf{v}^{1,k-1}(s') + (1-\alpha)\mathbf{v}^{1,k-2}(s') - \mathbf{v}^{1,k-2}(s')\right) \\
&= \alpha\delta \left(\sum_{s' \in S} \pi\left(s' | \mathbf{a}^1(s)\right) \mathbf{x}^{l-1}(s) \right) d^*,
\end{aligned}$$

so that the movement is still proportional to d^{T}.

Finally, let us argue that the algorithm converges and that $(\mathbf{a}^1, \mathbf{r}^1)$ generates basic equilibrium payoffs. While $\mathbf{r}^{1,k}$ is not changing, our procedure is essentially iteratively applying the Bellman operator of equation (5) which, as we have previously observed, is a contraction of modulus δ. Thus, the iterates $\mathbf{v}^{1,k}$ converge at a geometric rate to the unique fixed point. Now suppose that $\left(\mathbf{a}^1, \mathbf{r}^{1,k}\right)$ generates payoffs that are not equilibrium payoffs because (4) is



(a) Initial substitution.

(b) Second iteration.

(c) Third iteration. IC binds in s_2.

(d) Final.

Figure 9: The Bellman procedure.

violated for some state s. In that case, after finitely many iterations, an incentive constraint will be violated by $\mathbf{v}^{1,k}$, and $\mathbf{r}^{1,k}(s)$ will be changed to a binding regime. Since there are only finitely many states, there can be only finitely many switches from non-binding to binding regimes, $\mathbf{r}^{1,k}$ must converge to some $\mathbf{r}^1$ after finitely many iterations. At this point, $\mathbf{v}^{1,k}$ converges to the fixed point, which must be incentive compatible. Indeed, $\mathbf{v}^1 = \lim_{k\to\infty} \mathbf{v}^{1,k}$ are the payoffs generated by $(\mathbf{a}^1, \mathbf{r}^1)$, and since (4) is satisfied, $\mathbf{v}^1$ are basic equilibrium payoffs. □

This Bellman procedure is depicted graphically in Figure 9. In this example, $\mathbf{r}(s_1)$ is non-binding at every iteration, whereas $\mathbf{r}(s_2)$ starts out non-binding but eventually transitions to binding. In Figure 9a, the initial substitution is made that moves payoffs in state s_2 in a south-westerly direction. Figure 9b shows the second iteration, in which the movement in state s_2 is propagated through to state s_1. Through the second iteration, the expected pivots are incentive compatible in both states. At iteration three, however, the incentive constraint in state s_2 would be violated by using the pivot as continuation values. As a result, we fix the payoff in s_2 at the binding constraint, but move the continuation payoff for state s_1 all the way to the expected pivot. This is depicted in Figure 9c. In Figure 9d, we see the final configuration of the new pivot.

Note that we have described this procedure as if there are infinitely many iterations. In practice, we have implemented this procedure (and its analogue in Section 5) by inverting the system of equations (3) for each $\left(\mathbf{a}^1, \mathbf{r}^{1,k}\right)$ and checking if the resulting payoffs satisfy the incentive constraint (4). If not, we iterate as above until a constraint is violated, switch regimes from non-binding to binding in states where constraints are violated, and then invert again. After finitely many iterations, the payoffs obtained by inverting (3) must be incentive compatible.

4.5 Tracing the frontier

An immediate application of Propositions 1, 2, and 3 is an algorithm to trace the frontier of the equilibrium payoff correspondence $\mathbf{V}$ through basic pairs and basic equilibrium payoffs. This procedure requires as inputs an initial maximal basic pair $(\mathbf{a}^0, \mathbf{r}^0)$ and corresponding maximal basic equilibrium payoffs $\mathbf{v}^0$, which we refer to as the *pivot*. We first search over all IC test directions to find the shallowest test direction d^*, and its corresponding substitution (a^*, r^*). By Proposition 2, this test direction must point clockwise along the frontier of $\mathbf{V}$. Using the updating procedure of Section 4.4, we substitute (a^*, r^*) into $(\mathbf{a}^0, \mathbf{r}^0)$ to create a new basic pair $(\mathbf{a}^1, \mathbf{r}^1)$. By Proposition 3, this pair generates a new pivot $\mathbf{v}^1$ which moves clockwise along the frontier relative to $\mathbf{v}^0$. Moreover, if we set $d^1 = d^*$, then the payoffs $\mathbf{v}^1$ are d^1-maximal in $\mathbf{V}$. We then inductively repeat the procedure to generate a new basic pair $(\mathbf{a}^2, \mathbf{r}^2)$, pivot $\mathbf{v}^2$, and direction d^2, and so on. At every step, the payoffs move clockwise along the frontier. By Corollary 1, there are only finitely many basic pairs, so that after finitely many steps, we must have visited all of the basic pairs that generate clockwise maximal equilibrium payoff tuples, at which point the pivot has moved all the way around the frontier.

Of course, this procedure requires knowledge of the threat point $\underline{\mathbf{v}}$ and the available extreme binding continuation values $C(a)$ in order to measure incentive compatibility and compute binding test directions. Since it assumes knowledge of the very objects which we want to compute, this algorithm is of limited practical value for computing $\mathbf{V}$ from scratch. It is, however, of great deal didactic value, since our general algorithm in the next section is essentially a generalization of this procedure to the case where we only have approximations of $\underline{\mathbf{v}}$ and $C(a)$.

5 Calculating equilibrium payoffs

5.1 A general pencil sharpening algorithm

At the end of the last section, we described an algorithm for tracing the frontier of the equilibrium payoff correspondence. This procedure pivots between basic pairs by making changes one state at a time, and the substitutions are chosen so as to move the basic equilibrium payoffs clockwise along the boundary of $\mathbf{V}$. The obvious limitation of this algorithm is that it requires knowledge of which binding payoffs are available so that we know which binding directions can be generated. This in turn requires knowledge of $\mathbf{V}$ itself, which is of course the very object that we wish to compute. We will now develop a method for calculating $\mathbf{V}$ which only relies on the primitive specification of the game. This algorithm replaces foreknowledge of the equilibrium payoffs with generous estimates of which binding continuation values and threat payoffs are available in equilibrium, in a manner analogous to the algorithm of APS. By progressively refining the estimates, we get closer and closer to the correct sets of binding payoffs and threats that can be recursively generated in equilibrium.

Recall the APS procedure for calculating $\mathbf{V}$, which starts with a large initial correspondence $\mathbf{W}^0$ that contains $\mathbf{V}$. From this correspondence, we can calculate $B(\mathbf{W}^0)$, which is the correspondence of all tuples of payoffs that can be generated from incentive compatible continuation values in $\mathbf{W}^0$. If we iterate this operator, we know that the sequence of correspondences so generated will eventually converge to $\mathbf{V}$. An equivalent interpretation of this process is that at the first iteration, $B(\mathbf{W}^0)$ consists of those payoffs that can be generated with equilibria that promise arbitrary continuation values in $\mathbf{W}^0$ after the first period. At the second iteration, $B^2(\mathbf{W}^0)$ consists of those payoffs that can be generated by equilibria that promise arbitrary continuation values in $\mathbf{W}^0$ after *two* periods. Inductively, the more times we iterate the APS operator, the further we push back in time the date at which we promise the arbitrary continuation values, and as this date gets further away, the geometric discounting means that these arbitrary payoffs comprise a smaller and smaller share of the discounted expected payoff at time zero. Thus, intial error in specifying $\mathbf{W}^0$ is driven to zero through the force of discounting.

The operator B generates payoffs using any and all available incentive compatible continuation values, which is implicitly generating all kinds of equilibria. We know from Proposition 1, however, that it is without loss of generality to restrict attention to the much smaller class of equilibria that are implicitly represented by basic pairs, if one only wants to generate extremal equilibrium payoffs. These equilibria build in the property that when incentive constraints are slack, continuation payoffs must be maximal in the same direction as the equilibrium payoff itself. Indeed, the feasible payoff algorithm of Section 3 employed this

logic, in the absence of any incentive constraints, by keeping track of a tuple of pivot payoffs that maximize in a common direction. The pivot was recursively generated as our "best guess" of the payoffs that should be used as continuation values. In Section 4 this logic was adapted to take account of incentive constraints.

Our algorithm for equilibrium payoffs will employ an approach that is a hybrid between those of APS and of Section 4. Like APS, the algorithm will generate a sequence of approximate equilibrium payoff correspondences, where each correspondence will be generated with continuation values drawn from the previous approximation. Thus, initial errors will be driven out over time through discounting. Unlike APS, we will only generate payoffs using a suitable generalization of basic pairs, and it is only when incentive constraints are binding that we will use arbitrary and approximate continuation values. Each complete revolution of the pivot yields a new approximate equilibrium payoff correspondence, from which new binding payoffs and threats are drawn when generating the next revolution of the pivot, and so on.

We will first give an overview of the procedure, with additional details in the following subsections. The algorithm proceeds over a series of iterations, over the course of which we will generate a sequence of pivot payoff tuples $\mathbf{v}^k$ and accompanying action and regime tuples $\left(\mathbf{a}^k, \mathbf{r}^k\right)$. As before, $\mathbf{r}^k(s)$ is either NB (the non-binding case) or it is a payoff vector in $\mathbb{R}^2$ which gives the fixed and exogenous continuation payoffs. We will also keep track of a *current direction* d^k that satisfies

$$\mathbf{V} \subseteq H\left(\mathbf{v}^k, d^k\right) = \left\{ \mathbf{v} \,\middle|\, \mathbf{v}(s) \cdot \widehat{d}^k \leq \mathbf{v}^k(s) \cdot \widehat{d}^k \ \forall s \in S \right\}.$$

This means that the equilibrium payoff correspondence is always below $\mathbf{v}^k$ in levels with slope d^k. In addition, we will maintain a compact and convex payoff correspondence $\mathbf{W}^k$, which contains the equilibrium payoff correspondence and serves as our approximation of the payoffs that can be promised as binding continuation values on the equilibrium path. This $\mathbf{W}^k$ will in fact be the set of payoffs that have been circumscribed by the trajectory of the pivot $\mathbf{v}^k$ thus far. We will separately maintain a *threat tuple* $\underline{\mathbf{w}}^k \leq \underline{\mathbf{v}}$ that approximates the equilibrium punishment payoffs that players receive after a deviation. These payoffs will be used to determine the minimal incentive compatible continuation value $h(a)$. The algorithm can be initialized with any $\mathbf{v}^0$, d^0, $\mathbf{W}^0$, and $\underline{\mathbf{w}}^0$ that satisfy these conditions. The initial $\mathbf{a}^0$ and $\mathbf{r}^0$ can be arbitrary as long as $\mathbf{r}^0(s) \neq \text{NB}$ for all s.

At each iteration, we will search over test directions that can be generated from $\mathbf{v}^k$ in a manner analogous to how we traced the frontier in Section 4, using $\mathbf{W}^k$ and $\underline{\mathbf{w}}^k$ in lieu of $\mathbf{V}$ and $\underline{\mathbf{v}}$ for calculating the extreme binding continuation values in $C(a)$. The direction

which generates the smallest clockwise angle relative to d^k will be deemed *shallowest*, and it is generated by some action a^* and a regime r^* in state s^*. This shallowest direction will become the new current direction d^{k+1}. We then substitute this new action and regime into the system (3), and advance the pivot in the new direction using the same sequence of operations that we employed to trace the frontier of $\mathbf{V}$. We will argue that the direction identified by the algorithm will necessarily satisfy

$$\mathbf{V} \subseteq H\left(\mathbf{v}^k, d^{k+1}\right) = H\left(\mathbf{v}^{k+1}, d^{k+1}\right),$$

so that our approximations will continue to contain all of the equilibrium payoffs. We refer to this property as *local containment*.

The algorithm proceeds over a sequence of such iterations, through which the pivot tuple proceeds to move in a clockwise fashion, spiraling around and around the equilibrium payoff sets. Because of the local containment property, the trajectory of the pivot will never enter the interior of the equilibrium payoff correspondence. In addition, we will show that the algorithm cannot get "stuck", in the sense that starting from any iteration, a new revolution will be completed in finitely many steps. Our convention will be that the new revolution begins when the d^k passes due north, i.e., when d^{k-1} points somewhere to the west of due north, and d^k points somewhere to the east. The index of the iteration can therefore be decomposed as $k = r : c$, where r is the number of revolutions and c is the number of steps, or cuts, within the revolution. The current revolution and cut are denoted by $r(k)$ and $c(k)$, respectively. With slight abuse of notation, we will continue to write $k+1 = r+1 : 0$ if $k+1$ starts a new revolution and $k + 1 = r : c + 1$ otherwise. Over the course of a revolution, the local containment property implies that the pivot will travel around all edges of the equilibrium payoff set. As a result, the area that is encircled by the pivot over the course of a revolution must contain $\mathbf{V}$.

Now, in order for our tracing procedure to get closer to $\mathbf{V}$, we need the approximate binding continuation values and threat tuples to get closer to their equilibrium counterparts as well. To that end, $\mathbf{W}^k$ will be set equal to the set that has been circumscribed by the trajectory of the pivot up to this point in the algorithm:

$$\mathbf{W}^k = \mathbf{W}^0 \cap \left(\cap_{l=0}^{k} H\left(\mathbf{v}^l, d^l\right)\right),$$

and we will set $\underline{\mathbf{w}}^k = \underline{\mathbf{w}}\left(\mathbf{W}\right)$. Note that by definition the sets $\mathbf{W}^k$ are monotonically decreasing and the $\underline{\mathbf{w}}^k$ are monotonically increasing.

With additional details that will be specified below, we will argue that the sequence $\left\{\mathbf{W}^k\right\}_{k=0}^{\infty}$ converges to $\mathbf{V}$. First, containment implies that $\mathbf{W}^k$ will always contain $\mathbf{V}$, so

the sequence cannot converge to anything smaller. In addition, when new payoffs enter the pivot, they are generated using continuation values in $\mathbf{W}^k$. Since $\mathbf{W}^k$ is contained in the last revolution of the pivot, we have new pivots in the current revolution being inductively generated from the pivot's trajectory on the previous revolution. This is analogous to how the payoffs in the APS sequence are recursively generated from the previous correspondence in the sequence. In the limit, only those payoffs persist which can be perpetually bootstrapped in this manner, which rules out payoffs that are not in $\mathbf{V}$.

5.1.1 Finding the new substitution

Let us now provide more detail as to how we find the substitution that the algorithm makes at iteration k. As we previously indicated, we will generate a series of test directions, each of which is associated with a particular state, action, and continuation regime. As in Section 4, we will only generate test directions using feasible and incentive compatible continuation values, although these notions will be measured relative to our current approximation of $\mathbf{V}$. In particular, when the action is a, the set of feasible expected continuation values is

$$\overline{W}(a) = \sum_{s' \in S} \pi(s'|a) \mathbf{W}^{r(k):0}(s').$$

In other words, continuation values are considered feasible at iteration k only if they were feasible *at the beginning of the current revolution*. Similarly, incentive compatibility will be measured relative to the threat point $\underline{\mathbf{w}}^{r(k):0}$ at the beginning of the current revolution. Thus,

$$h_i(a) = \max_{a_i'} \left[\frac{1-\delta}{\delta} \left(g_i(a_i', a_j) - g_i(a) \right) + \sum_{s' \in S} \pi(s'|a_i', a_j) \underline{\mathbf{w}}^{r(k):0} \right]$$

and

$$IC(a) = \left\{ w \in \mathbb{R}^2 | w \geq h(a) \text{ for some } i \right\}$$

are respectively the minimum incentive compatible expected continuation value and the set of incentive compatible payoffs for action a. Note that $\overline{W}(a)$, $h(a)$, and $IC(a)$, are all constant within a revolution. The purpose of keeping these objects constant within a revolution is to simplify our subsequent convergence arguments.

The algorithm will generate candidate test directions at iteration k for each state s and action a. These test directions will be used to determine whether or not the action a may be substituted into the basic pair in state s, and if so, with what continuation regime. A

subset of those directions will be considered *admissible*, and the substitution will correspond to the shallowest admissible direction. We say that the test direction

$$d = (1-\delta)\, g\,(a) + \delta w - \mathbf{v}^k\,(s)$$

is (i) *feasible* if $w \in \overline{W}\,(a)$, (ii) *incentive compatible* if $w \geq h\,(a)$, and (iii) *non-zero* if $d \neq 0$. If the test direction d satisfies (i-iii), then we shall simply say it is *admissible*. Admissibility is almost but not quite sufficient for the algorithm to use the test direction, as we shall see.

The test directions generated by our algorithm fall into four categories. First, let

$$\widetilde{w}(a) = \sum_{s' \in S} \pi(s'|a)\mathbf{v}^k(s')$$

denote the expected pivot when actions a are played. The *non-binding test direction*

$$d^{\mathrm{NB}}(a) = (1-\delta)g(a) + \delta\widetilde{w}(a) - \mathbf{v}^k(s)$$

points from the pivot to the payoff that would be generated by playing a for one period and using the pivot itself as continuation values.

Second, let

$$C\,(a) = \mathrm{ext}\left(\overline{W}(a) \cap \mathrm{bd}IC(a)\right)$$

denote the set of extreme feasible and binding expected continuation values. The *binding test directions* are of the form

$$d^{\mathrm{B}}(a, w) = (1-\delta)g(a) + \delta w - \mathbf{v}^k(s),$$

for $w \in C(a)$.

Together, we refer to the non-binding and binding test directions as *regular test directions*. These directions have direct counterparts in our procedure for tracing the frontier of the equilibrium payoff set in Section 4, and they are the primary paths that are considered by our algorithm. In a simpler world, they would be the only directions we have to consider, but alas, this is not the case. Recall that the APS operator B is monotonic, so that if $B\,(\mathbf{W}^0) \subseteq \mathbf{W}^0$, then the APS sequence will be monotonically decreasing towards $\mathbf{V}$. The analogous property that one might hope to obtain for the current procedure is that the trajectory of the pivot moves closer and closer to $\mathbf{V}$ in a monotonic fashion in that $\mathbf{v}^{k+1} \in \mathbf{W}^k$ for all k, perhaps given a similar assumption about the initial conditions. Unfortunately, this notion of monotonicity will not be satisfied if we use only regular test directions. In other

words, there will sometimes be an admissible regular test direction d that is *non-monotonic*, in the sense that $\mathbf{v}^k(s)+d \notin \mathbf{W}^k(s)$. If we were to incorporate the substitution corresponding to such a direction, the pivot would necessarily move out of $\mathbf{W}^k$. Intuitively, the pencil sharpening algorithm is more selective than the APS operator in its choice of continuation values when generating payoffs, and the continuation values that are chosen can change in complicated ways as the threat tuple and incentive constraints are updated.[12,13]

Now, all of our definitions up to this point remain sensible even if the pivot is not in contained in the current feasible set. We will see, however, that infeasibility of the pivot is problematic for our convergence argument. Specifically, we will subsequently argue that the pivot will encircle $\mathbf{V}$ and never move into its interior, but our proof depends on the pivot payoffs themselves being feasible as continuation values. We will therefore impose that the pivot stay within $\mathbf{W}^k$ as an extra constraint by only considering those substitutions associated with test directions that are both admissible and *monotonic*, in that $\mathbf{v}^k + d \in \mathbf{W}^k$.[14]

For each action a, the algorithm will first consider the associated regular test directions. If the admissible regular test directions are all monotonic, then nothing special happens, and these are the only test directions that will be generated for a. If, however, an admissible regular test direction turns out to be non-monotonic, then this direction will *not* be used. Simply disallowing non-monotonic but otherwise admissible directions would create separate problems for our containment argument. Thus, to ensure containment, we will replace the non-monotonic direction with additional test directions as follows. Let

$$Gen\left(a\right) = \left(1-\delta\right) g\left(a\right) + \delta \overline{W}\left(a\right) \cap IC\left(a\right)$$

denote the set of payoffs that can be generated with action a using feasible and incentive compatible continuation values. The third kind of test direction is the *frontier test direction*,

[12]We note that these kinds of non-monotonicities also arose in the algorithm of AS for non-stochastic games. They develop a different and, in a sense, more classical algorithm, which did not exploit monotonicity in order to establish containment. Indeed, the APS sequence $B^k\left(\mathbf{W}^0\right)$ will asymptotically converge to $\mathbf{V}$ as long as $\mathbf{W}^0 \supseteq \mathbf{V}$, though for an arbitrary initial condition, the sequence need not be monotonically decreasing.

[13]This could not happen in the procedure for tracing the frontier of $\mathbf{V}$ in Section 4, since the existence of an admissible and non-monotonic test direction would contradict the hypothesis that $\mathbf{V}$ contains all of the equilibrium payoffs.

[14]Note that this constraint rules out two related but conceptually distinct forms of non-monotonicities. Since $\mathbf{W}^k$ is contained in $H\left(\mathbf{v}^k, d^k\right)$, monotonicity rules out directions that "bend backwards" by pointing above d^k. At the same time, monotonicity rules out directions that, while not back bending, would nonetheless cause the pivot to move outside of $\mathbf{W}^k$. These forms of non-monotonicity would cause distinct problems for our convergence arguments. If we admitted back-bending directions, it is no longer clear what shallowest means, and even with a suitable definition, there seems to be scope for the algorithm to stall (cf. Lemmas 6 and 10). On the other hand, if $\mathbf{v}^k \notin \mathbf{W}^k$, we could not prove, at a key step in Lemma 4, that there exists a binding test direction that does not cut into $\mathbf{V}$.

which is of the form

$$d^{\mathrm{F}} = v - \mathbf{v}^k(s),$$

where $v \in Gen(a)$ and $v \in \mathrm{bd}\mathbf{W}^k$. For the purposes of the rest of the algorithm, frontier directions are treated as binding directions, so that the continuation regime associated with the frontier test direction is simply the fixed continuation payoffs that are used to generate the direction. Note that the frontier test direction is admissible (when it is non-zero) and monotonic by construction.

The fourth and final test direction arises in the rather exceptional case that the only frontier test direction is the zero direction. In this case, we generate *interior test direction*s of the form

$$d^I = -xd$$

for some $x > 0$. Again, such a direction is admissible and monotonic if

$$\mathbf{v}^k(s) - xd \in Gen(a) \cap \mathbf{W}^k(s),$$

and it is treated as a binding test direction for the purposes of the rest of the algorithm.

This completes the specification of our search procedure. We collect all of the non-binding and binding, and (if required) frontier and interior, test directions. We select the shallowest of these that is both admissible and monotonic as the new current direction. We will subsequently show that there always exists an admissible and monotonic test direction d such that $H\left(\mathbf{v}^k, d\right)$ contains $\mathbf{V}$. The shallowest admissible and monotonic test direction will therefore not intersect the equilibrium payoff correspondence. It is the shallowest direction d^* which is chosen as d^{k+1}, and the corresponding substitution (a^*, r^*) is incorporated into the system that determines the pivot.

5.1.2 The rest of the iteration

Having identified (a^*, r^*), we substitute these into the configuration of the system that defines the pivot. The pivot will be updated as in our tracing procedure from Section 4. This involves incrementally moving the pivot in the best direction d^* by iteratively recomputing payoffs in states in which the regime is non-binding, using the pivot itself as the continuation value. If the advancing continuation value ever violates an incentive constraint, we switch that state's regime to the binding continuation value that lies between the expected pivot and the previous continuation value, and cease updating that state. For fixed regimes, the movement of the pivot is contracting with modulust δ. Thus, there is a unique fixed point to which the iterates are converging at a geometric rate, and if an incentive constraint is

violated at the fixed point, it will also be violated after a finite number of steps. Moreover, since there are only finitely many states and regimes can only switch from non-binding to binding, the regime tuple will converge after finitely many steps, at which point the payoffs must converge to the new pivot.

There are, however, two modest differences between the pivot updating in Section 4 and the present general algorithm. First, the general algorithm uses $\underline{\mathbf{w}}^{r(k):0}$ as the threat point rather than $\underline{\mathbf{v}}$ for the purpose of determining $h(a)$ and incentive compatibility. Second, since we are imposing monotonicity of the pivot, we will require that the updating step not take the pivot outside of $\mathbf{W}^k$. If this is going to happen in state s, we simply fix $\mathbf{v}^k(s)$ at the point where it is just about to move outside of $\mathbf{W}^k(s)$, and change its regime to binding with the appropriate continuation value. With these modifications, our process results in a new configuration for the pivot $\left(\mathbf{a}^{k+1}, \mathbf{r}^{k+1}\right)$ and pivot payoffs $\mathbf{v}^{k+1}$. After having updated the pivot, we conclude the iteration by generating a new payoff correspondence $\mathbf{W}^{k+1}$ and updating the threat tuple to $\underline{\mathbf{w}}\left(\mathbf{W}^{k+1}\right)$.

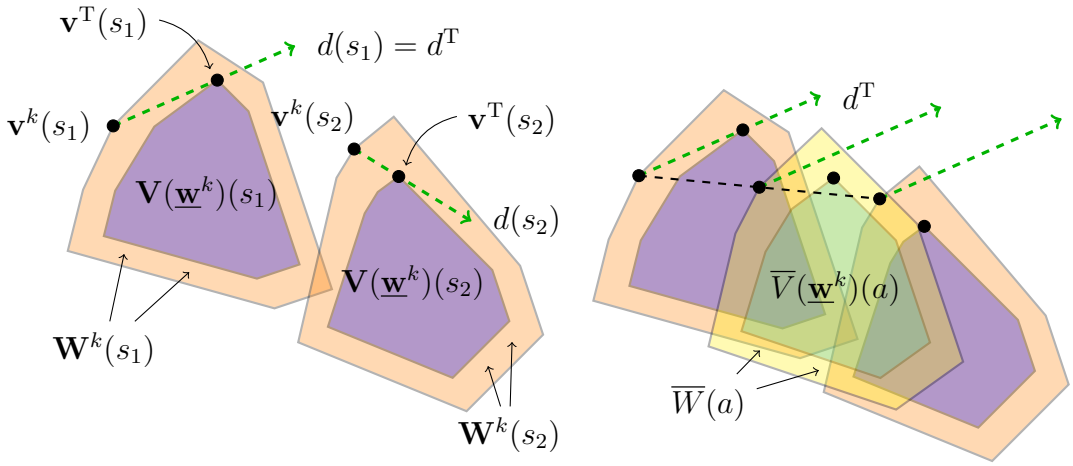
5.2 Containment

We assert that the trajectory generated by this procedure will eventually converge to the equilibrium payoff correspondence, in the sense that $\cap_{k\geq 0}\mathbf{W}^k = \mathbf{V}$. This assertion will be verified in several steps. First, we will show that as long as our approximation $\mathbf{W}^k$ contains $\mathbf{V}$, then the algorithm will necessarily find an admissible direction d^{k+1} that does not "cut into" $\mathbf{V}$, in the sense that $\mathbf{V} \subseteq H(\mathbf{v}^k, d^{k+1})$. We refer to this property as *local containment*. Local containment implies that the pivot will orbit around $\mathbf{V}$ in a clockwise manner, so that the sequence of trimmed approximations $\mathbf{W}^k$ will contain $\mathbf{V}$.

This conclusion is, however, dependent upon the pivot completing new revolutions. The second piece of our argument shows that it is impossible for the algorithm to stall, in that starting from any iteration, the pivot will complete a full revolution in finitely many steps. The algorithm must consequently complete infinitely many revolutions.

Finally, every payoff that is generated by the pencil sharpening algorithm is also a payoff that would be generated by the algorithm of APS. This leads to the conlusion that two revolutions of the pivot are contained in one iteration of the APS operator (see Lemma 7, below). Since the latter generates a sequence of approximations that converge to $\mathbf{V}$, the trajectory of the pivot will be forced to get close to $\mathbf{V}$ as well.

We now present in greater detail our containment argument. First, we formally state that as long as containment is satisfied up to iteration k, then the algorithm will find an admissible and monotonic test direction.



(a) Tangent directions.

(b) Expected continuations.

Figure 10: The tangent directions for an example with two states $S = \{s_1, s_2\}$. $d^{\mathrm{T}} = d(s_1)$ is the shallowest tangent among all of the $d(s)$.

Lemma 3 (Existence)**.** *Suppose that* $\mathbf{V} \subseteq \mathbf{W}^k$*. Then there exists an admissible and monotonic test direction.*

As a result, the shallowest test direction $d^* = d^{k+1}$ is well-defined. The following lemma establishes the inductive step of local containment.

Lemma 4 (Local containment)**.** *Suppose that* $\mathbf{V} \subseteq \mathbf{W}^k$*. Then* $\mathbf{V} \subseteq H\left(\mathbf{v}^{k+1}, d^{k+1}\right)$*, so that* $\mathbf{V} \subseteq \mathbf{W}^{k+1}$*.*

While Lemmas 3 and 4 are conceptually distinct, we shall prove them simultaneously by establishing the existence of an admissible and monotonic test direction that would satisfy containment if it were selected by the algorithm. Since d^{k+1} must be shallower than this particular direction, it will a fortiori also satisfy containment. Here we present the overview of the argument, with the omitted details in the Appendix.

We will show that the algorithm will find a new direction d^{k+1} such that by traveling in this direction from $\mathbf{v}^k$, the pivot will not cut into the partial equilibrium payoff correspondence $\mathbf{V}\left(\underline{\mathbf{w}}^k\right)$. If that is so, then the pivot will not cut into $\mathbf{V}\left(\underline{\mathbf{w}}\right)$ for any punishment tuple $\underline{\mathbf{w}} \geq \underline{\mathbf{w}}^k$, including the equilibrium payoff correspondence $\mathbf{V}$ itself. To prove that this will not happen, we will show that there is some admissible direction d that is shallow enough for the pivot to miss $\mathbf{V}\left(\underline{\mathbf{w}}^k\right)$. The direction chosen by the algorithm to be d^{k+1} must be weakly shallower than this particular direction, so that containment will be satisfied.

Now on to specifics. For each state, there is a clockwise tangent direction from $\mathbf{v}^k(s)$ to $\mathbf{V}\left(\underline{\mathbf{w}}^k\right)(s)$, which we denote by $d(s)$. Let d^{T} denote the shallowest of all of these tangents

across all states, which is attained in states in $S^{\mathrm{T}} \subseteq S$, and recall that $\widehat{d}^{\mathrm{T}}$ is the counter-clockwise normal to d^{T}. We write $\mathbf{v}^{\mathrm{T}}$ for the payoff tuple that is clockwise d^{T}-maximal in $\mathbf{V}\left(\underline{\mathbf{w}}^k\right)$, and note that $\mathbf{v}^{\mathrm{T}}(s)$ is the tangent point from $\mathbf{v}^k(s)$ to $\mathbf{V}\left(\underline{\mathbf{w}}^k\right)(s)$ for $s \in S^{\mathrm{T}}$. We depict these objects in Figure 10a. Figure 10b depicts expected continuation values and provides useful context for the figures and arguments below. Also write $\mathbf{a}^{\mathrm{T}}$ and $\mathbf{w}^{\mathrm{T}}$ for the tuples of actions and continuation values that generate $\mathbf{v}^{\mathrm{T}}$, i.e.,

$$\mathbf{v}^{\mathrm{T}}(s) = (1-\delta)g\left(\mathbf{a}^{\mathrm{T}}(s)\right) + \delta \mathbf{w}^{\mathrm{T}}(s) \ \forall s \in S.$$

Let $\widetilde{\mathbf{w}}(s)$ denote the expected pivot when using action $\mathbf{a}^{\mathrm{T}}(s)$:

$$\widetilde{\mathbf{w}}(s) = \sum_{s' \in S} \pi\left(s' | \mathbf{a}^{\mathrm{T}}(s)\right) \mathbf{v}^k(s').$$

Also, let $\mathbf{w}^*(s)$ denote the clockwise d^{T}-maximal element of $\mathrm{bd}IC\left(\mathbf{a}^{\mathrm{T}}(s)\right) \cap \overline{W}\left(\mathbf{a}^{\mathrm{T}}(s)\right)$, which is necessarily an element of $C\left(\mathbf{a}^{\mathrm{T}}(s)\right)$. We denote by

$$d^{\mathrm{B}}(s) = (1-\delta)g\left(\mathbf{a}^{\mathrm{T}}(s)\right) + \delta \mathbf{w}^*(s) - \mathbf{v}^k(s)$$
$$d^{\mathrm{NB}}(s) = (1-\delta)g\left(\mathbf{a}^{\mathrm{T}}(s)\right) + \delta \widetilde{\mathbf{w}}(s) - \mathbf{v}^k(s)$$

the "best" binding test direction, in the sense of shallowest relative to d^{T}, and the non-binding test direction generated by $\mathbf{a}^{\mathrm{T}}(s)$.

We will argue that one of the actions $\mathbf{a}^{\mathrm{T}}(s)$ for some $s \in S^{\mathrm{T}}$ will generate an admissible test direction that is shallower than d^{T}. Since this action will be considered by our algorithm over the course of the search for the new substitution, the test direction that is ultimately chosen as shallowest must be weakly shallower than this particular test direction. To simplify matters, we will assume that there is some state $s \in S^{\mathrm{T}}$ such that both $d^{\mathrm{NB}}(s)$ and $d^{\mathrm{B}}(s)$ are non-zero. We do not make this assumption in the proof in the Appendix, and it is a generic possibility that one or both of these objects will be zero. Nonetheless, it is a convenient case to consider for providing intuition for how the containment argument works.

Let us first argue that one of $d^{\mathrm{NB}}(s)$ and $d^{\mathrm{B}}(s)$ for some $s \in S^{\mathrm{T}}$ must be generated by feasible and incentive compatible continuation values and points "above" the shallowest tangent (in the sense that $d \cdot \widehat{d}^{\mathrm{T}} \geq 0$ for d being one of these directions). First, the direction $d^{\mathrm{NB}}(s)$ is *always* above d^{T}, regardless of whether or not it is incentive compatible. Why? Since d^{T} is the shallowest tangent across all states, it must be that $\mathbf{v}^k(s')$ lies above $\mathbf{V}\left(\underline{\mathbf{w}}^k\right)(s')$ in the direction $\widehat{d}^{\mathrm{T}}$ for all $s' \in S$. But this means that $\widetilde{\mathbf{w}}(s)$, the expected pivot, is also above any feasible continuation payoff in $\overline{V}\left(\underline{\mathbf{w}}^k\right)\left(\mathbf{a}^{\mathrm{T}}(s)\right)$ in the direction $\widehat{d}^{\mathrm{T}}$.

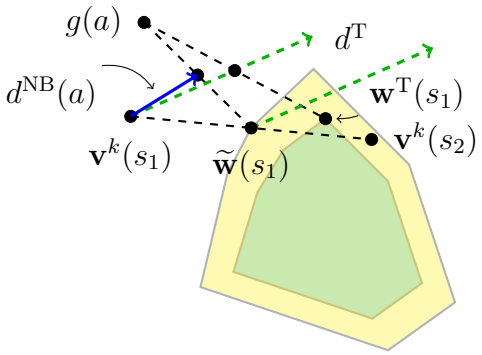


Figure 11: The non-binding direction is IC, and necessarily is shallower than d^{T}.

Hence, the payoff generated using $\mathbf{a}^{\mathrm{T}}(s)$ and $\widetilde{\mathbf{w}}(s)$ is higher than any payoff in $\mathbf{V}\left(\underline{\mathbf{w}}^k\right)(s)$ as well, and it is higher than $\mathbf{v}^{\mathrm{T}}(s)$ in particular. As a result, the direction from $\mathbf{v}^k(s)$ towards $(1-\delta)g\left(\mathbf{a}^{\mathrm{T}}(s)\right)+\delta\widetilde{\mathbf{w}}(s)$ is shallower than the direction from $\mathbf{v}^k(s)$ to $\mathbf{v}^{\mathrm{T}}(s)$, which is d^{T}, and we conclude that $d^{\mathrm{NB}}(s)$ is above d^{T}. If $\widetilde{\mathbf{w}}(s)$ is incentive compatible as well, then $d^{\mathrm{NB}}(s)$ will be both feasible and incentive compatible and above the shallowest tangent. This situation is depicted in Figure 11. Thus, whenever the non-binding direction is admissible, it must be shallower than the shallowest direction, and therefore moving the pivot in this direction will not cause it to intersect the interior of the equilibrium payoff set.

On the other hand, consider the case that $\widetilde{\mathbf{w}}(s)$ is not incentive compatible, so that the non-binding direction is not admissible. Since $\widetilde{\mathbf{w}}(s)$ is not in $IC\left(\mathbf{a}^{\mathrm{T}}(s)\right)$ but $\mathbf{w}^{\mathrm{T}}(s)$ is, there must be a unique convex combination $w=\alpha\widetilde{\mathbf{w}}(s)+(1-\alpha)\mathbf{w}^{\mathrm{T}}(s)$ that is on the boundary of $IC\left(\mathbf{a}^{\mathrm{T}}(s)\right)$. Moreover, both $\widetilde{\mathbf{w}}(s)$ and $\mathbf{w}^{\mathrm{T}}(s)$ are in the convex set $\overline{W}\left(\mathbf{a}^{\mathrm{T}}(s)\right)$, and therefore so is w.[15] Thus, w is in $\overline{W}\left(\mathbf{a}^{\mathrm{T}}(s)\right)\cap IC\left(\mathbf{a}^{\mathrm{T}}(s)\right)$. In addition, since $\widetilde{\mathbf{w}}(s)$ is higher in the $\widehat{d}^{\mathrm{T}}$ direction than is $\mathbf{w}^{\mathrm{T}}(s)$, w is higher in this direction as well, and so $\mathbf{w}^*(s)$ (being the clockwise d^{T}-maximal point in $\mathrm{bd}IC\left(\mathbf{a}^{\mathrm{T}}(s)\right)\cap\overline{W}\left(\mathbf{a}^{\mathrm{T}}(s)\right)$) is also above $\mathbf{w}^{\mathrm{T}}(s)$. This implies that the payoff generated using $\mathbf{w}^*(s)$, which is $\mathbf{v}^k(s)+d^{\mathrm{B}}(s)$, is higher than $\mathbf{v}^{\mathrm{T}}(s)$, so that $d^{\mathrm{B}}(s)$ is above d^{T}. This is the situation depicted in Figure 12.

We conclude that at least one of $d^{\mathrm{NB}}(s)$ or $d^{\mathrm{B}}(s)$ will be generated by feasible and incentive compatible continuation values and point above d^{T}. If this direction points into $\mathbf{W}^k(s)$, then we have shown the existence of an admissible and monotonic direction that is shallower than d^{T}, so that the shallowest admissible and monotonic test direction must be weakly shallower than this particular direction, and therefore d^{T}. It still might be the case, however, that one of these directions is admissible but not monotonic, in which case we have to check if there is a frontier or interior direction that will satisfy containment.

[15] This is the step in the containment argument that uses the hypothesis that $\mathbf{v}^k\in\mathbf{W}^k$ and necessitates monotonicity.

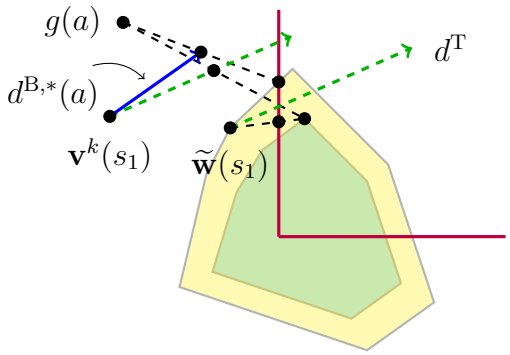


Figure 12: The non-binding direction is not incentive compatible: $\widetilde{\mathbf{w}}(s_1)$ is outside the IC region.

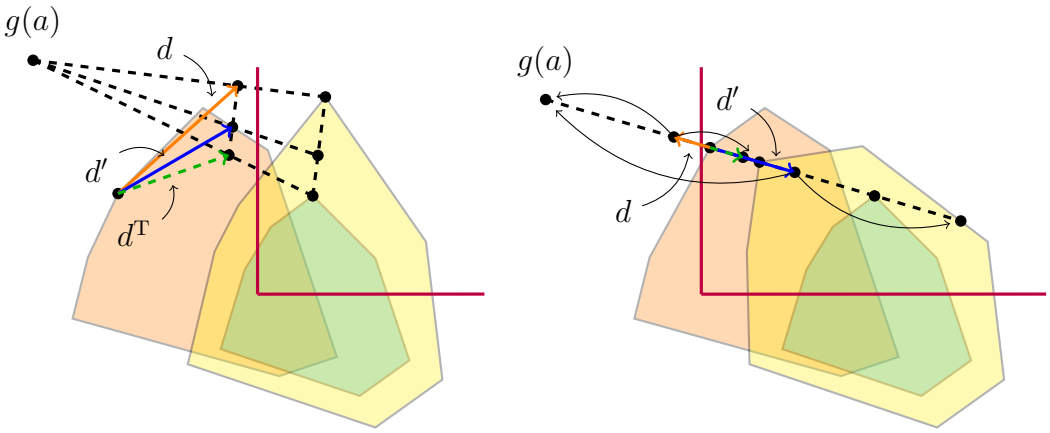
Intuitively, the direction d^{T} itself can be generated in state s using feasible and incentive compatible payoffs, and this direction can never be non-monotonic by the inductive hypothesis of containment. At the same time, there is some non-monotonic direction d that points towards a payoff $v = \mathbf{v}^k(s) + d$ that can also be generated by feasible and incentive compatible continuation values. This implies that there is a payoff $v' = \alpha v + (1-\alpha)\mathbf{v}^{\mathrm{T}}(s)$ that can also be generated and is on the boundary of $\mathbf{W}^k$. If $v' \neq \mathbf{v}^k(s)$, we conclude that there is a non-zero frontier direction $d^{\mathrm{F}} = v' - \mathbf{v}^k(s)$ that is admissible. Moreover, d^{F} is a convex combination of d and $d^{\mathrm{T}}(s)$, and since both of those directions are both weakly shallower than d^{T}, d^{F} must be weakly shallower than d^{T} as well.

This situation is depicted in Figure 13a. The orange test direction d is non-monotonic because it points to payoffs outside of $\mathbf{W}^k$. There is, however, a convex combination of that payoff and the tangent payoff (which is pointed to by the dashed green arrow d^{T}) that lies on the frontier. The frontier direction points to this payoff.

Finally, what if $v' = \mathbf{v}^k(s)$? This means that $\mathbf{v}^k(s)$ is itself a convex combination of v and $\mathbf{v}^{\mathrm{T}}(s)$, which can only happen if d is proportional to $-d^{\mathrm{T}}$. But in that case, there is an interior direction that is admissible and is *equal* to the shallowest tangent (cf. Figure 13b).

We conclude that even if $d^{\mathrm{NB}}(s)$ and $d^{\mathrm{B}}(s)$ are non-monotonic, either a frontier direction or an interior direction must be admissible and is shallower than d^{T}. The algorithm will therefore find a new direction that does not cut into the equilibrium payoff set, so the pivot will travel around $\mathbf{V}$ in a clockwise manner. This result implies that

Lemma 5 (Containment)**.** $\mathbf{V} \subseteq \mathbf{V}\left(\underline{\mathbf{w}}^k\right) \subseteq \mathbf{W}^k$ *for all* k*.*



(a) A frontier direction is admissible.

(b) An interior direction is admissible.

Figure 13: Frontier and interior directions. In both cases, the non-monotonic direction is in orange, while the frontier/interior direction is in blue.

5.3 No stalling

Having established that $\mathbf{W}^k$ will not converge to anything smaller than $\mathbf{V}$, we now argue that it also cannot converge to anything larger. A key step is showing that the algorithm cannot stall, in the sense that starting from any iteration, the pivot will complete a full revolution around $\mathbf{V}$ in finitely many steps.

Let us be more precise about our definition of revolutions. Let $d^N = (0, 1)$ denote the direction "due north." We say that an iteration *starts a revolution* if (i) $d^{k-1} \cdot \widehat{d}^N > 0$ or $d^{k-1} = xd^N$ for some $x > 0$ and (ii) $d^k \cdot \widehat{d}^N < 0$ or $d^k = -xd^N$ for some $x > 0$. Loosely speaking, d^{k-1} points weakly to the west, and d^k points weakly to the east. We will refer to a subsequence of iterations of the form $\{l|(r, -1) \leq l \leq (r + 1, 0)\}$ as a *complete revolution*.

Our anti-stalling result is that starting from any k, there exists a $k' > k$ such that the sequence $\{k, \ldots, k'\}$ contains a complete revolution. The logic behind this result is as follows. The algorithm must find a new admissible and monotonic test direction at every iteration. If the pivot stopped completing new revolutions around the equilibrium payoff correspondence, then these admissible directions must get stuck at some limit direction, which we denote by d^∞. Thus, for l sufficiently large, $\mathbf{v}^l$ will be strictly increasing in the direction of d^∞.

New test directions can only be generated by four methods: non-binding, binding, frontier, and interior. Moreover, new pivots are always generated as the solution to the system (3) for some configuration of actions and continuation regimes. Since there are only finitely many states and actions, if the binding payoffs can only take on one of finitely many values, then there are only finitely many ways to configure (3) to generate different pivots. This would be at odds with our hypothesis that there are infinitely many pivots being generated

that all increase in the direction d^∞. We therefore conclude that there must be infinitely many new binding payoffs being introduced into the system.

Now, recall that the sets $\overline{W}(a)$ and $IC(a)$ are defined from feasible payoffs at the beginning of the revolution. As a result, if the pivot gets stuck and is no longer completing revolutions, the sets $C(a)$ of extreme binding continuation values that can be used to generate new binding test directions are not changing either, and these infinitely many new binding payoffs must be coming from elsewhere. In particular, they must be coming from (i) hitting an IC or monotonicity constraint during the pivot update procedure, at which point the regime for that state is changed from non-binding to binding, or (ii) from a frontier or interior direction, in which the pivot travels as far as it can go in the given direction while maintaining feasibility, incentive compatibility, and monotonicity.

However, (i) or (ii) cannot occur more than once with a given action if d^l is sufficiently close to d^∞. Let us suppose for the sake of exposition that d^l is exactly d^∞ and is not changing from iteration to iteration. If, say, the best direction at iteration l is a frontier direction generated by action a, then the pivot will travel as far as possible in the direction d^∞ while staying within the set of payoffs that can be generated by a using feasible and incentive compatible continuation values, intersected with $\mathbf{W}^k(s)$. This set is a compact and convex polytope that is not changing between iterations since new revolutions are not being completed. Thus, if $\mathbf{v}^l(s)$ is already maximized in the direction d^∞, then at subsequent iterations, it will be impossible to move further in this direction using action a without violating either feasibility or incentive compatibility. Even if d^l is only very close to d^∞, this will still be true, because the set of payoffs that can be generated by a has only finitely many extreme points, so eventually d^l will be close enough to d^∞ that moving in any direction between d^l and d^∞ would cause the pivot to become infeasible.

Thus, (i) and (ii) can only happen finitely many times, so that the existence of new directions will eventually require the pivot to complete new revolutions. We therefore have the following:

Lemma 6 (No stalling)**.** *The pivot completes infinitely many revolutions, i.e.,*

$$\lim_{k\to\infty} r(k) = \infty.$$

5.4 Convergence

We are almost done. The algorithm generates an infinite sequence of monotonically decreasing approximations, each of which must contain the equilibrium payoff correspondence. Thus, the sequence $\mathbf{W}^k$ converges and it must converge to something weakly larger than

$\mathbf{V}$. It remains to be shown that $\mathbf{W}^k$ converges to $\mathbf{V}$ exactly. The proof of this fact will conveniently piggy-back on the convergence results of APS. All along, we have maintained that our algorithm generates payoffs in a similar manner as the APS algorithm, though it is more selective as to which continuation payoffs can be used. It should therefore be no surprise that the sequence $\mathbf{W}^k$ is contained in the APS sequence:

Lemma 7 (Dominated by APS)**.** *Suppose that* $B(\mathbf{W}^{0:0}) \subseteq \mathbf{W}^{0:0}$*. Then* $\mathbf{W}^{2r:0} \subseteq B^r(\mathbf{W}^{0:0})$*.*

This may seem surprising: every *two* revolutions of the pivot are contained in *one* iteration of the APS operator. Let us provide some intuition for this result. Consider the initial state of the algorithm on iteration r, with $\mathbf{v}^{2r:0} \in \mathbf{W}^{2r:0}$. When we look for the best direction, we generate new payoffs using either the expected pivot as a continuation value or some fixed payoff in $\mathbf{W}^{2r:0}$ as a continuation value, so by the inductive hypothesis we are generating payoffs using continuation values in $B^r\left(\mathbf{W}^{0:0}\right)$. Moreover, this continuation payoff must be incentive compatible with respect to threats that are larger (and therefore weaker) than those used by APS. Thus, any payoff that is used by the pencil sharpening algorithm to generate a test direction is also available to the APS operator and is in $B^{r+1}\left(\mathbf{W}^{0:0}\right)$. Incidentally, because the APS sequence is monotonically decreasing, the new payoffs that are generated are also in $B^r\left(\mathbf{W}^{0:0}\right)$ (which contains $B^{r+1}\left(\mathbf{W}^{0:0}\right)$. This means that the payoffs that are generated during the pivot update procedure are also in $B^{r+1}\left(\mathbf{W}^{0:0}\right)$. Thus, any elements of the pivot which are modified after iteration $2r:0$ must also be in $B^{r+1}\left(\mathbf{W}^{0:0}\right)$.

Now, containment (together with our full dimension hypothesis) implies that each element of the pivot must be updated at least once over the course of a complete revolution. Hence, at the end of the $2r+1$th revolution, every element of the pivot must be in $B^{r+1}\left(\mathbf{W}^{0:0}\right)$. This means that *all* of the pivots on the second revolution are contained in $B^{r+1}\left(\mathbf{W}^{0:0}\right)$, so

$$\operatorname{co}\left(\cup_{k=2r+1:0}^{2(r+1):0}\{\mathbf{v}^k\}\right) \subseteq B^{r+1}\left(\mathbf{W}^{0:0}\right).$$

The only remaining piece of the argument is to relate the trajectory of the pivot to the new feasible set. Recall that $\mathbf{W}^{2(r+1):0}$ is defined as the intersection of $\mathbf{W}^{0:0}$ and all of the half-spaces generated by cuts up to the beginning of the $2(r+1)$th revolution. This set must be weakly smaller than the intersection of the half spaces on just the $2r+1$th revolution, so that

$$\mathbf{W}^{2(r+1):0} \subseteq \cap_{k=2r+1:0}^{2(r+1):0} H\left(\mathbf{v}^k, d^k\right) = X.$$

We show in the Appendix that the set X is contained in the convex hull of the trajectory of the pivot, i.e.,

$$X \subseteq \text{co}\left(\cup_{k=2r+1:0}^{2(r+1):0}\{\mathbf{v}^k\}\right)$$

Combining these observations, we conclude that

$$\mathbf{W}^{2(r+1):0} \subseteq B^{r+1}\left(\mathbf{W}^{0:0}\right),$$

as desired.

The end result is that the APS sequence squeezes the $\mathbf{W}^k$ so that they cannot converge to anything larger than the equilibrium payoff correspondence. We also know from containment that $\mathbf{W}^k$ cannot converge to anything smaller than $\mathbf{V}$. Combining Lemmas 5, 6, and 7, we have our main result:

Theorem 1 (Convergence)**.** *The sequence of approximations* $\{\mathbf{W}^k\}_{k=0}^{\infty}$ *converges to the equilibrium payoff correspondence, i.e.,* $\cap_{k=0}^{\infty}\mathbf{W}^k = \mathbf{V}$.

We note that in practice the pencil sharpening algorithm is likely to converge faster than APS, even though our theoretical result relies on the coarse bound of two revolutions of the pivot being contained in one round of APS. The reason is that pencil sharpening generates many fewer points than APS, thereby quickly eliminating payoffs that otherwise would have to be driven out through the force of discounting.

6 Applications

6.1 A risk sharing example

We will now illustrate our algorithm and methodology by solving a game of informal insurance in the vein of Kocherlakota (1996).[16] Our primary purpose in this exercise is to demonstrate the efficacy and efficiency of our algorithm for solving rich and complex models that are of substantial economic interest. Each period, player $i \in \{1, 2\}$ has an endowment of consumption good $e_i \in [0, 1]$. The total amount of the good in the economy is constant at $e_1 + e_2 = 1$, so that we can simply write $e = e_1$ and $e_2 = 1 - e$ (cf. Ljungqvist and Sargent, 2004, ch. 20). Thus, e describes the state of the world, which was denoted by s in the preceding sections. The endowment evolves stochastically over time, depending on the current endowment and actions taken by the players, as we shall see.

[16] See also Dixit, Grossman, and Gul (2000), Ligon, Thomas, and Worrall (2000, 2002), and Ljungqvist and Sargent (2004).

The good cannot be stored and must be consumed each period. If player i consumes c_i in a given period, then the flow utility is

$$u(c_i) = \sqrt{c_i}.$$

Note that the flow utility function is concave, so that players prefer to smooth their consumption over time. For example, if players could perfectly smooth their endowment so that $c_i = 0.5$ in every period, then the resulting payoffs would be $\sqrt{0.5} \approx 0.705$. On the other hand, if a player were to consume their own endowment in each period, payoffs would be somewhat lower. For example, if the endowment were independently and uniformly distributed each period, then average utility across states would be only $\int_{x=0}^{1} \sqrt{x} dx \approx 0.667$.

Thus, if possible, the players would like to insure themselves against the riskiness of the endowment. While there are no formal insurance contracts available, the players make unilateral transfers that smooth out one another's consumption. Let t_i denote the transfer that player i makes to player j, and let $t = t_1 - t_2$ denote the net transfer from player 1 to player 2. Again, writing $c = c_1$ for player 1's consumption, player 2's consumption must be $c_2 = 1 - c$. Thus, the realized consumption profile is

$$(c, 1-c) = (e - t, 1 - e + t).$$

In our subsequent equilibrium analysis, we will restrict attention to action profiles in which at most one player makes a positive transfer. This is without loss of generality, since any net transfer can be replicated with such an action profile, while at the same time relaxing incentive constraints for both players.

As we have said, the endowment evolves stochastically over time. We will consider a law of motion in which tomorrow's endowment is correlated with today's consumption. In particular, we assume that tomorrow's state e' is distributed according to the density

$$f(e'|c) = \frac{\rho \exp(-\rho |e' - c|)}{2 - \exp(-\rho c) - \exp(-\rho (1-c))}$$

where $\rho \geq 0$ parametrizes the degree of persistence of the endowment around consumption. This density is symmetric around a mode of c with exponential tails that have a shape parameter ρ. As ρ goes to infinity, the distribution converges to a Dirac measure on c, and as ρ goes to zero, the distribution converges to the standard uniform. The density is plotted for various parameter values in Figure 14.

An economic interpretation of the technology for generating new endowments is that each player's productivity is tied to their health and diet, so that players who are well-

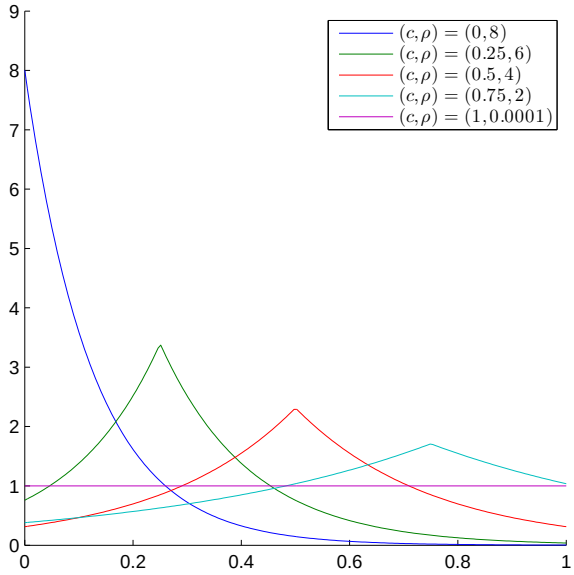


Figure 14: The density $f\left(e'|c\right)$ for various values of c and ρ.

nourished will, on average, be more productive in the subsequent period. At the same time, there is a fixed amount of resources in the economy, so that one player is well-nourished only if the other player is somewhat malnourished. As an aside, we regard the perfect negative correlation between endowments as somewhat artificial, but it facilitates an apples-to-apples comparisons between economies with different levels of persistence, in the sense that all of these economies could attain the same maximally efficient level of welfare with perfect insurance. Thus, the persistence of the endowment affects welfare in equilibrium only through incentives and not by changing the aggregate resource constraint of the economy.

For the purposes of the computation, both endowment and consumption were taken to be on finite grids

$$e \in E = \left\{0, \frac{1}{K_e - 1}, \ldots, \frac{K_e - 2}{K_e - 1}, 1\right\};$$
$$c \in C = \left\{0, \frac{1}{K_c - 1}, \ldots, \frac{K_c - 2}{K_c - 1}, 1\right\},$$

with K_e grid points for the endowment and K_c grid points for consumption. The consumption grid was chosen to include the endowment grid, so that $K_c = 1 + L\left(K_e - 1\right)$ for a positive integer $L \geq 1$. We adapted the continuous probabilities above by assigning to each level of the endowment e' the mass in the bin $\left[e' - 1/\left(2K_e\right), e' + 1/\left(2K_e\right)\right]$.

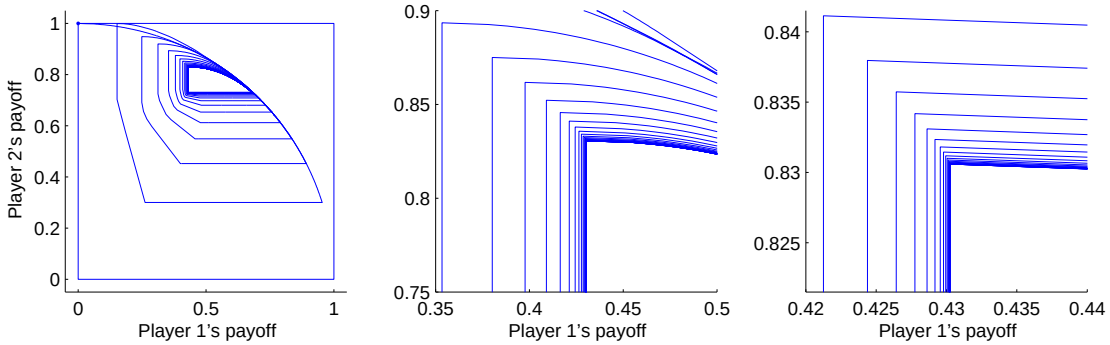


Figure 15: The trajectory of the pivot in state $e = 0$, with $\delta = 0.7$, $K_e = 5$, $K_c = 201$, and $\rho = 0$. The second and third panels show the trajectory around the northwestern corner of $\mathbf{V}(0)$ at greater magnification.

We used our algorithm to compute the equilibria of the risk sharing game for various discount factors, grid sizes, and values of the persistence parameter ρ. Let us take a moment to briefly describe the software that was used to perform the computation. We implemented the pencil sharpening algorithm in a C++ library which we call SGSolve. This library contains a series of classes that operationalize our model and algorithm; for example, there is a class for representing a stochastic game, there is a class for representing the current approximation $\mathbf{W}^k$, and so on. SGSolve is written in a general manner and can be used to solve any game, not just the risk sharing example that we consider here. We note that the algorithm as implemented differs slightly from the algorithm as specified in Section 5, in that the program only generates the binding and non-binding test directions, which may cause the pivot to move non-monotonically. However, the algorithm checks for a sufficient condition that containment will be satisfied, and it emits a warning if the sufficient condition fails.[17] We have also written a graphical interface called SGViewer for interacting with the library. Both the library and the viewer are available through the authors' website.[18] Precompiled versions of the viewer can be downloaded for Windows, OS X, and Linux, and the source code can also be downloaded under the terms of the GPLv3 license.[19]

[17]The key step in our containment argument that relies on monotonicity is showing the existence of a binding test direction that is shallower than the shallowest tangent, in the event that the non-binding test direction is not incentive compatible. If the pivot is not feasible, then the shallowest binding test direction may in fact cut into the equilibrium payoff correspondence. However, a sufficient condition for a given binding test direction associated with an action a to not cut into the set is that it is shallower than the slope of the frontier of $Gen(a)$ at the binding payoff that generates the test direction. SGSolve verifies that this is the case whenever a binding test direction is selected as the best direction, and emits a warning if it is not shallower.

[18]www.benjaminbrooks.net/software.shtml

[19]http://www.gnu.org/licenses/gpl-3.0.en.html

Let us now describe the computations. Figures 15 and 16 present output of the algorithm for the parameter values $\delta = 0.85$, $K_e = 5$, $K_c = 101$, and $\rho = 0$ (so that the endowment is i.i.d. uniform). The algorithm terminated when the distance between successive approximations was less than 10^{-8}. This condition was met after 84 revolutions and 52,766 cuts and a run time of one minute and nine seconds, and the final sets have 637 maximal payoff tuples. Figure 15 shows the path taken by the pivot $\mathbf{v}^k(0)$ over the course of the first 20,000 cuts. Figure 16 shows the trajectory of the pivots on the final revolution, which comprise our last best approximation of the equilibrium payoff correspondence. The equilibrium payoff sets are outlined in blue and overlap one another along a northwest-southeast axis. As player 1's endowment e increases from 0 to 1, payoffs for player 1 generally increase and payoffs for player 2 generally decrease.

We will use this particular computation to review key properties of the equilibrium payoff correspondence that are known from prior work, before considering computations with other parameter values. We note that the following properties hold for *all* parameter values, not just the ones used for the computation in Figure 16 (e.g., for $\rho > 0$). First, notice that all of the equilibrium payoff sets have right angles at their southwest corners, which are highlighted with blue circles. These corner payoffs, which coincide with the threat tuple $\underline{\mathbf{v}}$, are generated by the "autarkic" equilibrium in which neither player ever makes positive transfers. Indeed, this equilibrium must generate the minimum payoff, since it is always a feasible deviation for a player to keep their endowment for themself, which must generate at least the autarky payoff.

Second, notice that the Pareto frontiers of the equilibrium payoff sets all lie along a common frontier, which is indicated by an extra thick blue line. This is again a consequence of special features of the example. Notice that the evolution of the endowment only depends on current consumption, and not on the current value of the endowment itself. As a result, the feasible expected continuation value sets, which we can write as $\overline{V}(c)$, are independent of e. This implies that

$$X(c) = (1-\delta)\, u(c) + \delta \overline{V}(c)$$

is independent of e as well. Moreover, a player's best deviation is always to choose a transfer of zero, which results in a payoff of exactly $\underline{\mathbf{v}}(e)$. Thus, the set of payoffs that can be generated when the endowment is e is simply

$$X \cap \{v | v \geq \underline{\mathbf{v}}(e)\}$$

where

$$X = \cup_{c \in C} X(c),$$

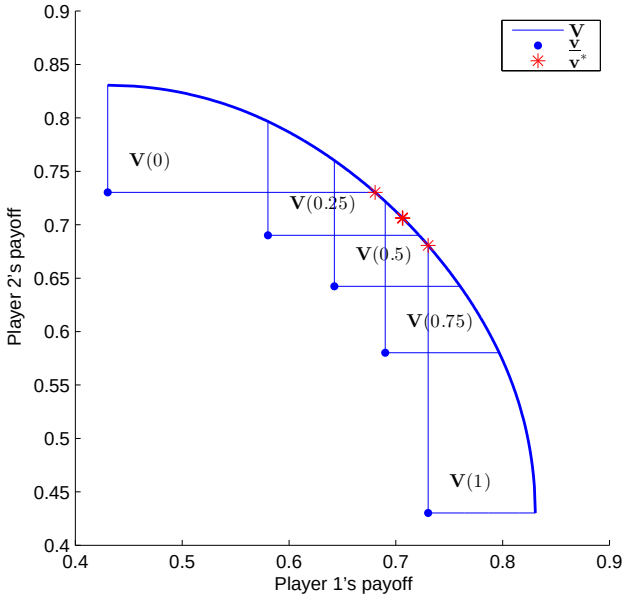


Figure 16: The equilibrium payoff correspondence for $\delta = 0.7$, $K_e = 5$, $K_c = 201$, and $\rho = 0$.

and the common northeastern frontier is simply the Pareto frontier of X.

Third, Figure 16 allows us to see, in a vivid way, the recursive structure we characterized in Section 4. Consider the payoff tuple $\mathbf{v}^*$ that maximizes the sum of players' payoffs, which is depicted with red stars. Since the Pareto frontiers of $\mathbf{V}(0.25)$, $\mathbf{V}(0.5)$, and $\mathbf{V}(0.75)$ overlap at the 45 degree line, it must be that the utilitarian efficient payoffs coincide for these states, i.e., $\mathbf{v}^*(0.25) = \mathbf{v}^*(0.5) = \mathbf{v}^*(0.75)$. Moreover, due to the structure described in the previous paragraph, it must be the same consumption vector which generates all of these payoffs, which is in fact $c = 0.5$. Indeed, since constraints are slack at these levels of the endowment and at this payoff, we know that perfect insurance with consumption constant at 0.5 will obtain until the endowment reaches 0 or 1. Thus, the general structure described in Section 4 implies the particular structure in the example which has previously been described by Kocherlakota (1996) and others.

Having explained this structure, let us now explore how equilibrium payoffs change with the level of persistence. Figure 17 presents four computations for $\rho \in \{0, 4, 9, 14\}$, with $\delta = 0.85$, $K_e = 9$, and $K_c = 201$. Intuitively, the higher is ρ, the more persistent is the endowment around consumption. This has the effect of tightening incentive constraints, because when a player deviates by grabbing more consumption today, they also make themselves more likely to have a high endowment in the future. As a result, deviations induce a transition to autarky in a relatively favorable state, which weakens the punishment. When ρ is equal to zero, so that the endowment tomorrow is uniformly distributed regardless of consumption

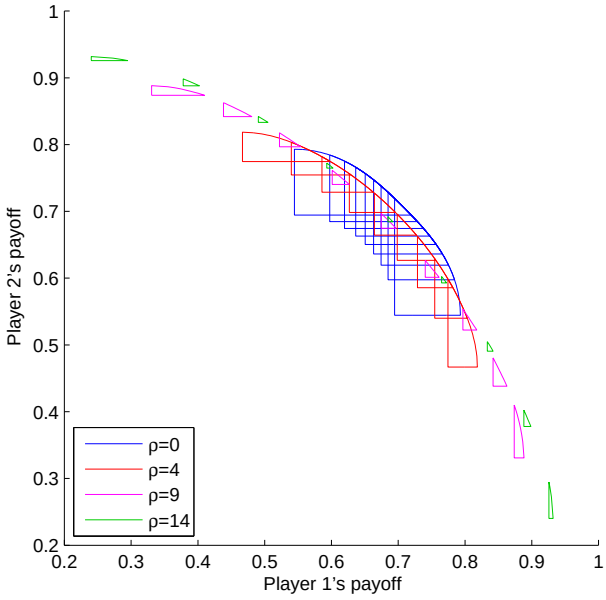


Figure 17: Equilibrium payoff correspondences for $\delta = 0.85$, $K_e = 9$, $K_c = 201$, and various values of ρ.

today, it is in fact possible to implement perfect insurance, which maximizes the sum of the players' utilities. This can be seen because $\underline{\mathbf{v}}_1(1) = \underline{\mathbf{v}}_2(0) < \mathbf{v}^*(e) = \sqrt{0.5}$, so that there is a region of the frontier of X that is incentive compatible both when $e = 0$ and when $e = 1$. This part of the frontier is generated by $c = 0.5$. As ρ increases, the equilibrium payoff sets spread out along the northwest-southeast axis, and even for $\rho = 4$ it is no longer possible to support perfect insurance. Indeed, when $\rho = 9$, incentive constraints are so tight that consumption can only be held constant over time if the endowment does not change.

We will now provide two other and complementary visualizations of how payoffs change with the level of persistence. For each level of the endowment, there is a rich set of possible equilibrium payoffs that could obtain. Figure 18 focuses on a particular point on the efficient frontier, namely the Nash bargaining payoffs. An interpretation of these payoffs is as follows. Suppose that the endowment starts at some particular level, and players play the Nash bargaining game to decide which equilibrium should be implemented, where the threat point is the autarky equilibrium. Figure 18 shows how the payoffs that result from such bargaining depend on the degree of persistence and on the initial state. The results are not terribly surprising. The player who starts with the higher endowment has an advantage in bargaining, and this advantage generally increases as the endowment becomes more persistent.

For our last visualization, we consider the relationship between persistence and the surplus that is generated by insurance over the long run. It is well known that starting from

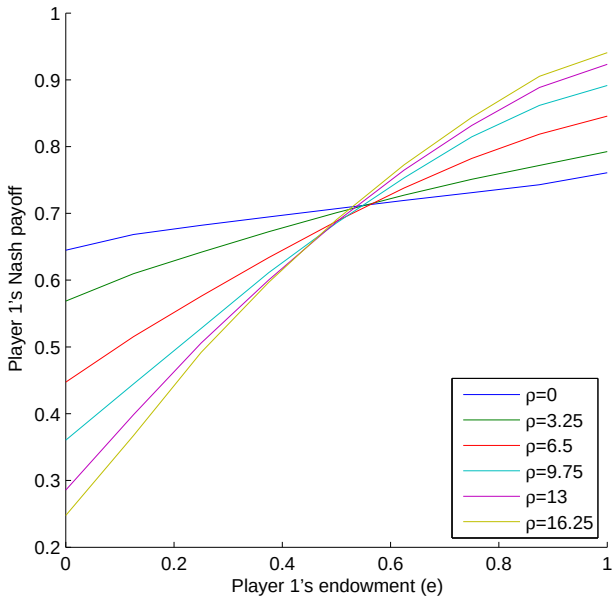


Figure 18: Nash bargaining payoffs for $\delta = 0.85$, $K_e = 9$, $K_c = 201$, and various values of ρ.

any Pareto efficient equilibrium, there is a unique long run distribution over the endowment and consumption (Kocherlakota, 1996; Ligon, Thomas, and Worrall, 2002). We used our computed solution to forward simulate efficient equilibrium behavior in order to estimate this long run distribution. The blue curve in Figure 19 presents the average long run payoffs as a function of ρ. These numbers can equivalently be interpreted as the steady-state average payoff in a large economy of agents that are engaged in constrained-efficient bilateral risk sharing arrangements. The figure indicates that when ρ is close to zero, it is possible to support efficient risk sharing in which $c = 0.5$ in every period. Thus, players obtain the efficient surplus of $\sqrt{0.5} \approx 0.705$. As ρ increases, this average payoff declines until risk sharing breaks down altogether for $\rho > 19$.[20] We note that this breakdown occurs somewhat abruptly at high ρ due to the finite consumption grid and discontinuous changes in the equilibrium payoff sets when particular discrete transfers can no longer be incentivized. In contrast to long run constrained-efficient payoffs, autarky payoffs are non-monotonic in ρ. This is presumably due to the tradeoff between greater persistence around moderate endowments, which is welfare improving on average, versus persistence at the extremes, which is welfare decreasing.

[20]The current implementation of our algorithm simply stops when there are no admissible directions. This may happen because (a) the equilibrium payoff correspondence $\mathbf{V}$ has a single element or (b) there are no pure strategy subgame perfect Nash equilibria, so that $\mathbf{V}$ is empty. In this case, we know that the autarkic equilibrium always exists. Thus, for $\rho > 19$, the efficient and autarky curves would coincide.

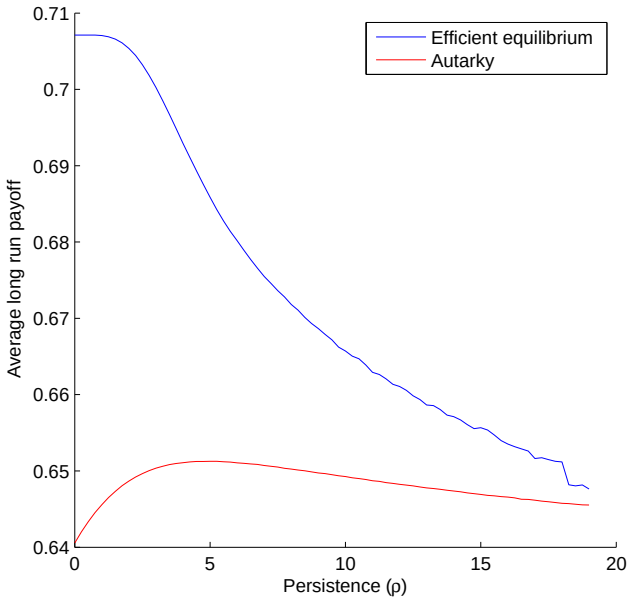


Figure 19: Long run payoffs for $\delta = 0.85$, $K_e = 9$, $K_c = 201$, and various values of ρ.

6.2 Computational efficiency

We conclude this section with a systematic assessment of the efficiency of our algorithm. For this purpose, it is useful to compare the pencil sharpening algorithm to existing methods. Because the APS algorithm uses all extreme points of the correspondence $\mathbf{W}$ to calculate the new correspondence $B(\mathbf{W})$, the number of extreme points along the APS sequence could grow very quickly. In particular, the number of extreme points of $B(\mathbf{W})(s)$ could in principle be as large as $|\mathbf{A}(s)||\text{ext}(\mathbf{W})|$. In contrast, whenever the monotonicity constraint is not binding, pencil sharpening takes a bounded amount of memory and time to complete one revolution (proportional to the bound on the number of basic tuples).

Judd, Yeltekin, and Conklin (2003), hereafter JYC, proposed an alternative procedure that also has bounded memory and time requirements per iteration, at the cost of only approximating the equilibrium correspondence.[21] Fix a finite set of directions

$$D = \{d_1, \ldots, d_{K_d}\}.$$

For a given set $W \subset \mathbb{R}^2$, we can define its *outer approximation* to be the smallest convex set with edges parallel to the d_j that contains W. Let $\widehat{d}_j$ be the counter-clockwise normal to d_j.

[21] JYC's original article concerned non-stochastic games, but the approach generalizes to stochastic games in a natural manner.

			Run times (h:m:s)							
			$K_d = 100$		$K_d = 200$		$K_d = 400$		$K_d = 600$	
K_e	K_c	#BP	JYC	ABS	JYC	ABS	JYC	ABS	JYC	ABS
3	31	126	1:21.8	1.2						
3	51	203	2:21.2	3.1	7:46.2	3.1				
3	101	395	5:21.9	11.6	16:42.6	11.7	56:50.3	11.5		
5	101	637	13:43	50.7	43:53.2	50.1	2:29:13.1	51.2	5:17:25.5	50.3

Table 2: Run times for various specifications of the risk sharing model and algorithms, in hours:minutes:seconds. #BP denotes the number of basic pairs on the final revolution of pencil sharpening. The convergence criterion for JYC is that distances between sets are less than 10^{-6}, and the convergence criterion for pencil sharpening (ABS) is that the approximation is contained within the final JYC set.

Letting

$$b_j = \max\left\{\widehat{d}_j \cdot w | w \in W\right\},$$

then the outer approximation of a set W (with respect to D) is defined as

$$\widehat{W} = \left\{w | \widehat{d}_j \cdot w \leq b_j \ \forall j = 1, \ldots, K_d\right\}.$$

The definition is extended to correspondences in the natural manner: $\widehat{\mathbf{W}}(s) = \widehat{\mathbf{W}(s)}$.

JYC propose an operator that, starting with a given payoff correspondence, produces a new correspondence via $\widehat{B}(\mathbf{W}) = \widehat{B(\widehat{\mathbf{W}})}$, i.e., the outer approximation of the set which is generated by applying the APS operator to an outer approximation. Since the APS operator and the outer approximation operator are both monotonic, $\widehat{B}$ will be monotonic as well. This implies that if $\mathbf{W}^0$ contains $\mathbf{V}$, then so does $\widehat{B}^k(\mathbf{W}^0)$. By taking a rich set of gradients, each $\widehat{W}$ converges to W, so one hopes that the limit of this iteration will closely approximate $\mathbf{V}$. Finally, the implementation of $\widehat{B}$ can be reduced to solving a large number of linear programming problems, one for each $s \in S$, $a \in A(s)$, and $d \in D$, each of which is relatively tractable.

For purposes of comparison, we implemented our own version of the JYC algorithm within our C++ libary. To solve the linear programs, we used the high-performance large-scale linear programming package Gurobi. In our implementation, we also exploited some common structure in the linear programs for different j to better streamline the computation.

In Table 2, we report run times for the pencil sharpening algorithm and the JYC algorithm on the risk-sharing example with $\delta = 0.85$, $\rho = 0$, and various values of K_e and K_c. In these trials, we first ran the JYC algorithm until the distance between successive approximations was less than 10^{-6}. The pencil sharpening algorithm was then run until its approximation was contained in the final JYC set. Thus, the interpretation of the numbers in the "ABS" column is the time for pencil sharpening to provide a better approximation than JYC. Also, from the output of our algorithm, we know the number of maximal tuples of $\mathbf{V}$ that are visited by the pencil sharpening algorithm which, for generic games, will be quite close to the number of extreme points.[22] To obtain a comparable level of accuracy, we configured the JYC algorithm with $K_d \approx$ the number of basic pairs on the final revolution of pencil sharpening, so that both algorithms could in principle generate the same level of detail in their final approximations.

The result are striking. For example, pencil sharpening generates 395 basic pairs on the last revolution when $K_e = 3$ and $K_c = 101$, and it takes 11.5 seconds to converge. In contrast, JYC with 400 gradients takes approximately 56 minutes and 43 seconds, which is about 300 times longer. Naturally, all of these numbers should be taken with a grain of salt: the run time will depend greatly on how the algorithm is implemented, and we have no doubt that it is possible to refine the implementations of both our own and JYC's algorithm to reduce the computation time. Nonetheless, the results are strongly suggestive that our algorithm is significantly faster than existing methods while providing an even greater level of accuracy.

7 Conclusion

It has been the purpose of this paper to study the subgame perfect equilibria of stochastic games. We have argued that the equilibria that generate payoffs that maximize in a common direction also have a common recursive structure. In particular, if incentive constraints are slack in the first period of the maximal equilibrium for a given state, then the continuation equilibria must generate payoffs that are maximal in the same direction, regardless of which

[22] There are two caveats to make here. First, it is a generic possibility that the pencil sharpening algorithm may take two steps in the same direction; this could occur when the next extreme payoff is generated using a non-binding action a, but the current pivot $\mathbf{v}$ is not incentive compatible for a. In this case, the algorithm may first introduce a into the basic pair with a binding payoff, and then move in the same direction by introducing a with a non-binding regime. We regard this as a somewhat exceptional case. Second, on the risk sharing example, the algorithm generates a large number of non-extreme pivots that lie on the "flats" at the southern and western edges of the equilibrium payoff set. This is due to the highly non-generic payoff structure, which causes payoffs generated with binding continuation values to lie along the same line. For a more generic game, each action would generate binding payoffs along different lines. In contrast, the payoffs that are generated along the northeastern frontier are all extreme points.

new state is reached. In contrast, the direction may change when incentive constraints bind. In this case, however, there is a sparse set of possible continuation equilibrium payoffs which may be used, namely one of the at most four extreme binding continuation payoffs.

This structure allows us to describe the generation of maximal equilibrium payoff tuples in terms of a basic pair, which consists of the actions played in the first period for each of the maximal equilibria, as well as a continuation regime that says to keep the direction constant if incentive constraints are slack or, if an incentive constraint binds, which binding continuation value to use. As an ancillary benefit, the fact that the number of basic pairs is bounded implies that the number of extremal subgame perfect Nash equilibrium payoffs is bounded as well.

Aside from their independent theoretical interest, these results are the basic building blocks of our pencil sharpening algorithm for computing the equilibrium payoff correspondence. This procedure generates a sequence of "pivots" using approximate basic pairs. We show that it is possible to incrementally modify the basic pair so that the pivot moves along a trajectory that circles around and asymptotically traces frontier of the equilibrium payoff correspondence. As a part of this process, our algorithm uncovers the basic pairs (and hence the equilibrium structures) that generate extremal equilibrium payoffs.

Finally, we have provided a user-friendly implementation of our method that can be used by other researchers. The ability to study rich but concrete examples can be useful for academics and students alike, who want to develop a better intuitive understanding of behavior in a particular stochastic game.

In closing, our work has three distinct components:

(i) We uncover key structural properties of extremal equilibria for a particular class of games.

(ii) We develop algorithms that exploit these properties.

(iii) We implement the algorithms in an accessible format for use by the research community.

AS undertook this research program for two player repeated games with perfect monitoring, and we have done the same for the considerably more complex class of stochastic games. The insights that we have used are obviously particular to the two player and perfect monitoring setting. Of course, these are fundamental classes of games both for theory and application and eminently worthy of study. Nonetheless, it is our hope that similar efforts will bear fruit for other classes of games, for example, those involving imperfect monitoring or more than two players.

References

Abreu, D., D. Pearce, and E. 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.

Abreu, D. and Y. Sannikov (2014): "An algorithm for two-player repeated games with perfect monitoring," *Theoretical Economics*, 9, 313–338.

Atkeson, A. (1991): "International lending with moral hazard and risk of repudiation," *Econometrica: Journal of the Econometric Society*, 1069–1089.

Blackwell, D. (1965): "Discounted dynamic programming," *The Annals of Mathematical Statistics*, 226–235.

Dixit, A., G. M. Grossman, and F. Gul (2000): "The dynamics of political compromise," *Journal of political economy*, 108, 531–568.

Ericson, R. and A. Pakes (1995): "Markov-perfect industry dynamics: A framework for empirical work," *The Review of Economic Studies*, 62, 53–82.

Hörner, J., T. Sugaya, S. Takahashi, and N. Vieille (2011): "Recursive methods in discounted stochastic games: An algorithm for $\delta \to 1$ and a folk theorem," *Econometrica*, 79, 1277–1318.

Judd, K. L., S. Yeltekin, and J. Conklin (2003): "Computing supergame equilibria," *Econometrica*, 71, 1239–1254.

Kocherlakota, N. R. (1996): "Implications of efficient risk sharing without commitment," *The Review of Economic Studies*, 63, 595–609.

Ligon, E., J. P. Thomas, and T. Worrall (2000): "Mutual insurance, individual savings, and limited commitment," *Review of Economic Dynamics*, 3, 216–246.

——— (2002): "Informal insurance arrangements with limited commitment: Theory and evidence from village economies," *The Review of Economic Studies*, 69, 209–244.

Ljungqvist, L. and T. J. Sargent (2004): *Recursive macroeconomic theory*, MIT press.

Mailath, G. J. and L. Samuelson (2006): "Repeated games and reputations: long-run relationships," *OUP Catalogue*.

Phelan, C. and E. Stacchetti (2001): "Sequential equilibria in a Ramsey tax model," *Econometrica*, 69, 1491–1518.

A Omitted proofs

A.1 Containment

Recall that the direction d is admissible at iteration k and for action a in state s if there exists a $v \in Gen(a)$ such that $xd = v - \mathbf{v}^k(s)$ for some $x > 0$. Before proving Lemma 4, we prove the following intermediate result:

Lemma 8 (Admissible regular test directions)**.** *For all iterations k, there exists a $s \in S^T$ such that at least one of $d^{NB}(s)$ or $d^B(s)$ is admissible and shallower than the shallowest tangent.*

Proof of Lemma 8. The existence of the admissible regular test direction is shown by considering four cases:

Case 1: There exists an $s \in S^{\mathrm{T}}$ such that $\mathbf{a}^{\mathrm{T}}(s)$ generates an admissible non-binding direction $d^{\mathrm{NB}}(s)$. Let $\mathbf{v}' \in \mathbf{V}\left(\underline{\mathbf{w}}^k\right)$ denote the vector of equilibrium continuation values that generate $\mathbf{w}^{\mathrm{T}}(s)$, i.e.,

$$\mathbf{w}^{\mathrm{T}}(s) = \sum_{s' \in S} \pi\left(s' | \mathbf{a}^{\mathrm{T}}(s)\right) \mathbf{v}'(s').$$

We claim that for all s, $d^{\mathrm{T}} \cdot \mathbf{v}^k(s) \geq d^{\mathrm{T}} \cdot \mathbf{v}'(s)$. This is immediate if $d(s) \propto d^{\mathrm{T}}$ from the fact that the tangent is a supporting hyperplane of $\mathbf{V}\left(\underline{\mathbf{w}}^k\right)(s)$ through $\mathbf{v}^k(s)$. Otherwise, we must have that $\widehat{d^{\mathrm{T}}} \cdot d(s) < 0$, so that $\widehat{d^k} \cdot d(s) < 0$ as well. Hence, d^k and $d(s)$ form a basis for $\mathbb{R}^2$, and we can decompose $d^{\mathrm{T}} = \alpha d^k + \beta d(s)$. These coefficients must be non-negative, since:

$$\underbrace{d^{\mathrm{T}} \cdot \widehat{d}(s)}_{\geq 0} = \alpha \underbrace{d^k \cdot \widehat{d}(s)}_{\geq 0}$$
$$\underbrace{d^{\mathrm{T}} \cdot \widehat{d^k}}_{\leq 0} = \beta \underbrace{d(s) \cdot \widehat{d^k}}_{\leq 0}$$

For all $w \in \mathbf{V}\left(\underline{\mathbf{w}}^k\right)(s)$, $\widehat{d}^k \cdot w \leq \widehat{d}^k \cdot \mathbf{v}^k(s')$ by the inductive hypothesis of containment (and the definition of $\mathbf{W}^k$), and $\widehat{d}(s) \cdot w \leq \widehat{d}(s) \cdot \mathbf{v}^k(s)$ from the definition of the tangent. Thus,

$$\begin{aligned}\widehat{d}^{\mathrm{T}} \cdot w &= \alpha \widehat{d}^k \cdot w + \beta \widehat{d}(s) \cdot w \\ &= \alpha \widehat{d}^k \cdot \mathbf{v}^k(s) + \beta \widehat{d}(s) \cdot \mathbf{v}^k(s) \\ &= \widehat{d}^{\mathrm{T}} \cdot \mathbf{v}^k(s).\end{aligned}$$

Thus, $\widehat{d}^{\mathrm{T}} \cdot \mathbf{w}^{\mathrm{T}}(s) \leq \widehat{d}^{\mathrm{T}} \cdot \widetilde{\mathbf{w}}(s)$, and

$$\begin{aligned}\widehat{d}^{\mathrm{T}} \cdot d^{\mathrm{NB}}(s) &= (1-\delta)\widehat{d}^{\mathrm{T}} \cdot g\left(\mathbf{a}^{\mathrm{T}}(s)\right) + \delta \widehat{d}^{\mathrm{T}} \cdot \widetilde{\mathbf{w}}(s) - \widehat{d}^{\mathrm{T}} \cdot \mathbf{v}^k(s) \\ &\geq (1-\delta)\widehat{d}^{\mathrm{T}} \cdot g\left(\mathbf{a}^{\mathrm{T}}(s)\right) + \delta \widehat{d}^{\mathrm{T}} \cdot \mathbf{w}^{\mathrm{T}}(s) - \widehat{d}^{\mathrm{T}} \cdot \mathbf{v}^k(s) \\ &= \widehat{d}^{\mathrm{T}} \cdot d^{\mathrm{T}} = 0,\end{aligned}$$

so that $d^{\mathrm{NB}}(s)$ is shallower than d^{T}.

Case 2: There exists an $s \in S^{\mathrm{T}}$ such that $d^{\mathrm{NB}}(s)$ is inadmissible, but $d^{\mathrm{NB}}(s) \neq 0$ and $d^{\mathrm{B}}(s) \neq 0$. Let

$$\overline{V}(s) = \sum_{s' \in S} \pi\left(s' | \mathbf{a}^{\mathrm{T}}(s)\right) \mathbf{V}\left(\underline{\mathbf{w}}^k\right)(s')$$

denote the expected partial equilibrium payoffs with the fixed threat tuple. For the purpose of this proof, we adopt the shorthand $\overline{W}(s) = \overline{W}\left(\mathbf{a}^{\mathrm{T}}(s)\right)$, $IC(s) = IC\left(\mathbf{a}^{\mathrm{T}}(s)\right)$, $C(s) = C\left(\mathbf{a}^{\mathrm{T}}(s)\right)$, and $Gen(s) = Gen\left(\mathbf{a}^{\mathrm{T}}(s)\right)$. By the inductive hypothesis, $\overline{V}(s) \subseteq \overline{W}(s)$. Note that $\mathbf{w}^{\mathrm{T}}(s)$ is incentive compatible, so $\mathbf{w}^{\mathrm{T}}(s) \in \overline{V} \cap IC(s) \subseteq \overline{W}(s) \cap IC(s)$. Let $w(t) = t\widetilde{\mathbf{w}}(s) + (1-t)\mathbf{w}^{\mathrm{T}}(s)$. Since $IC(s)$ is closed, there exists a $\widehat{t} \in (0,1)$ such that $w(t) \in IC(s)$ only if $t \geq \widehat{t}$. Thus, $w(\widehat{t}) \in \mathrm{bd}IC(s)$. Moreover, by the inductive hypothesis of feasibility, $\widetilde{\mathbf{w}}(s) \in \overline{W}(s)$, so that $w\left(\widehat{t}\right) \in \mathrm{bd}IC(s) \cap \overline{W}(s)$. Since $\mathbf{w}^*(s)$ is d^{T}-maximal in $\mathrm{bd}IC(s) \cap \overline{W}(s)$, it must be that

$$\begin{aligned}\widehat{d}^{\mathrm{T}} \cdot \mathbf{w}^*(s) &\geq \widehat{d}^{\mathrm{T}} \cdot w(\widehat{t}) \\ &= t\widehat{d}^{\mathrm{T}} \cdot \widetilde{\mathbf{w}}(s) + (1-t)\widehat{d}^{\mathrm{T}} \cdot \mathbf{w}^{\mathrm{T}}(s) \\ &\geq \widehat{d}^{\mathrm{T}} \cdot \mathbf{w}^{\mathrm{T}}(s),\end{aligned}$$

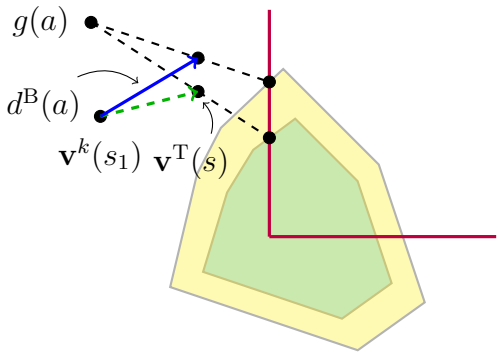


Figure 20: Case 3, in which $\mathbf{v}^{\mathrm{T}}(s)$ is generated with a binding constraint for some $s \in S^{\mathrm{T}}$, and the best binding direction is non-zero.

since as shown in Case 1, $\widehat{d}^{\mathrm{T}} \cdot \widetilde{\mathbf{w}}(s) \geq \widehat{d}^{\mathrm{T}} \cdot \mathbf{w}^{\mathrm{T}}(s)$. Thus,

$$\begin{aligned}\widehat{d}^{\mathrm{T}} \cdot d^{\mathrm{B}}(s) &= \widehat{d}^{\mathrm{T}} \cdot \left((1-\delta) g\left(\mathbf{a}^{\mathrm{T}}(s)\right) + \delta \mathbf{w}^*(s) - \mathbf{v}^k(s)\right) \\ &= \widehat{d}^{\mathrm{T}} \cdot d^{\mathrm{T}} + \delta d^{\mathrm{T}} \cdot (\mathbf{w}^*(s) - \mathbf{w}^{\mathrm{T}}(s)) \\ &= \delta d^{\mathrm{T}} \cdot (\mathbf{w}^*(s) - \mathbf{w}^{\mathrm{T}}(s)) \geq 0,\end{aligned}$$

since $\mathbf{v}^{\mathrm{T}}(s) = (1-\delta) g\left(\mathbf{a}^{\mathrm{T}}(s)\right) + \delta \mathbf{w}^{\mathrm{T}}(s) - \mathbf{v}^k(s)$, and therefore $d^{\mathrm{B}}(s)$ is shallower than d^{T}.

Case 3: There exists $s \in S^{\mathrm{T}}$ such that $d^{\mathrm{NB}}(s)$ is inadmissible and $\mathbf{v}^{\mathrm{T}}(s)$ is generated using a binding constraint. In that case, $\mathbf{w}^{\mathrm{T}}(s) \in \mathrm{bd} IC(s) \cap \overline{V}(s) \subseteq \mathrm{bd} IC(s) \cap \overline{W}(s)$, so that $\widehat{d}^{\mathrm{T}} \cdot \mathbf{w}^*(s) \geq \widehat{d}^{\mathrm{T}} \cdot \mathbf{w}^{\mathrm{T}}(s)$, and therefore:

$$\begin{aligned}\widehat{d}^{\mathrm{T}} \cdot d^{B}(s) = \widehat{d}^{\mathrm{T}} &\cdot (\mathbf{v}^{\mathrm{T}}(s) - \mathbf{v}^k(s)) \\ &+ \delta \widehat{d}^{\mathrm{T}} \cdot (\mathbf{w}^*(s) - \mathbf{w}^{\mathrm{T}}(s)) \geq 0.\end{aligned}$$

If, in addition, $d^{\mathrm{B}} \neq 0$, we are done. If $d^{B} = 0$, then we conclude that $\alpha(s) d^{\mathrm{T}} = \delta(\mathbf{w}^{\mathrm{T}}(s) - \mathbf{w}^*(s)) \neq 0$. But both $\mathbf{w}^*(s), \mathbf{w}^{\mathrm{T}}(s) \in \mathrm{bd} IC(s) \cap \overline{W}(s)$ contradicts that $\mathbf{w}^*(s)$ is the clockwise $\widehat{d}^{\mathrm{T}}$-maximal continuation value (both continuation values lie on the same d^{T} plane, but $\mathbf{w}^{\mathrm{T}}(s)$ is further in the d^{T} direction.

Case 4: For all $s \in S^{\mathrm{T}}$, □

(i) $d^{\mathrm{NB}}(s)$ is either not IC or $d^{\mathrm{NB}}(s) = 0$ (and therefore inadmissible).

(ii) $\mathbf{v}\,(s)$ is generated without a binding constraint.

(iii) Either $d^{\mathrm{NB}}(s) = 0$ or $d^{\mathrm{B}}(s) = 0$.

Proof. Case 4 describes all situations not covered in Cases 1-3. In fact, we will show by contradiction that Case 4 cannot occur.

We argue that properties (ii) and (iii) imply that for all states s that can be reached with positive probability from $s' \in S^{\mathrm{T}}$ using $\mathbf{a}^{\mathrm{T}}(s')$, $\mathbf{v}^{\mathrm{T}}(s) - \mathbf{v}^{k}(s)$ is proportional to d^{T}. In particular, $\forall s \in \cup_{s' \in S^{\mathrm{T}}} \mathrm{supp}\pi(\cdot|\mathbf{a}^{\mathrm{T}}(s'))$, we must be able to find $\mathbf{x}(s) \in \mathbb{R}$ such that

$$\mathbf{v}^{\mathrm{T}}(s) - \mathbf{v}^{k}(s) = \mathbf{x}(s) d^{\mathrm{T}}. \tag{6}$$

Note that this is obviously the case for $s \in S^{\mathrm{T}}$ with $\mathbf{x}(s) > 0$.

Before showing that this is the case, let us see why it leads to a contradiction. First, note that property (ii) implies that

$$\mathbf{v}^{\mathrm{T}}(s) = (1-\delta) g\left(\mathbf{a}^{\mathrm{T}}(s)\right) + \delta \sum_{s' \in S} \pi\left(s'|\mathbf{a}^{\mathrm{T}}(s)\right) \mathbf{v}^{\mathrm{T}}(s') \tag{7}$$

so that

$$\begin{aligned} d^{\mathrm{NB}}(s) &= (1-\delta) g\left(\mathbf{a}^{\mathrm{T}}(s)\right) + \delta \widetilde{\mathbf{w}}(s) - \mathbf{v}_k(s) \\ &\quad + \mathbf{v}^{\mathrm{T}}(s) - \mathbf{v}^{\mathrm{T}}(s) \\ &= \mathbf{v}^{\mathrm{T}}(s) - \mathbf{v}_k(s) - \delta \sum_{s' \in S} \pi\left(s'|\mathbf{a}^{\mathrm{T}}(s)\right) \left(\mathbf{v}^{\mathrm{T}}(s') - \mathbf{v}^{k}(s')\right) \\ &= \left(\mathbf{x}(s) - \delta \sum_{s' \in S} \pi\left(s'|\mathbf{a}^{\mathrm{T}}(s)\right) \mathbf{x}(s')\right) d^{\mathrm{T}} \end{aligned}$$

Letting $s^* \in \arg\max_{s \in S^{\mathrm{T}}} \mathbf{x}(s)$, we have that $d^{\mathrm{NB}}(s^*) = \gamma d^{\mathrm{T}}$ with

$$\gamma = \mathbf{x}(s^*) - \delta \sum_{s' \in S} \pi\left(s'|\mathbf{a}^{\mathrm{T}}(s^*)\right) \mathbf{x}(s') \geq \mathbf{x}(s^*)(1-\delta) > 0$$

so that $d^{\mathrm{NB}}(s^*)$ is non-zero and is a positive scaling of d^{T}.

Thus, property (iii) implies that $d^{\mathrm{B}}(s^*) = 0$, so that

$$d^{\mathrm{NB}}(s^*) = \gamma d^{\mathrm{T}} = \delta\left(\widetilde{\mathbf{w}}(s^*) - \mathbf{w}^*(s^*)\right)$$

and also

$$\mathbf{x}(s^*) d^{\mathrm{T}} = \delta(\mathbf{w}^{\mathrm{T}}(s^*) - \mathbf{w}^*(s^*))$$

Moreover, by property (i), it must be that $\widetilde{\mathbf{w}}(s^*)$ is not incentive compatible, even though $\mathbf{w}^{\mathrm{T}}(s)$ is, so that there is a convex combination of the two

$$w = (1-\mu)\widetilde{\mathbf{w}}(s^*) + \mu \mathbf{w}^{\mathrm{T}}(s^*)$$

which is in $\mathrm{bd}IC(s^*)$, with $\mu \in (0,1)$. Hence,

$$\delta(w - \mathbf{w}^*(s^*)) = ((1-\mu)\mathbf{x}(s^*) + \mu\gamma)d^{\mathrm{T}}$$

where the coefficient $(1-\mu)\mathbf{x}(s^*) + \mu\gamma > 0$, which contradicts that $\mathbf{w}^*(s^*)$ is clockwise $\widehat{d}^{\mathrm{T}}$-maximal in $\mathrm{bd}IC(s^*) \cap \overline{W}(s^*)$.

Finally, let us prove the existence of $\mathbf{x}(s)$ such that (6) holds. Certainly, the $\mathbf{x}(s)$ exist and are strictly positive for $s \in S^{\mathrm{T}}$. If $d^{\mathrm{NB}}(s) = 0$, we can write

$$\mathbf{v}^k(s) = (1-\delta)g\left(\mathbf{a}^{\mathrm{T}}(s)\right) + \delta \sum_{s' \in S} \pi\left(s'|\mathbf{a}^{\mathrm{T}}(s)\right)\mathbf{v}^k(s').$$

Combined with (7), we obtain

$$\mathbf{x}(s)d^{\mathrm{T}} = \delta \sum_{s' \in S} \pi\left(s'|\mathbf{a}^{\mathrm{T}}(s)\right)\left(\mathbf{v}^{\mathrm{T}}(s') - \mathbf{v}^k(s')\right).$$

Thus,

$$\widehat{d}^{\mathrm{T}} \cdot \delta \sum_{s' \in S} \pi\left(s'|\mathbf{a}^{\mathrm{T}}(s)\right)\left(\mathbf{v}^{\mathrm{T}}(s') - \mathbf{v}^k(s')\right) = 0.$$

If $\widehat{d}^{\mathrm{T}} \cdot (\mathbf{v}^{\mathrm{T}}(s') - \mathbf{v}^k(s')) \neq 0$ for some $s' \in \mathrm{supp}\pi(\cdot|\mathbf{a}^{\mathrm{T}}(s))$, then there must exist some s' for which $\widehat{d}^{\mathrm{T}} \cdot (\mathbf{v}^{\mathrm{T}}(s') - \mathbf{v}^k(s')) > 0$, which contradicts that d^{T} is the shallowest tangent.

Now suppose instead that $d^{\mathrm{NB}}(s) \neq 0$ but $d^{\mathrm{B}}(s) = 0$, so that $\widehat{d}^{\mathrm{T}} \cdot (\mathbf{w}^{\mathrm{T}}(s) - \mathbf{w}^*(s)) = 0$. We have already shown that $\widehat{d}^{\mathrm{T}} \cdot \widetilde{\mathbf{w}}(s) \geq \widehat{d}^{\mathrm{T}} \cdot \mathbf{w}^{\mathrm{T}}(s)$. If $\widehat{d}^{\mathrm{T}} \cdot \widetilde{\mathbf{w}}(s) > \widehat{d}^{\mathrm{T}} \cdot \mathbf{w}^{\mathrm{T}}(s)$, then since $\mathbf{w}^{\mathrm{T}}(s) \in \overline{W}(s) \cap \mathrm{int}IC(s)$ (since IC constraints are not binding), there exists a $t \in (0,1)$ such that $w = t\widetilde{\mathbf{w}}(s) + (1-t)\mathbf{w}^{\mathrm{T}}(s) \in \mathrm{bd}IC(s)$, so that $\widehat{d}^{\mathrm{T}} \cdot w > \widehat{d}^{\mathrm{T}} \cdot \mathbf{w}^*(s)$, contradicting the fact that $\mathbf{w}^*(s)$ is $\widehat{d}^{\mathrm{T}}$-maximal in $\mathrm{bd}IC(s) \cap \overline{W}(s)$. Thus, $\widehat{d}^{\mathrm{T}} \cdot (\mathbf{w}^{\mathrm{T}}(s) - \widetilde{\mathbf{w}}(s)) = 0$, so by the argument of the previous paragraph, we conclude that $\mathbf{v}^{\mathrm{T}}(s') - \mathbf{v}_k(s'') \propto d^{\mathrm{T}}$ for all s' such that $\pi\left(s'|\mathbf{a}^{\mathrm{T}}(s)\right) > 0$ for some $s \in S^{\mathrm{T}}$. □

Proof of Lemmas 3 and 4. By Lemma 8, there exists a non-zero and *Gen* regular test direction for some action $a = \mathbf{a}^{\mathrm{T}}(s)$ with $s \in S^{\mathrm{T}}$ that is shallower than the shallowest tangent. If this direction is also monotonic, then it will be admissible, and therefore the best direction

will be weakly shallower and containment will be satisfied. If, on the other hand, this direction is non-monotonic, then we argue that either a frontier or interior direction must be admissible.

To see that this is the case, first note that both $\mathbf{v}^{\mathrm{T}}(s)$ and the payoff v that generates the non-monotonic direction d are in $Gen(a)$. Since $Gen(a)$ is convex, all payoffs between $\mathbf{v}^{\mathrm{T}}(s)$ and v are in $Gen(a)$ as well. Thus, the payoff

$$v(t) = tv + (1-t)\mathbf{v}^{\mathrm{T}}(s)$$

can be generated for all $t \in [0,1]$, and it is possible to generate directions

$$d(t) = v(t) - \mathbf{v}^k(s) = td + (1-t)d^{\mathrm{T}}.$$

Moreover, since d is non-monotonic but d^{T} is monotonic, there must be a largest t such that $\mathbf{v}^k(s) + d(t) \in \mathbf{W}^k(s)$. Moreover, it must be that $d \cdot \widehat{d^{\mathrm{T}}} \geq 0$, so that the frontier direction $d(t)$ satisfies $d(t) \cdot \widehat{d^{\mathrm{T}}} \geq 0$ as well. If this inequality is strict, then it must be that $v(t) \neq \mathbf{v}^k(s)$, so that an admissible and monotonic frontier direction exists which is shallower than d^{T}, and we are done. Otherwise, it must be that $d \propto -d^{\mathrm{T}}$, in which case the shallowest tangent is itself an admissible interior direction. □

A.2 No stalling

The argument for this result will not rely on the particulars of our algorithm, but only on the fact that the pencil sharpening algorithm generates a trajectory that (1) is monotonically moving in the clockwise direction and (2) contains the non-empty convex sets $\mathbf{V}\left(\underline{\mathbf{w}}^k\right)(s)$. These properties are formalized as:

1. $\mathbf{v}^l(s) = \mathbf{v}^{l-1}(s) + \mathbf{x}^l(s)d^l$ where $d^l \cdot \widehat{d}^{l-1} \leq 0$ and $\mathbf{x}^l(s) \in \mathbb{R}_+$.
2. $v \cdot \widehat{d}^l \leq \mathbf{v}^l(s) \cdot \widehat{d}^l$ for all $v \in \mathbf{V}\left(\underline{\mathbf{w}}^k\right)(s)$.

We will refer to such a sequence $\left\{\mathbf{v}^l, d^l\right\}$ as an *orbital trajectory*. The local containment argument proves that the algorithm generates an orbital trajectory.

We will say that the orbital trajectory $\left\{\mathbf{v}^k, d^k\right\}$ has the *rotation property* if for all directions d and for all k, there exists an $l \geq k$ such that $d^l \cdot \widehat{d} \leq 0$. In words, this property says that for every direction, the sequence will point weakly below that direction infinitely many times.

We will argue that the trajectory generated by the twist algorithm satisfies the rotation property, which in turn implies that the algorithm completes infinitely many revolutions. The

proof is lengthy, but the intuition is quite straightforwrd. If the pivot ever got stuck in a way that prevented it from completing orbits of $\mathbf{V}$, then eventually the direction would move around and cut some payoffs in $\mathbf{V}$ out of the trajectory, which contradicts local containment.

Lemma 9. *If the orbital trajectory* $\left\{\mathbf{v}^k, d^k\right\}$ *satisfies the rotation property,* $\lim_{k\to\infty} r(k) = \infty$.

Proof of Lemma 9. We will show that from any iteration k, there exists an iteration $k' > k$ that starts a revolution, which implies the result.

First, for any k, there must exist a $k' > k$ such that $d^{k'} \cdot \widehat{d}^k < 0$. The rotation property would clearly fail for the direction $-\widehat{d}^k$ if $d^l \propto d^k$ for all $l > k$, and if we only have $d^l \propto d^k$ or $d^l \propto -d^k$, then containment would be violated under the hypothesis that $\mathbf{V}$ has full dimension.

Now let us consider two cases. If $d^k \propto d^N$, then there exists a smallest $k' > k$ such that $d^{k'} \cdot \widehat{d}^k = d^{k'} \cdot \widehat{d}^N < 0$, which in fact must start a revolution.

Otherwise, if $d^k \cdot \widehat{d}^N \geq 0$, there is a smallest $k_1 \geq k$ such that $d^{k_1} \cdot \widehat{d}^N > 0$. There is then a smallest $k_2 \geq k_1$ such that $d^{k_2} \cdot \widehat{d}^N \leq 0$, and finally a smallest $k_3 \geq k_2$ such that $d^{k_3} \cdot \widehat{d}^N < 0$. We claim that k_3 starts a revolution. If $d^{k_3-1} \cdot \widehat{d}^N > 0$, then this is obvious. Otherwise, we claim that $d^{k_3-1} \propto d^N$. For if $d^{k_3-1} \propto -d^N$, then $d^{k_3} \cdot \widehat{d}^{k_3-1} = -d^{k_3} \cdot \widehat{d}^N > 0$, a contradiction. □

Lemma 10. *If the rotation property fails, then there exists a direction* d^∞ *such that* $d^l/\|d^l\| \to d^\infty$*, and moreover* $d^l \cdot \widehat{d}^\infty \geq 0$ *for* l *sufficiently large.*

Proof of Lemma 10. Suppose that there exists a k and direction $\underline{d}$ such that $d^l \cdot \widehat{\underline{d}} > 0$ for all $l \geq k$. We can write each direction d^l as

$$d^l/\|d^l\| = x^l \underline{d} + y^l \widehat{\underline{d}}$$

for some coordinates x^l and y^l. Note that the hypothesis $d^l \cdot \widehat{\underline{d}} > 0$ implies that $y^l > 0$.

Claim: x^l is monotonically increasing in l. The best direction d^l must satisfy $d^l \cdot \widehat{d}^{l-1} \leq 0$, which implies that

$$\begin{aligned} d^l \cdot \widehat{d}^{l-1} &= (x^l \underline{d} + y^l \widehat{\underline{d}})(x^{l-1} + y^{l-1} \widehat{\underline{d}}) \\ &= (x^{l-1} y^l - x^l y^{l-1}) \|\underline{d}\|^2 \leq 0 \end{aligned}$$

so that

$$x^{l-1} y^l \leq x^l y^{l-1}.$$

Suppose that $x^l < x^{l-1}$. Then $y^l > y^{l-1}$ (since $(x^l)^2 + (y^l)^2 = 1$), so

$$x^l y^{l-1} < x^l y^l < x^{l-1} y^l$$

since $y^l > 0$, a contradiction. Thus, it must be that $x^l > x^{l-1}$. It must also be that $x^l \leq 1$, so that x^l converges to some x^∞. Finally, $y^l = \sqrt{1-(x^l)^2}$, so y^l converges to $y^\infty = \sqrt{1-(x^\infty)^2}$, and the limit direction is

$$d^\infty = x^\infty \underline{d} + y^\infty \widehat{\underline{d}}.$$

In the limit, $d^l \cdot \widehat{d^\infty}$ is proportional to $x^\infty y^l - x^l y^\infty$. Monotonicity implies that $x^\infty \geq x^l$ and x^∞ and x^l have the same sign. Thus, if $x^\infty > 0$, x^l must be positive so that $y^l \geq y^\infty$, so that $x^\infty y^l \geq x^l y^\infty$. If $x^l \leq x^\infty \leq 0$, then $y^l \leq y^\infty$, and again we conclude that $x^\infty y^l \geq x^l y^\infty$. □

Having established these general results about orbital trajectories, we can now return to the particulars of our algorithm and prove the anti-stalling lemma.

Proof of Lemma 6. Suppose that the trajectory generated by the algorithm does not complete infinitely many revolutions. Then from Lemma 9, we conclude that the rotation property fails, so that there exists a k and a $\underline{d}$ such that $d^l \cdot \widehat{\underline{d}} \geq 0$ for all $l \geq k$. We then conclude from Lemma 10 there exists a direction d^∞ such that $d^l/\|d^l\| \to d^\infty$. Moreover, there exists a k' such that for all $l \geq k'$, $d^l \cdot \widehat{d^\infty} \geq 0$ and $d^l \cdot d^\infty > 0$. We also note that there must exist a k' such that no new revolutions are completed after iteration k'. In particular, if d^∞ points west of due north, then eventually all of the d^l will point west of due north, so that it will be impossible for d^l to point east again and no new revolutions can be completed. The analysis would be symmetric if d^∞ points east. Thus, we can assume without loss of generality that the sets $\overline{W}(a)$, $IC(a)$, $Gen(a)$, $C(a)$, and the function $h(a)$ are not changing.

Because there are finitely many actions and states, there must be some action which generates infinitely many best test directions, and for l sufficiently large, we must have that $\mathbf{v}^l(s)$ is strictly increasing in the d^∞ direction. Now, there are only finitely many binding payoffs in the pivot $\mathbf{v}^k$ and only finitely many extreme continuation values in $C(a)$. As a result, there are only finitely many configurations of the system that defines $\mathbf{v}^l$ that use (i) binding payoffs that were in the original pivot $\mathbf{v}^k$ or (ii) extreme binding continuation values in $C(a)$. Thus, it is impossible that the algorithm generates infinitely many new pivots that are monotonically increasing in the d^∞ direction using only non-binding and binding payoffs.

There are therefore only three possibilities for generating new directions, by introducing new binding payoffs into the pivot: (i) new binding payoffs introduced when changing a

non-binding regime to a binding regime in state s during the pivot updating procedure, (ii) generating a new frontier test direction, or (iii) generating a new interior test direction.

Now let us consider two cases. In the first case, there exists a k such that for all $l \geq k$, $d^l/\|d^l\| = d^\infty$, i.e., d^l converges exactly in finitely many iterations. Now, it is not too hard to see that this is incompatible with a generating infinitely many non-zero movements according to (i), (ii), or (iii). For example, when (i) occurs, the pivot must travel as far as possible in the direction d^∞ while maintaining incentive compatibility. If $\mathbf{v}^l(s)$ were to travel any further in the direction d^∞ on a subsequent iteration using the same action, incentive compatibility would be violated, which rules out new pivots of the forms (i), (ii), or (iii) being generated with this action. Similarly, if a new pivot were generated according to (ii) or (iii), the pivot again moves as far as possible in the direction d^∞ subject to the continuation value being (a) incentive compatible, (b) feasible in the sense of $w \in \overline{W}(a)$, and (c) contained in $\mathbf{W}^k(s)$ (which contains $\mathbf{W}^l(s)$ for all $l > k$). Thus, any further movement in the direction d^∞ on a subsequent iteration must violate one of these requirements (a-c), and therefore is impossible.

In the second case, d^l approaches d^∞ in the limit but never converges exactly. Let

$$X(a) = Gen(a) \cap \mathbf{W}^k(s)$$

denote the payoffs in $\mathbf{W}^k$ that can be generated using a and feasible and incentive compatible continuation values in state s. The set $X(a)$ is a convex polytope in $\mathbb{R}^2$. Let M denote the set of directions which are the slopes of edges of $X(a)$. In other words, if E is an edge of $X(a)$, then

$$E \subseteq \overline{H}\,(v, m)$$

for some $v \in X(a)$ and $m \in M$, where

$$\overline{H}(v, m) = \{w | w \cdot \widehat{m} = v \cdot \widehat{m}\}$$

is the line through v with slope m. Since there are only finitely many pivots up to iteration k, $X(a)$ has finitely many extreme points, and therefore M is a finite set. Let k' be large enough so that (i) $d^{k'} \cdot \widehat{m} \neq 0$ for all $m \in M$, and (ii) $\text{sgn}\left(d^{k'} \cdot \widehat{m}\right) = \text{sgn}\left(d^l \cdot \widehat{m}\right)$ for all $l \geq k'$. This k' must exist beause d^l converges. For example, if $d^\infty \cdot \widehat{m} > 0$, then obviously there exists a k_m so that for all $l \geq k_m$, $d^l \cdot \widehat{m} > 0$ as well. This will symmetrically be true for $d^\infty \cdot \widehat{m} < 0$. If $d^\infty \cdot \widehat{m} = 0$, then we can establish the existence of k_m so that if $d^\infty \cdot m > 0$ (< 0), then there exists a k_m such that $d^l \cdot \widehat{m} > 0$ (< 0). For in the former case, $d^\infty = xm$

for some $x > 0$, so $d^l \cdot \widehat{m} = d^l \cdot x\widehat{d^\infty}$, which is strictly positive. In the other case, we use the fact that $d^\infty = -xm$ for some $x > 0$.

Now let

$$Y^l = \left\{ w | w \cdot \widehat{d^l} \leq \mathbf{v}^l(s) \cdot \widehat{d^l}, w \cdot \widehat{d^\infty} > \mathbf{v}^l(s) \cdot \widehat{d^\infty} \right\}$$

be the set of payoffs which could generate a new direction d^{l+1} such that $d^{l+1} \cdot \widehat{d^l} \leq 0$ and $d^{l+1} \cdot \widehat{d^\infty} > 0$. Note that $Y^l \subseteq Y^{l-1}$. Any payoff $w \in Y^l$ can be written as

$$\begin{aligned} w &= xd^l + yd^\infty; \\ \mathbf{v}^l(s) &= x^l d^l + y^l d^\infty; \\ \mathbf{v}^{l-1}(s) &= x^{l-1} d^l + y^{l-1} d^\infty. \end{aligned}$$

The fact that $w \cdot \widehat{d^\infty} > \mathbf{v}^l(s) \cdot \widehat{d^\infty}$ implies that $x > x^l$, since $d^l \cdot \widehat{d^\infty} > 0$. In turn, $\mathbf{v}^l(s) \in Y^{l-1}$ implies that $\mathbf{v}^l(s) \cdot \widehat{d^\infty} > \mathbf{v}^{l-1}(s) \cdot \widehat{d^\infty}$ and therefore $x^l \geq x^{l-1}$, which proves that $x \geq x^{l-1}$. On the other hand,

$$\begin{aligned} w \cdot \widehat{d^l}^{-1} &= xd^l \cdot \widehat{d^l}^{-1} + yd^\infty \cdot \widehat{d^l}^{-1} \\ &\leq x^l d^l \cdot \widehat{d^l}^{-1} + y^l d^\infty \cdot \widehat{d^l}^{-1} \end{aligned}$$

since $w \cdot \widehat{d^l} \leq \mathbf{v}^l(s) \cdot \widehat{d^l}$ implies that $y \leq y^l$, as $d^\infty \cdot \widehat{d^l} < 0$, and $d^l \cdot \widehat{d^l}^{-1} \leq 0$ as well. The latter implies that $\mathbf{v}^l(s) \cdot \widehat{d^l}^{-1} \leq \mathbf{v}^{l-1}(s) \cdot \widehat{d^l}^{-1}$, so that $w \cdot \widehat{d^l}^{-1} \leq \mathbf{v}^{l-1}(s) \cdot \widehat{d^l}^{-1}$.

Now, let $k'' \geq k'$ such that a generates the best direction according to (ii) or (iii). (The analysis for (i) is entirely analogous, with the incentive constraints replacing the half-space constraints that define $X(a)$.) This implies that $\mathbf{v}^{k''}(s)$ is on the boundary of $X(a)$, and in particular that $\mathbf{v}^{k''}(s)$ is on an edge E with slope m.

Claim: $Y^{k''} \cap X(a) = \emptyset$. Note that any $\mathbf{v}^l(s)$ for $l \geq k''$ must be contained in $Y^l \cap X(a)$, which is contained in $Y^{k''} \cap X(a)$. Thus, a consequence of this claim is that a cannot generate any more non-zero directions at iterations later than k'' as we had supposed.

Now, let us see why the claim is true. Note that $X(a) \subseteq H\left(\mathbf{v}^{k''}(s), m\right)$ for some $m \in M$, which is the slope of an edge that contains $X(a)$. If $\mathbf{v}^l(s)$ is not an extreme point, this m is unique, and we note that it must be the case that $d^{k''-1} \cdot \widehat{m} > 0$. Otherwise, traveling further in the direction $d^{k''-1}$ would move the pivot into the interior of $X(a)$, contradicting that we had moved as far as possible in the direction $d^{k''-1}$ without violating feasibility or incentive compatibility.

On the other hand, if $\mathbf{v}^{k''}(s)$ is an extreme point, there are two such m. We can distinguish these as m^1 and m^2, where m^2 is the slope of the clockwise edge and m^1 is the slope of the counter-clockwise edge. We claim that for at least one of these m, it must be that $d^{k''-1} \cdot \widehat{m} > 0$. Otherwise, the same claim applies as above. In particular, if $d^{k''-1} = xm^2$ or if $d^{k''-1} = -xm^1$ for some $x > 0$, then it is clearly possible to move along one of the edges. If $d^{k''-1} = -xm^2$ or if $d^{k''} = xm$, then because $m^2 \cdot \widehat{m}^1 < 0$, either $d^{k''-1} \cdot \widehat{m}^1 < 0$ or $d^{k''-1} \cdot \widehat{m}^2 < 0$. Finally, if $d^{k''-1} \cdot \widehat{m}^1 > 0$ and $d^{k''-1} \cdot \widehat{m}^2 > 0$, then by traveling in the direction $d^{k''-1}$, the pivot would move into the interior of $X(a)$.

Thus, we can find an m for which $X(a) \subseteq H\left(\mathbf{v}^{k''}(s), m\right)$ and $d^{k''-1} \cdot \widehat{m} < 0$. This implies that $d^l \cdot \widehat{m} > 0$ for all $l \geq k''$, and in particular, that $d^\infty \cdot \widehat{m} \geq 0$. It will be sufficient to show that for all $w \in Y^l$, $w \cdot \widehat{m} > \mathbf{v}^{k''}(s) \cdot \widehat{m}$. Let us write

$$\begin{aligned} w &= xd^{k''-1} + yd^\infty \\ \mathbf{v}^l(s) &= x^l d^{k''-1} + y^l d^\infty. \end{aligned}$$

Then

$$\begin{aligned} w \cdot \widehat{d^\infty} &= xd^{k''-1} \cdot \widehat{d^\infty} \\ &> x^{k''} d^{k''-1} \cdot \widehat{d^\infty} \\ &= \mathbf{v}^{k''}(s) \cdot \widehat{d^\infty}, \end{aligned}$$

which implies that $x > x^{k''}$, since $d^{k''-1} \cdot \widehat{d^\infty} > 0$. Similarly,

$$\begin{aligned} w \cdot \widehat{d^{k''-1}} &= yd^\infty \cdot \widehat{d^{k''-1}} \\ &\leq y^{k''} d^\infty \cdot \widehat{d^{k''-1}} \\ &= \mathbf{v}^{k''}(s) \cdot \widehat{d^{k''-1}}, \end{aligned}$$

which implies that $y \geq y^{k''}$, since $d^\infty \cdot \widehat{d^{k''-1}} < 0$. Thus, since $d^\infty \cdot \widehat{m} > 0$ and $d^{k''-1} \cdot \widehat{m} \geq 0$, we conclude that

$$\begin{aligned} w \cdot \widehat{m} &= xd^{k''-1} \cdot \widehat{m} + yd^\infty \cdot \widehat{m} \\ &> x^{k''} d^{k''-1} \cdot \widehat{m} + y^{k''} d^\infty \cdot \widehat{m} \\ &= \mathbf{v}^{k''}(s) \cdot \widehat{m}, \end{aligned}$$

so $w \notin H\left(\mathbf{v}^{k''}, m\right)$ and $w \notin X(a)$. □

A.3 Convergence

To prove convergence, we will need another "purely geometric" fact about orbital trajectories. Let us say that the subsequence $\left\{\left(\mathbf{v}^l, d^l\right)\right\}_{l=k'}^{k''}$ has the *covering property* if for all $d \in \mathbb{R}^2$, there exist $l \in \{k', \ldots, k''-1\}$ and $\alpha, \beta \geq 0$ such that

$$d = \alpha d^l + \beta d^{l+1}.$$

In other words, d lies between d^l and d^{l+1}. The first result is the following:

Lemma 11 (Covering)**.** *If the subsequence $\left\{\left(\mathbf{v}^l, d^l\right)\right\}_{l=k'}^{k''}$ satisfies the covering property, then*

$$\cap_{l=k'}^{k''} H\left(\mathbf{v}^l, d^l\right) \subseteq co\left(\cup_{l=k'}^{k''}\{\mathbf{v}^l\}\right).$$

Proof of Lemma 11. Let

$$X = \cap_{l=k'}^{k''-1} H\left(\mathbf{v}^l(s), d^l\right),$$

and let

$$Y = \text{co}\left(\cup_{l=k'}^{k''}\left\{\mathbf{v}^l(s)\right\}\right)$$

denote the convex hull of the trajectory of the subsequence in state s, which are both convex sets.

Suppose that there exists a $v \in X \setminus Y$. By the separating hyperplane theorem, there is a direction d such that $w \cdot \widehat{d} < v \cdot \widehat{d}$ for all $w \in Y$. In particular, $\mathbf{v}^l(s) \cdot \widehat{d} < v \cdot \widehat{d}$ for all $l = k', \ldots, k''$. Because of the covering property, we can find an $l \in \{k', \ldots, k''-1\}$ and $\alpha, \beta \geq 0$ such that $d = \alpha d^l + \beta d^{l+1}$. Note that $v \in X$ implies that $v \in H\left(\mathbf{v}^l(s), d^l\right)$ and also that $v \in H\left(\mathbf{v}^{l+1}(s), d^{l+1}\right) = H\left(\mathbf{v}^l(s), d^{l+1}\right)$. This means that

$$\begin{aligned} v \cdot \widehat{d^l} &\leq \mathbf{v}_l(s) \cdot \widehat{d^l} \\ v \cdot \widehat{d^{l+1}} &\leq \mathbf{v}^l(s) \cdot \widehat{d^{l+1}} \end{aligned}$$

so that

$$\begin{aligned} v \cdot \widehat{d} &= \alpha v \cdot \widehat{d^l} + \beta v \cdot \widehat{d^{l+1}} \\ &\leq \alpha \mathbf{v}^l(s) \cdot \widehat{d^l} + \beta \mathbf{v}^l(s) \cdot \widehat{d^{l+1}} \\ &= \mathbf{v}^l(s) \cdot \widehat{d}, \end{aligned}$$

so that d cannot in fact separate v from Y. □

Naturally, we will need the fact that complete revolutions of the pivot satisfy the covering property.

Lemma 12 (Complete revolutions)**.** *A complete revolution satisfies the covering property.*

Proof of Lemma 12. Let $d \in \mathbb{R}^2$ be an arbitrary direction, and let us suppose that $d \cdot \widehat{d^N} \leq 0$. The case where $d \cdot \widehat{d^N} \geq 0$ is symmetric and is omitted.

We will show that $d \in X^l$ for some l, where

$$X^l = \left\{\alpha d^l + \beta d^{l+1} | \alpha \geq 0, \beta \geq 0\right\}$$

for $l = k, \ldots, k' - 1$. Note that the X^l are additive (convex) cones, and some of the X^l may be one-dimensional if $d^{l+1} = x d^l$ for some $x > 0$. Note that a sufficient condition for d to be in X^l, as long as $d^l \not\propto d^{l+1}$, is that $d \cdot \widehat{d^{l+1}} \leq 0$ and $d \cdot \widehat{d^l} \geq 0$. This can be easily verified by decomposing d in (d^l, d^{l+1}) coordinates and verifying that the coefficients are both positive.

Now, observe that X^k contains d^N. Since $d^{k+1} \cdot \widehat{d^N} < 0$, there is a smallest $\widetilde{k} \in \{k + 2, \ldots, k'\}$ such that $d^{\widetilde{k}} \cdot \widehat{d^N} \geq 0$, which we know exists because k' starts a revolution so it is true for $\widetilde{k} = k' - 1$.

Claim: $-d^N \in X^{\widetilde{k}-1}$. Why? $d^{\widetilde{k}-1} \cdot \widehat{d^N} < 0$ and $d^{\widetilde{k}} \cdot \widehat{d^N} \geq 0$, so that there exists some positive weights α and β such that $d' = \alpha d^{\widetilde{k}-1} + \beta d^{\widetilde{k}}$ satisfies $d' \cdot \widehat{d^N} = 0$. By scaling up or down, we can ensure that either $d' = d^N$ or $d' = -d^N$. Moreover, we know that $d' \cdot \widehat{d^{\widetilde{k}-1}} \leq 0$, which cannot be true if $d' = d^N$.

Thus, we can define the following sets:

$$\begin{aligned} Y^k &= \left\{\alpha d^N + \beta d^{k+1} | \alpha \geq 0, \beta \geq 0\right\}; \\ Y^{\widetilde{k}-1} &= \left\{\alpha d^{\widetilde{k}-1} + \beta(-d^N) | \alpha \geq 0, \beta \geq 0\right\}; \\ Y^l &= X^l \text{ if } k < l < \widetilde{k} - 1. \end{aligned}$$

Suppose that $d \notin X^l$ for any l. Then since $Y^l \subseteq X^l$ for $l = k, \ldots, \widetilde{k} - 1$, we conclude that $d \notin Y^l$ either. We shall see that this leads to a contradiction.

In particular, since $d \cdot \widehat{d^N} \leq 0$, $d \notin Y^k$ implies that $d \cdot \widehat{d^{k+1}} < 0$. Continuing inductively for $l = k + 1, \ldots, \widetilde{k} - 2$, if $d \cdot \widehat{d^l} < 0$ and $d \notin Y^l = X^l$, then we conclude that $d \cdot \widehat{d^{l+1}} < 0$. Once we reach $l = \widetilde{k} - 1$, we know that $d \cdot \widehat{d^{\widetilde{k}-1}} < 0$, and since $d \notin Y^{\widetilde{k}-1}$, we conclude that $-d \cdot \widehat{d^N} < 0$, or equivalently that $d \cdot \widehat{d^N} > 0$, a contradiction. □

Proof of Lemma 7. The base case $r = 0$ is trivially satisfied with $B^0(\mathbf{W}^0) = \mathbf{W}^0$. Note that we will assume that the APS sequence is monotonically decreasing, so that $B(\mathbf{W}^0) \subseteq \mathbf{W}^0$.

Let us write $\widetilde{\mathbf{W}}^r = B^r\left(\mathbf{W}^0\right)$ for the APS sequence, and $\underline{\widetilde{\mathbf{w}}}^r = \underline{\mathbf{w}}\left(\widetilde{\mathbf{W}}^r\right)$ for the corresponding sequence of threat tuples.

Let us suppose that the hypothesis is inductively true at iteration k, so that $\mathbf{W}^k \subseteq \widetilde{\mathbf{W}}^r$ and $\underline{\mathbf{w}} \geq \underline{\widetilde{\mathbf{w}}}^r$, where $r = \lfloor r(k)/2 \rfloor$. We write $h\left(a, \underline{\mathbf{w}}^k\right)$, $IC\left(a, \underline{\mathbf{w}}^k\right)$, $\overline{W}\left(a, \mathbf{W}^k\right)$, and $Gen\left(a, \underline{\mathbf{w}}^k, \mathbf{W}^k\right)$ to emphasize the dependence of these objects on the threat tuple and the feasible correspondence. Note that $\underline{\mathbf{w}}^k \geq \underline{\mathbf{w}}\left(\widetilde{\mathbf{W}}^r\right) = \underline{\widetilde{\mathbf{w}}}^{r(k):0}$ and $\mathbf{W}^k \subseteq \widetilde{\mathbf{W}}^r$, so that

$$\begin{aligned} h\left(a, \underline{\mathbf{w}}^k\right) &\geq h\left(a, \underline{\mathbf{w}}^k, \mathbf{W}^k\right); \\ IC\left(a, \underline{\mathbf{w}}^k\right) &\subseteq IC\left(a, \underline{\widetilde{\mathbf{w}}}^r\right); \\ \overline{W}\left(a, \mathbf{W}^k\right) &\subseteq \overline{W}\left(a, \widetilde{\mathbf{W}}^r\right); \\ Gen\left(a, \underline{\mathbf{w}}^k, \mathbf{W}^k\right) &\subseteq Gen\left(a, \underline{\widetilde{\mathbf{w}}}^r, \widetilde{\mathbf{W}}^r\right). \end{aligned}$$

Let us suppose that a generates the best test direction in state s at iteration k. During the updating procedure, we inductively construct a sequence of new pivots $\{\mathbf{v}^{k+1,l}\}_{l=0}^{\infty}$ where $\mathbf{v}^{k,0} = \mathbf{v}^k$ and $\mathbf{v}^{k+1,l+1}(s) = \mathbf{v}^{k+1,l}(s) + \mathbf{x}^{l+1}(s)d^{k+1}$. The best direction is always generated by some payoff $v \in Gen\left(a, \underline{\mathbf{w}}^k, \mathbf{W}^k\right) \subseteq Gen\left(a, \underline{\widetilde{\mathbf{w}}}^r, \widetilde{\mathbf{W}}^r\right)$, so that the new pivot $\mathbf{v}^{k+1,0}(s)$ is in $Gen\left(a, \underline{\widetilde{\mathbf{w}}}^r, \widetilde{\mathbf{W}}^r\right)$, which is necessarily contained in $B\left(\widetilde{\mathbf{W}}^r\right)(s)$.

Now consider the states in which the pivot is updated at $l > 0$ because $\mathbf{r}^{k+1,l}(s) = \text{NB}$. Again, new payoffs are generated in those states using a convex combination of $\mathbf{v}^{k+1,l-2}$ and $\mathbf{v}^{k+1,l}$ (with the convention that $\mathbf{v}^{k+1,-1} = \mathbf{v}^k$), which, inductively, must be contained in $Gen\left(a, \mathbf{W}^k\right)$. Moreover, this convex combination puts the highest possible weight on $\mathbf{v}^{k+1,l-1}$ such that the expected continuation value is incentive compatible with respect to the more larger and more stringent threats $\underline{\mathbf{w}}^k$, so that the generated payoff must be in $Gen\left(a, \underline{\widetilde{\mathbf{w}}}^r, \widetilde{\mathbf{W}}^r\right)$. Hence, the updated pivot $\mathbf{v}^{k+1,l}(s) \in B\left(\widetilde{\mathbf{W}}^r\right)(s)$ for all l. We conclude that $\mathbf{v}^{k+1}(s) \in B\left(\widetilde{\mathbf{W}}^r\right)(s)$ for any state for which $\mathbf{v}^{k+1}(s) \neq \mathbf{v}^k(s)$.

Thus, every time a pivot is updated, it must be contained in $B\left(\widetilde{\mathbf{W}}^r\right)(s)$, and therefore contained in $\widetilde{\mathbf{W}}^r$ as well. Since $\mathbf{V}$ is full dimension, we have to update $\mathbf{v}^k(s)$ at least twice per revolution for each s. Starting from iteration $2r:0$, the trajectory on the $2r$th revolution, $\{\mathbf{v}^k\}_{k=2r:0}^{2r+1:-1}$, will be contained in $\widetilde{\mathbf{W}}^r$, so that $\mathbf{W}^{2r+1:c} \subseteq \widetilde{\mathbf{W}}^r$ as well. Now, for $k \geq 2r+1:0$, we have $\mathbf{v}^k \in B\left(\widetilde{\mathbf{W}}^r\right)$. Thus, the trajectory of the pivot on the $2(r+1)$th revolution, $\{\mathbf{v}^k\}_{k=2r+1:0}^{2(r+1):-1}$, is contained in $B\left(\widetilde{\mathbf{W}}^r\right)$. Since the trajectory of the pivot satisfies the rotation

property, Lemma 11 implies that for $k \geq 2(r+1):0$,

$$\begin{aligned} W^k &= \mathbf{W}^{0:0} \cap_{l=0}^{k} H\left(\mathbf{v}^k, d^k\right) \\ &\subseteq \cap_{l=2r+1:0}^{2(r+1):0} H\left(\mathbf{v}^k, d^k\right) \\ &\subseteq \operatorname{co}\left(\cup_{l=2r+1:0}^{2(r+1):0} \{\mathbf{v}^l\}\right) \\ &\subseteq B\left(\widetilde{\mathbf{W}}^r\right) = \widetilde{\mathbf{W}}^{r+1}. \end{aligned}$$

□