Algorithms for Stochastic Games
Supplemental Appendix

Dilip Abreu Benjamin Brooks Yuliy Sannikov

January 30, 2020

C Pseudocode for Sections 3, 4, and 5

Algorithm 1 Minimize regimes.

procedure MINIMIZEREGIMES(λ,$\mathbf{a}$, $\mathbf{r}$, $\mathbf{W}$)
    define $\widetilde{S}$ to be the states with $\gamma(\mathbf{a}(s), \lambda, \mathbf{W}) > 0$
    define $\mathbf{r}' := \mathbf{r}$
    for all $s \in \widetilde{S}$ **do**
        $\mathbf{r}'(s) := R$ ▷ For these states, recursive can be taken to be minimal
    loop
        $\mathbf{r}'' := \mathbf{r}'$
        for all $s \notin \widetilde{S}$ **do**
            if $\widehat{x}^{APS}(\mathbf{a}(s), \lambda, \mathbf{W}) < x(s, \lambda, \mathbf{a}, \mathbf{r}', \mathbf{W})$ **then**
                $\mathbf{r}''(s) := APS$ ▷ The best APS payoff is lower
            else if $x^R(\mathbf{a}(s), \lambda, \mathbf{a}, \mathbf{r}', \mathbf{W}) < x(s, \lambda, \mathbf{a}, \mathbf{r}', \mathbf{W})$ **then**
                $\mathbf{r}''(s) := R$ ▷ The recursive payoff is lower
        if $\mathbf{r}'' \neq \mathbf{r}'$ **then**
            $\mathbf{r}' := \mathbf{r}''$ ▷ Continue updating
        else
            return $\mathbf{r}'$ ▷ These regimes are minimal

Algorithm 2 Optimize the policy.

procedure OptimizePolicy(λ,$\mathbf{W}$)
  define $\mathbf{a} \in \mathbf{A}(\mathbf{W})$
  define $\mathbf{r} \in \mathbf{R}$
  loop
    define $\mathbf{a}' := \mathbf{a}$
    $\mathbf{r} :=$ MinimizeRegimes($\lambda, \mathbf{a}, \mathbf{r}, \mathbf{W}$)
    for all $s \in S$, $a \in \mathbf{A}(\mathbf{W})(s)$ **do**
      if and($x^R(a, \lambda, \mathbf{a}, \mathbf{r}, \mathbf{W}) > x(s, \lambda, \mathbf{a}, \mathbf{r}, \mathbf{W})$,
          or($\gamma(a, \lambda, \mathbf{W}) > 0$, $\widehat{x}^{APS}(a, \lambda, \mathbf{W}) > x(s, \lambda, \mathbf{a}, \mathbf{r}, \mathbf{W})$)) **then**
        $\mathbf{a}'(s) := a$
    if $\mathbf{a} \neq \mathbf{a}'$ **then**
      $\mathbf{a} := \mathbf{a}'$ ▷ Continue updating
    else
      return $(\mathbf{a}, \mathbf{r})$ ▷ The policy is optimal

Algorithm 3 Compute the shallowest legitimate test direction with $N = 2$.

Require: $\mathbf{u}$ is robustly optimal for direction λ

procedure FindNextDirection(λ,$\mathbf{u}$,$\mathbf{W}$)
  define $\lambda' := \lambda$
  for all $s \in S$, $a \in \mathbf{A}(\mathbf{W})(s)$, $p \in \{R\} \cup C(a, \mathbf{W})$ **do** ▷ Iterate over all substitutions
    for all λ'' that is a test direction for (s, a, p) given $\mathbf{u}$ **do**
      if and (λ'' is legitimate, λ'' is shallower than λ') **then**
        $\lambda' := \lambda''$
  return λ' ▷ The optimal payoffs may change at λ'

Given $N = 2$, let $\widehat{u}^{APS}(a, \lambda, \mathbf{W})$ be the highest binding APS payoff in the direction λ where comparisons are made lexicographically using $>_\lambda$. In a slight abuse of notation, we write $\gamma(a, \lambda^+, \mathbf{W}) > 0$ if the APS gap for a is lexicographically positive at λ. The following procedure, analogous to Algorithm 1, uses lexicographic comparisons to choose the regimes which become minimal for the action tuple $\mathbf{a}$ after direction λ

Algorithm 4 Lexicographically minimize regimes for $N = 2$.

procedure LexMinimizeRegimes(λ,$\mathbf{a}, \mathbf{p}, \mathbf{W}$)
    define $\widetilde{S}$ to be the states where $\gamma(a, \lambda^+, \mathbf{W}) > 0$
    define $\mathbf{p}' := \mathbf{p}$
    for all $s \in \widetilde{S}$ **do**
        $\mathbf{p}'(s) := R$ ▷ For these states, recursive must be minimal
    loop
        define $\mathbf{p}'' := \mathbf{p}'$
        define $\mathbf{u}$:= the payoffs induced by $(\mathbf{a}, \mathbf{p}')$
        for all $s \notin \widetilde{S}$ **do**
            if $\mathbf{u}(s) >_\lambda \widehat{u}^{APS}(\mathbf{a}(s), \lambda, \mathbf{W})$ **then**
                $\mathbf{r}''(s) := APS$ ▷ The best APS payoff is lexicographically lower
            else if $\mathbf{u}(s) >_\lambda u^R(\mathbf{a}(s), \lambda, \mathbf{u})$ **then**
                $\mathbf{p}''(s) := R$ ▷ The recursive payoff is lexicographically lower
        if $\mathbf{p}'' \neq \mathbf{p}'$ **then**
            $\mathbf{p}' := \mathbf{p}''$ ▷ Continue updating
        else
            return $\mathbf{p}'$ ▷ This $\mathbf{p}$ is minimal

The following procedure, analogous to Algorithm 2, uses lexicographic comparisons to find the pair which is optimal after direction λ (i.e. the robustly optimal pair).

Algorithm 5 Lexicographically optimize the policy.

procedure LexOptimizePolicy(λ,$\mathbf{a}$, $\mathbf{p}$, $\mathbf{W}$)
    define $\mathbf{a}' := \mathbf{a}$
    define $\mathbf{p}' := \mathbf{p}$
    loop
        define $\mathbf{a}'' := \mathbf{a}'$
        $\mathbf{p}' :=$ LexMinimizeRegimes($\lambda, \mathbf{a}', \mathbf{p}', \mathbf{W}$)
        define $\mathbf{u} :=$ the payoffs induced by $(\mathbf{a}', \mathbf{p}')$
        for all $s \in S$, $a \in \mathbf{A}(\mathbf{W})(s)$ **do**
            if and $(u^R(a, \lambda, \mathbf{u}) >_\lambda \mathbf{u}(s)$,
              or$(\gamma(a, \lambda^+, \mathbf{W}) > 0$, $\widehat{u}^{APS}(a, \lambda, \mathbf{W}) >_\lambda \mathbf{u}(s)))$ **then**
                $\mathbf{a}''(s) := a$
        if $\mathbf{a}'' \neq \mathbf{a}'$ **then**
            $\mathbf{a}' := \mathbf{a}''$ ▷ Continue updating the actions
        else
            for all $s \in S$ **do**
                if $\mathbf{p}'(s) = u^R(\mathbf{a}'(s), \mathbf{u})$ **then**
                    $\mathbf{p}'(s) := R$ ▷ Make the pair canonical
            return $(\mathbf{a}', \mathbf{p}')$ ▷ Return the optimal pair

Algorithm 6 Compute $\widetilde{B}$ for $N = 2$.

Require: $B(\mathbf{W}) \subseteq \mathbf{W}$

procedure $\widetilde{B}(\mathbf{W})$
    for all $s \in S$ **do**
        define $\mathbf{A}(\mathbf{W})(s) = \emptyset$
        for all $a \in \mathbf{A}(s)$ **do**
            Compute $C(a, \mathbf{W})$
            if $C(a, \mathbf{W}) \neq \emptyset$ **then**
                $\mathbf{A}(\mathbf{W})(s) := \mathbf{A}(\mathbf{W})(s) \cup \{a\}$
    if $\mathbf{A}(\mathbf{W})(s) = \emptyset$ for some s **then**
        return an empty correspondence
    define $\mathbf{W}' := (\mathbb{R}^N)^S$ ▷ There are supportable actions
    define $\lambda := (1, 0)$ ▷ Begin pointing due east
    define $(\mathbf{a}, \mathbf{p})$ to be an arbitrary pair
    loop
        define $(\mathbf{a}', \mathbf{p}') :=$ LexOptimizePolicy$(\lambda, \mathbf{a}, \mathbf{p}, \mathbf{W})$
        define $\mathbf{u} :=$ the payoffs induced by $(\mathbf{a}, \mathbf{p})$
        $\lambda' :=$ FindNextDirection$(\lambda, \mathbf{u}, \mathbf{W})$
        $\mathbf{W}' := \mathbf{W}' \cap \{\lambda' \cdot \mathbf{v} \leq \lambda' \cdot \mathbf{u}\}$ ▷ Intersect $\mathbf{W}'$ with the new half space
        if λ points strictly north **and** λ' points weakly south **then**
            return $\mathbf{W}'$ ▷ Completed a full revolution
        else
            $\lambda := \lambda'$, $\mathbf{a} := \mathbf{a}'$ ▷ Continue with the new direction

Algorithm 7 Compute $\mathbf{V}$ to a tolerance ϵ in the metric d. Returns the approximation.

Require: $B(\widetilde{\mathbf{W}}^0) \subseteq \widetilde{\mathbf{W}}^0$ and $\mathbf{V} \subseteq \widetilde{\mathbf{W}}^0$

procedure Solve$(\widetilde{\mathbf{W}}^0, \epsilon)$
    define $k := 0$
    do
        $k := k + 1$
        $\widetilde{\mathbf{W}}^k := \widetilde{B}(\widetilde{\mathbf{W}}^{k-1})$
    while $d(\widetilde{\mathbf{W}}^k, \widetilde{\mathbf{W}}^{k-1}) > \epsilon$ ▷ Stop when the movement is small
    return $\widetilde{\mathbf{W}}^k$

Next, given directions λ and $\widetilde{\lambda}$, we define the $(\lambda, \widetilde{\lambda})$-line to be the subset of directions in Λ of the form $\cos(\theta)\lambda + \sin(\theta)\widetilde{\lambda}$, where $\theta \in (0, 2\pi]$. We order $(\lambda, \widetilde{\lambda})$-line according to θ in this parameterization. We also extend the notion of test directions for the substitution (s, a, p) given the payoffs $\mathbf{u}$ to be any direction satisfying (14). Legitimacy also extends to this setting. Finally, we redefine robust optimality in the many player setting by saying that $\mathbf{u}$ is robustly optimal if it remains optimal in a neighborhood of λ. (Note that this definition is more restrictive than what is used in Section 4, where robustly optimal payoffs only had to remain optimal for perturbations in one direction.)

Algorithm 8 Update the direction by rotating towards $\widetilde{\lambda}$. Returns the new direction of optimization and the direction in which payoffs move.

Require: $\mathbf{u}$ is robustly optimal at λ

```
procedure RotateDirection(λ, λ̃, u, W)
    for all s ∈ S, a ∈ A(W)(s), p ∈ {R} ∪ C(a, W) do
        for all test directions λ'' for (s, a, p) and u in the (λ, λ̃)-line do
            if λ'' is legitimate and a smaller rotation than λ' then
                λ' := λ''
                d := u(s, a, p, u) − u(s)        ▷ the direction in which (s, a, p) moves payoffs
    return (λ', d)
```

Algorithm 9 Compute a randomly chosen face of $\widetilde{B}(\mathbf{W})$. Return the direction and the corresponding half space.

Require: $B(\mathbf{W}, \widehat{\Lambda}) \subseteq \mathbf{W}$

```
procedure FindFace(W)
    define λ^0 randomly
    define (a, r) := OptimizePolicy(λ^0, W)
    define p ∈ P(a, W) to be min-max for (a, r, W)
    define u := payoffs induced by (a, p)
    for n = 1, ..., N − 1 do
        define λ̃^n randomly to be orthogonal to {λ^0} ∪ {d^l | l = 1, ..., n − 1}
        define (λ^n, d^n) := RotateDirection(λ^{n−1}, λ̃^n, u, W)
    define H := {v | λ^{N−1} · v ≤ λ^{N−1} · u}
    return (λ^{N−1}, H)
```

Algorithm 10 Approximate $\widetilde{B}(\mathbf{W})$, given an incumbent set of directions $\widehat{\Lambda}$. Returns a new approximation and a new set of directions.

Require: $B(\mathbf{W}, \widehat{\Lambda}) \subseteq \mathbf{W}$

procedure $\widetilde{B}(\mathbf{W},\widehat{\Lambda},L)$
  define $\mathbf{W}' := (\mathbb{R}^N)^S$
  define $\widehat{\Lambda}' := \emptyset$
  for all $\lambda \in \widehat{\Lambda}$ **do**
    $(\mathbf{a}, \mathbf{r}) :=$ OptimizePolicy$(\lambda, \mathbf{W})$
    define $\mathbf{p} \in \mathbf{P}(\mathbf{a}, \mathbf{W})$ to be min-max for $(\mathbf{a}, \mathbf{r}, \mathbf{W})$
    define $H := \{\mathbf{v} | \mathbf{v} \cdot \lambda' \leq x(\lambda', \mathbf{p}, \mathbf{W})\}$
    if $\mathbf{W}'$ and $\mathbf{W}' \cap H$ do not have the same local binding frontier **then**
      $\widehat{\Lambda}' := \widehat{\Lambda}' \cup \{\lambda\}$
      $\mathbf{W}' := \mathbf{W}' \cap H$
  define $K := |\widehat{\Lambda}'|$
  for $k = 1, \ldots, L - K$ **do**
    $(\lambda, H) :=$ FindFace$(\mathbf{W})$
    if $\mathbf{W}'$ and $\mathbf{W}' \cap H$ do not have the same local binding frontier **then**
      $\mathbf{W}' := \mathbf{W}' \cap H$
      $\widehat{\Lambda}' := \widehat{\Lambda}' \cup \{\lambda\}$
  return $(\mathbf{W}', \widehat{\Lambda}')$

Algorithm 10 can be combined with an analogue of Algorithm 7 to approximate $\mathbf{V}$ when $N > 2$.

D Connections to linear programming and dynamic programming

To the student of linear programming, our procedure may evoke the simplex algorithm and sensitivity analysis. The choice of $(\mathbf{a}, \mathbf{r})$ bears a resemblance to the choice of a basis, and our use of test directions and optimization is similarly reminiscent of parametric programming in the theory of linear programming (see Dantzig and Thapa, 2006, for a comprehensive treatment). In this section, we attempt to elucidate the connection.

Suppose we were not concerned with incentives at all and simply wanted to compute the feasible payoff correspondence $\mathbf{F}$, i.e., payoffs that can be obtained with some pure-strategy profile starting in state s (still allowing public randomization). For a fixed direction λ, the problem of computing the optimal levels

$$x(s, \lambda) = \max\{\lambda \cdot v | v \in \mathbf{F}(s)\}$$

is a Markov decision problem. It is shown by Blackwell (1962) that there is an optimal strategy profile which is stationary and given by some $\mathbf{a} \in \mathbf{A}$. There are many ways to compute the solution, including value function iteration, policy function iteration, and linear programming. In particular, the levels $\{x(s, \lambda)\}_{s \in S}$ are the solution to the linear program

$$\min_{y^R(\cdot)} \sum_{s \in S} y^R(s) \tag{1a}$$

$$\text{s.t. } y^R(s) \geq (1 - \delta)\lambda \cdot g(a) + \delta \sum_{s' \in S} \pi(s'|a) y^R(s') \text{ for all } s \in S, a \in \mathbf{A}(\mathbf{W})(s). \tag{1b}$$

A solution can be computed via the simplex algorithm, which will select exactly $|S|$ of the constraints to bind, so that their intersection uniquely pins down the value of y^R. At an optimum, there must be a binding constraint in each state, since otherwise we could decrease $y^R(s)$, and simultaneously decrease the right-hand side of every constraint. The choice of binding constraints is therefore a choice of exactly one action profile for each state, i.e., an $\mathbf{a} \in \mathbf{A}(\mathbf{W})$, which is an optimal policy. The simplex algorithm would identify such an optimal policy as a basic solution to the LP (1).

It has long been understood that the output of the simplex algorithm can be used to conduct "sensitivity analysis": how much can we perturb the original problem without changing the optimal basis? In our case, we are concerned with sensitivity to λ, and for what range of directions would the optimal solution remain the same. As we rotate λ, we change the constants in the constraints. Eventually the optimal basis will change, and generically a single constraint will leave the basis and be replaced by a new one. This corresponds to changing the optimal policy in a single state. The next action to enter can be determined using well-known techniques, as in Dantzig and Thapa (2006). Mapping out the set of solutions for all λ is known as *parametric programming*, which is also a well established concept in mathematical optimization. In fact, this is precisely how our algorithm would behave if we restricted ourselves to using $\mathbf{r}(s) = R$, in which case the algorithm would converge in exactly one iteration (provided we start with any compact and convex valued correspondence that contains the feasible correspondence, e.g., large boxes whose bounds are given by the maximum and minimum flow payoffs across all states and actions.)

This is not our program, since we do have incentive constraints. It is in that sense closer to the problem of APS, which can also be formulated as an LP thusly: $x^{APS}(s,\lambda)$ is the solution to

$$\min_{y^{APS}(\cdot)} \sum_{s \in S} y^{APS}(s) \tag{2a}$$

$$\text{s.t. } y^{APS}(s) \geq \max\{\lambda \cdot v | v \in B(a, \mathbf{W})\} \text{ for all } s \in S, a \in \mathbf{A}(\mathbf{W})(s). \tag{2b}$$

This is not an LP in standard form, because of the inner maximization which is also an LP. But that problem can be replaced with its dual, in which case we have a single minimization program. Suppose that $\mathbf{W}$ has finitely many faces with normals $\{\lambda^l\}_{l=0}^L$ and corresponding levels $\{z_l(s)\}_{l=0}^L$. Let $\mu_l(a,s)$ denote the multiplier on feasibility of the continuation value for action a in state s in the direction λ, and let $\alpha_i(a)$ denote the multiplier on the incentive constraint for player i. Applying the strong duality theorem of linear programming, we conclude that the best APS payoff is equal to the minimum of

$$y^{APS}(a) = \sum_{s' \in S} \sum_{l=1}^{L} \mu_l(a,s)((1-\delta)\lambda^l \cdot g(a) + \delta z_l(s)) - \sum_{i=1}^{N} \alpha_i(a) \underline{u}_i(a)$$

across all μ_l and α_l that are non-negative. Thus, we can expand (2) to

$$\min_{y^{APS}(\cdot), \mu_l(\cdot), \alpha_i(\cdot)} \sum_{s \in S} y^{APS}(s)$$
$$\text{s.t. } y^{APS}(s) \geq y^{APS}(a) \text{ for all } s \in S, a \in \mathbf{A}(\mathbf{W})(s) \tag{3}$$

Again, this LP could be solved using the simplex algorithm, and one can map out the set of all basic solutions for all λ using sensitivity analysis and parametric programming.

Again, this is not our program. Ours is in fact a hybrid of the two:

$$\min_{y(\cdot),y^R(\cdot),y^{APS}(\cdot),\mu_l(\cdot)\geq 0,\alpha_i(\cdot)\geq 0} \sum_{s\in S} y(s) \tag{4a}$$

$$\text{s.t. (1b) and (2b) and (3)}$$

$$y(s) \geq \min\{y^R(a), y^{APS}(a)\}\ \forall s \in S, a \in \mathbf{A}(\mathbf{W})(s). \tag{4b}$$

This is *not* an LP, because of the min operator in (4b). However, we can modify this program to make it into a larger LP, so that one could again use sensitivity analysis and parametric programming to map out solutions.

Specifically, we can add parameters $r(a) \in \{R, APS\}$ (which are not variables in the LP) and replace (4a) and (4b) with

$$\begin{aligned} \min_{y(\cdot),y^R(\cdot),y^{APS}(\cdot),\mu_l(\cdot)\geq 0,\alpha_i(\cdot)\geq 0} \sum_{s\in S} & \left[y(s) + \sum_{a\in\mathbf{A}(\mathbf{W})(s)} (y^R(a) + y^{APS}(a)) \right] \\ \text{s.t. (1b) and (2b) and (3)} & \\ y(s) \geq y^{r(a)}(a)\ & \forall s \in S, a \in \mathbf{A}(\mathbf{W})(s) \end{aligned} \tag{5}$$

This is now an LP, and the $y(s)$ in the solution corresponds to the optimal levels under a particular conjecture as to which are the minimizing regimes, action profile by action profile. We could compute the level $x(s,\lambda)$ for a fixed direction by solving a sequence of such LPs, where at each step, we replace $r(a)$ with $\arg\min_r y^r(a)$, where $y^r(a)$ is taken from the previous solution. This will necessarily produce a decreasing sequence of solutions, whose limit is $x(s,\lambda)$.

Now, once we reach the optimal solution regimes $r(a)$, if we add one more constraint:

$$y^{r(a)}(a) \leq y^{r'}(a) \text{ for all } s \in S, a \in \mathbf{A}(\mathbf{W})(s), r \in \{R, APS\}, \tag{6}$$

the optimal solution will not change. Moreover, if we do sensitivity analysis on this expanded program, we will exactly find the range of directions λ under which the optimal actions and level-minimizing regimes do not change, action profile by action profile. So, in principle, one way to map out $x(s,\lambda)$ is to do sensitivity analysis on the expanded program of (5) and (6) to find adjacent directions where the solution to that program would change, and for those adjacent directions, resolve (5), re-optimizing the regimes $r(a)$ as needed.

Overall, this is quite a bit more work than what we have done in our more direct implementation. Effectively, the LP-based approach involves computing optimal regimes for every action profile, even those which are not optimal, whereas our main procedure only computes minimal regimes for maximal action profiles. We have even implemented the LP based algorithm for two players using Gurobi, a high-performance commercial linear programming package. We found that this program took an order of magnitude longer to solve than the more direct implementation described in Section 4.2.

Nonetheless, this discussion may help to explain where the linear structure comes from, and why we end up using similar objects as those which arise in linear programming. It may also explain why we cannot simply use off-the-shelf techniques from linear programming in determining the function $x(s, \lambda)$.

E Redux for $\widetilde{B}^\epsilon$

This appendix extends the key results from Sections 3 and 4 to the operator $\widetilde{B}^\epsilon$.

E.1 Convergence results

Define the operator

$$T^\epsilon(\mathbf{y}, \lambda, \mathbf{a}, \mathbf{r}, \mathbf{W})(s) = -\epsilon + \begin{cases} (1-\delta)\lambda \cdot g(\mathbf{a}(s)) + \delta \sum_{s' \in S} \mathbf{y}(s')\pi(s'|\mathbf{a}(s)) & \text{if } \mathbf{r}(s) = R; \\ x^{APS}(\mathbf{a}(s), \lambda, \mathbf{W}) & \text{if } \mathbf{r}(s) = APS. \end{cases}$$

Lemma 1 (Operator T^ϵ)**.** *Fix λ, $\mathbf{a}$, $\mathbf{r}$, and $\mathbf{W}$. As a function of $\mathbf{y} : S \to \mathbb{R}$, T^ϵ is*

(L1.i) *increasing;*

(L1.ii) *a contraction with modulus δ and hence has a unique fixed point $\mathbf{y}^*$;*

(L1.iii) *if $T^\epsilon(\mathbf{y}) \leq (\geq)\mathbf{y}$ then $\mathbf{y}^* \leq (\geq) T^\epsilon(\mathbf{y})$.*

Proof. The proof coincides verbatim with that of Lemma 1, changing T to T^ϵ. □

Theorem 1 (The perturbed max-min-max algorithm)**.** *For every $\epsilon > 0$, as a function of $\mathbf{W} : S \to 2^{\mathbb{R}^N}$, the operator $\widetilde{B}^\epsilon$ has the following properties:*

(T1.i) *$\widetilde{B}^\epsilon$ is increasing in $\mathbf{W}$, and if $\mathbf{W}$ is compact, then $\widetilde{B}^\epsilon(\mathbf{W})$ is compact;*

(T1.ii) *$\widetilde{B}^\epsilon(\mathbf{W}) \subseteq B^\epsilon(\mathbf{W})$. Thus, if $\mathbf{W} \subseteq \widetilde{B}^\epsilon(\mathbf{W})$, then $\mathbf{W}$ is self-generating and $\mathbf{W} \subseteq \mathbf{V}^\epsilon$;*

(T1.iii) *$\mathbf{V}^\epsilon = \widetilde{B}^\epsilon(\mathbf{V}^\epsilon)$;*

(T1.iv) *Fix a correspondence $\widetilde{\mathbf{W}}^0$ that contains $\mathbf{V}^\epsilon$. Define the sequence $\left\{\widetilde{\mathbf{W}}^k\right\}_{k=0}^\infty$ by $\widetilde{\mathbf{W}}^k = \widetilde{B}^\epsilon\left(\widetilde{\mathbf{W}}^{k-1}\right)$. Then $\mathbf{V}^\epsilon = \cap_{k=0}^\infty \widetilde{\mathbf{W}}^k$.*

Proof of Theorem 1.

(T1.i) For every λ and $(\mathbf{a}, \mathbf{r})$, we can write

$$\eta(s, \mathbf{a}, \mathbf{r}) = x(s, \lambda, \mathbf{a}, \mathbf{r}, \mathbf{W}) - x^\epsilon(s, \lambda, \mathbf{a}, \mathbf{r}, \mathbf{W}).$$

Then η uniquely solves the system of equations

$$\eta(s, \mathbf{a}, \mathbf{r}) = \epsilon + \begin{cases} 0 & \text{if } \mathbf{r}(s) = APS; \\ \delta \sum_{s' \in S} \pi(s'|\mathbf{a}(s))\eta(s', \mathbf{a}, \mathbf{r}) & \text{otherwise.} \end{cases}$$

Note for future reference that η is independent of both λ and $\mathbf{W}$. Thus, since x is monotonic in $\mathbf{W}$, so is x^ϵ. This implies monotonicity of $\widetilde{B}^\epsilon$. $\widetilde{B}^\epsilon(\mathbf{W})$ is also closed, being the intersection of closed half-spaces, and bounded because $\widehat{x}^{APS}$ is bounded, so that x^ϵ is bounded as well.

(T1.ii) Clearly, $x^\epsilon(s, \lambda, \mathbf{W}) \leq x^{APS}(s, \lambda, \mathbf{W}) - \epsilon$, which implies that $\widetilde{B}^\epsilon$ is always contained in B^ϵ. Thus, if $\mathbf{W} \subseteq \widetilde{B}^\epsilon(\mathbf{W})$, then $\mathbf{W} \subseteq B^\epsilon(\mathbf{W})$ and hence, by APS, $B^\epsilon(\mathbf{W}) \subseteq \mathbf{V}^\epsilon$. Consequently, $\widetilde{B}^\epsilon(\mathbf{W}) \subseteq \mathbf{V}$.

(T1.iii) From (T1.ii), it suffices to show that $\mathbf{V}^\epsilon \subseteq \widetilde{B}^\epsilon(\mathbf{V}^\epsilon)$, i.e., for all λ, $x^\epsilon(s, \lambda, \mathbf{V}^\epsilon) \geq x^{APS}(s, \lambda, \mathbf{V}^\epsilon) - \epsilon$. To that end, fix λ, and for all s, let $\mathbf{a}(s)$ be an action that maximizes $x^{APS}(a, \lambda, \mathbf{V}^\epsilon)$ and let $\mathbf{w}(\cdot)$ be the associated continuation values as a function of the next-period state s'. We will show that $\min_{\mathbf{r}} x^\epsilon(s, \lambda, \mathbf{a}, \mathbf{r}, \mathbf{V}^\epsilon) \geq x^{APS}(s, \lambda, \mathbf{V}^\epsilon) - \epsilon$, so that $x^\epsilon(s, \lambda, \mathbf{V}^\epsilon) \geq x^{APS}(s, \lambda, \mathbf{V}^\epsilon) - \epsilon$, which implies the result. Since $\mathbf{V}^\epsilon = B^\epsilon(\mathbf{V}^\epsilon)$, $x^{APS}(s, \lambda, \mathbf{V}^\epsilon) - \epsilon \geq \lambda \cdot u$ for all $u \in \mathbf{V}^\epsilon(s')$ for all s'. Since $\mathbf{w}(s') \in \mathbf{V}^\epsilon(s')$ for all s',

$$\begin{aligned} x^{APS}(s, \lambda, \mathbf{V}^\epsilon) &= (1-\delta)\lambda \cdot g(\mathbf{a}(s)) + \delta \sum_{s' \in S} \pi(s'|\mathbf{a}(s)\lambda \cdot \mathbf{w}(s') \\ &\leq (1-\delta)\lambda \cdot g(\mathbf{a}(s)) + \delta \sum_{s' \in S} \pi(s'|\mathbf{a}(s)(x^{APS}(s', \lambda, \mathbf{V}^\epsilon) - \epsilon). \end{aligned}$$

Thus, if we let $\mathbf{y}(s) = x^{APS}(s, \lambda, \mathbf{V}^\epsilon) - \epsilon$ for all s, then for *any* regimes $\mathbf{r}$, $T^\epsilon(\mathbf{y}, \lambda, \mathbf{a}, \mathbf{r}, \mathbf{V}^\epsilon) \geq \mathbf{y}$ (with equality if $\mathbf{r}(s) = APS$ and weak inequality if $\mathbf{r}(s) = R$). By (L1.iii), we conclude that $\mathbf{y}(s) = x^{APS}(s, \lambda, \mathbf{V}^\epsilon) - \epsilon \leq x^\epsilon(s, \lambda, \mathbf{a}, \mathbf{r}, \mathbf{V}^\epsilon) = \mathbf{y}^*(s)$, as required.

(T1.iv) (T1.ii) implies that $\widetilde{\mathbf{W}}^k \subseteq \mathbf{W}^k$, where the latter is the kth element of the APS sequence for B^ϵ starting from $\widetilde{\mathbf{W}}^0$. Also, the fact that $\widetilde{\mathbf{W}}^0$ contains $\mathbf{V}$, (T1.i), and (T1.iii) imply that $\mathbf{V}^\epsilon \subseteq \widetilde{\mathbf{W}}^k$. Thus, $\mathbf{V}^\epsilon \subseteq \cap_k \widetilde{\mathbf{W}}^k \subseteq \cap_k \mathbf{W}^k = \mathbf{V}^\epsilon$.

□

E.2 State independence of the optimal policy

We now restate the results for minimal regimes. Let us define

$$x^{R,\epsilon}(a, \lambda, \mathbf{a}, \mathbf{r}) = (1-\delta)\lambda \cdot g(a) + \delta \sum_{s' \in S} \pi(s'|a) x^\epsilon(s', \lambda, \mathbf{a}, \mathbf{r}).$$

For given λ, $\mathbf{W}$, and $\mathbf{a} \in \mathbf{A}(\mathbf{W})$, we say that the regimes $\mathbf{r}$ are *minimal* if and only if for all $s \in S$,

$$x^{\epsilon}(s, \lambda, \mathbf{a}, \mathbf{r}) = \min_{\mathbf{r}' \in \mathbf{R}} x^{\epsilon}(s, \lambda, \mathbf{a}, \mathbf{r}').$$

Lemma 2 (Minimal regimes)**.** *For all* $\mathbf{a} \in \mathbf{A}(\mathbf{W})$*,* λ*, and* $\epsilon > 0$*,*

(L2.i) *there exists minimal regimes;*

(L2.ii) $\mathbf{r}$ *is minimal if and only if for all* $s \in S$*,*

$$x^{\epsilon}(s, \lambda, \mathbf{a}, \mathbf{r}) = \left\{x^{APS}(\mathbf{a}(s), \lambda), x^{R,\epsilon}(\mathbf{a}(s), \lambda, \mathbf{a}, \mathbf{r})\right\} - \epsilon; \tag{7}$$

(L2.iii) *if (7) is violated for some* s*, then* $\mathbf{r}$ *is not minimal. Moreover, for all* $s' \in S$*,* $x^{\epsilon}(s', \lambda, \mathbf{a}, \mathbf{r} \setminus s) \leq x^{\epsilon}(s', \lambda, \mathbf{a}, \mathbf{r})$*, with strict inequality in state* s*.*

Proof of Lemma 1. The proof follows verbatim that of Lemma 1, replacing T with T^{ϵ}. □

Proof of Lemma 2. The proof follows verbatim that of Lemma 2, replacing T with T^{ϵ}, x with x^{ϵ}, references to equation (6) with (7), and references to Lemma 1 with references to Lemma 1. □

We next extend the results for maximal actions. Define $x^{\epsilon}(s, \lambda, \mathbf{a})$ to be $x^{\epsilon}(s, \lambda, \mathbf{a}, \mathbf{r})$ for some minimal regimes $\mathbf{r}$. Also, define

$$T^{min,\epsilon}(\mathbf{y}, \lambda, \mathbf{a})(s) = \min\left\{x^{APS}(\mathbf{a}(s), \lambda), (1-\delta)\lambda \cdot g(\mathbf{a}(s)) + \delta \sum_{s' \in S} \mathbf{y}(s')\pi(s'|\mathbf{a}(s))\right\} - \epsilon.$$

Lemma 3 (Operator $T^{min,\epsilon}$)**.** *Fix* $\epsilon > 0$*,* λ*, and* $\mathbf{a} \in \mathbf{A}(\mathbf{W})$*. As a function of* $\mathbf{y} : S \to \mathbb{R}$*,* $T^{min,\epsilon}$ *is*

(L3.i) *increasing;*

(L3.ii) *a contraction with modulus* δ*, and hence has a unique fixed point* $\mathbf{y}^*$*;*

(L3.iii) *if* $T^{min,\epsilon}(\mathbf{y}) \leq (\geq)\mathbf{y}$ *then* $\mathbf{y}^* \leq (\geq) T^{min,\epsilon}(\mathbf{y})$*;*

Proof of Lemma 3. The proof follows verbatim that of Lemma 3, replacing T^{min} with $T^{min,\epsilon}$. □

We further define

$$x^{R,\epsilon}(a, \lambda, \mathbf{a}) = (1-\delta)\lambda \cdot g(a) + \delta \sum_{s' \in S} \pi(s'|a) x^{\epsilon}(s', \lambda, \mathbf{a}, \mathbf{r}),$$

where $\mathbf{r}$ is minimal for $\mathbf{a}$ and λ.

Lemma 4 (Maximal actions)**.** *Suppose that $\mathbf{A}(\mathbf{W})$ is non-empty valued. For all $\epsilon > 0$ and λ,*

(L4.i) *there exist maximal actions;*

(L4.ii) *$\mathbf{a} \in \mathbf{A}(\mathbf{W})$ is maximal if and only if for all $s \in S$ and $a \in \mathbf{A}(\mathbf{W})(s)$,*

$$x^{\epsilon}(s, \lambda, \mathbf{a}) \geq \min\left\{x^{APS}(a, \lambda), x^{R,\epsilon}(a, \lambda, \mathbf{a})\right\} - \epsilon, \tag{8}$$

with equality when $a = \mathbf{a}(s)$;

(L4.iii) *if (8) is violated for some $s \in S$ and $a \in \mathbf{A}(\mathbf{W})(s)$, then $\mathbf{a}$ is not maximal. Indeed, for all $s' \in S$, $x^{\epsilon}(s', \lambda, \mathbf{a} \setminus (s, a)) \geq x^{\epsilon}(s', \lambda, \mathbf{a})$, with strict inequality in state s.*

Proof of Lemma 4. Once again, this follows verbatim the proof of Lemma 4, replacing x with x^{ϵ}, $T^{\min}$ with $T^{\min,\epsilon}$, references to (7) with references to (8), and references to Lemma 3 with references to Lemma 3. □

E.3 Sufficiency of binding payoffs

Lemma 5. *For any direction λ, if B^{ϵ} sub-generates at $\mathbf{W}$ in the direction λ, then for any $\mathbf{a} \in \mathbf{A}(\mathbf{W})$, if $\gamma(\mathbf{a}(s), \lambda, \mathbf{W}) > 0$, then*

$$x^{APS}(\mathbf{a}(s), \lambda, \mathbf{W}) - \epsilon \geq x^{R,\epsilon}(\mathbf{a}(s), \lambda, \mathbf{a}, \mathbf{W}) - \epsilon = x^{\epsilon}(s, \lambda, \mathbf{a}, \mathbf{W})$$

Moreover, there exist minimal regimes such that $\mathbf{r}(s) = R$ for s with $\gamma(\mathbf{a}(s), \lambda, \mathbf{W}) > 0$.

Proof of Lemma 5. Suppose that $\gamma(\mathbf{a}(s), \lambda, \mathbf{W}) > 0$. Then the best continuation values from $\mathbf{W}$ in the direction λ, denoted $\mathbf{w}$, must be incentive compatible for $\mathbf{a}(s)$, and

$$x^{APS}(\mathbf{a}(s), \lambda, \mathbf{W}) = (1-\delta)\lambda \cdot g(\mathbf{a}(s)) + \delta \sum_{s' \in S} \pi(s'|\mathbf{a}(s))\lambda \cdot \mathbf{w}(s').$$

Sub-generation and the definition of x^{ϵ} imply that $\lambda \cdot \mathbf{w}(s') \geq x^{APS}(\mathbf{a}(s'), \lambda, \mathbf{W}) - \epsilon \geq x^{\epsilon}(s', \lambda, \mathbf{W})$. Hence,

$$\begin{aligned} x^{APS}(\mathbf{a}(s), \lambda, \mathbf{W}) - \epsilon &\geq (1-\delta)\lambda \cdot g(\mathbf{a}(s)) + \delta \sum_{s' \in S} \pi(s'|\mathbf{a}(s)) x^{\epsilon}(s', \lambda, \mathbf{W}) \\ &\geq x^{R,\epsilon}(\mathbf{a}(s), \lambda, \mathbf{a}, \mathbf{W}) \end{aligned}$$

as desired.

Finally, suppose $\mathbf{r}$ is minimal and $\gamma(\mathbf{a}(s), \lambda, \mathbf{W}) > 0$. If $x^{APS}(\mathbf{a}(s), \lambda, \mathbf{W}) > x^{R,\epsilon}(\mathbf{a}(s), \lambda, \mathbf{a}, \mathbf{W})$, then $\mathbf{r}(s)$ is necessarily R. Otherwise, (9) implies that $x^{APS}(\mathbf{a}(s), \lambda, \mathbf{W}) = x^{R,\epsilon}(\mathbf{a}(s), \lambda, \mathbf{a}, \mathbf{W})$. Thus, if we set $\mathbf{r}'(s) = R$ for all states with $\gamma(\mathbf{a}(s), \lambda, \mathbf{W}) > 0$ and $\mathbf{r}'(s') = \mathbf{r}(s')$ otherwise, then $x^\epsilon(\cdot, \lambda, \mathbf{a}, \mathbf{r}, \mathbf{W})$ is clearly a fixed point of $T^\epsilon(\cdot, \lambda, \mathbf{a}, \mathbf{r}', \mathbf{W})$, so that $\mathbf{r}'$ also satisfies (6) and is minimal. □

Lemma 6. *If $\widetilde{B}^\epsilon$ sub-generates at $\mathbf{W}$, then B^ϵ sub-generates at $\widetilde{B}^\epsilon(\mathbf{W})$.*

Proof of Lemma 6. Towards a contradiction, suppose that some action profile $a \in \mathbf{A}(\mathbf{W})(s)$, with continuation values $\mathbf{w} \in \widetilde{B}^\epsilon(\mathbf{W})$, generates a payoff outside the convex set $\widetilde{B}^\epsilon(\mathbf{W})$. Then for some direction λ, $x^{APS}(a, \lambda, \widetilde{B}^\epsilon(\mathbf{W})) - \epsilon > x^\epsilon(s, \lambda, \mathbf{W})$, so

$$\begin{aligned} x^\epsilon(s, \lambda, \mathbf{W}) + \epsilon < x^{APS}(a, \lambda, \widetilde{B}(\mathbf{W})) &= \lambda \cdot \left((1-\delta)g(a) + \delta \sum_{s' \in S} \pi(s'|a)\mathbf{w}(s')\right) \\ &\leq (1-\delta)\lambda \cdot g(a) + \delta \sum_{s' \in S} \pi(s'|a) x^\epsilon(s', \lambda, \mathbf{W}), \end{aligned}$$

where the last inequality holds because $\lambda \cdot \mathbf{w}(s') \leq x^\epsilon(s', \lambda, \mathbf{W})$, since $\mathbf{w}(s') \in \widetilde{B}^\epsilon(\mathbf{W})(s')$. The right-hand side of this inequality equals $x^{R,\epsilon}(a, \lambda, \mathbf{a}, \mathbf{W})$ for any $a \in \mathbf{A}(\mathbf{W})(s)$ that is maximal in the direction λ (given $\mathbf{W}$). Since $\widetilde{B}^\epsilon(\mathbf{W}) \subseteq \mathbf{W}$, we know that $x^{APS}(s, \lambda, \mathbf{W})$ is greater than $x^\epsilon(s, \lambda, \mathbf{W})$ as well. That is, $x^\epsilon(s, \lambda, \mathbf{a}, \mathbf{W}) < \min\{x^{APS}(a, \lambda, \mathbf{W}), x^{R,\epsilon}(a, \lambda, \mathbf{a}, \mathbf{W})\} - \epsilon$, contradicting (L4.ii). □

Proposition 1 (Sufficiency of binding payoffs)**.** *As long as B^ϵ sub-generates at $\widetilde{\mathbf{W}}^0$, then for any $k \geq 0$, B^ϵ sub-generates at $\widetilde{\mathbf{W}}^k$. As a result, for any λ and $\mathbf{a} \in \mathbf{A}(\mathbf{W})$, if $\gamma(a, \lambda, \widetilde{\mathbf{W}}^k) > 0$ is strictly positive, then $\mathbf{r}^*(s) = R$.*

Proof of Proposition 1. Follows verbatim the proof of Proposition 1, replacing B and $\widetilde{B}$ with B^ϵ and $\widetilde{B}^\epsilon$, respectively. □

Finally, we extend the characterizations of optimal policies and optimal pairs.

Lemma 7. *If B^ϵ sub-generates at $\mathbf{W}$ in the direction λ, the actions $\mathbf{a} \in \mathbf{A}(\mathbf{W})$ are maximal if and only if for all (s, a),*

$$x^\epsilon(s, \lambda, \mathbf{a}) \geq \begin{cases} \min\left\{\widehat{x}^{APS}(a, \lambda), x^{R,\epsilon}(a, \lambda, \mathbf{a})\right\} - \epsilon & \text{if } \gamma(a, \lambda, \mathbf{W}) = 0; \\ x^{R,\epsilon}(a, \lambda, \mathbf{a}) - \epsilon & \text{if } \gamma(a, \lambda, \mathbf{W}) > 0. \end{cases}$$

Note that Lemma 10 implies, via the same argument in Corollary 1, that $\mathbf{V}^\epsilon$ has at most $\overline{L}$ extreme points. Moreover, we can adapt the algorithm in Section 4 to compute $\widetilde{B}^\epsilon(\mathbf{W})$. It is still the case that direction where robustly optimal payoffs $\mathbf{u}$ cease to be optimal corresponds to a substitution (s, a, p). When $p = R$, the change must occur at a direction λ' such that

$$\lambda' \cdot (u^R(a, \mathbf{u}) - \mathbf{u}(s)) = \sum_{s' \in S} \pi(s'|a)\eta(s', \mathbf{a}, \mathbf{r}) + \epsilon - \eta(s, \mathbf{a}, \mathbf{r}), \tag{9}$$

so that the change in level is exactly offset by a change in penalty, and otherwise

$$\lambda' \cdot (p - \mathbf{u}(s)) = \epsilon - \eta(s, \mathbf{a}, \mathbf{r}), \tag{10}$$

where $\mathbf{r}$ are the regimes associated with the incumbent optimal pair that induces $\mathbf{u}$. There are at most $2\overline{L}\,\overline{M}$ directions that satisfy (9) or (10). Such a direction is again called legitimate if $(\mathbf{a}, \mathbf{p})$ is optimal in that direction. We can therefore compute $\widetilde{B}^\epsilon$ by finding the optimal pair in one direction, then iteratively computing the legitimate substitution direction with the smallest clockwise rotation, and then lexicographically optimizing the pair in the new direction. This produces a sequence of directions and optimal payoffs $\{(\lambda^k, \mathbf{u}^k)\}_{k=0}^K$.

Note that a subtle issue is that the new optimal level $x^\epsilon(s, \lambda, \mathbf{W})$ is no longer piecewise linear, but is piecewise affine of the form $\lambda \cdot u - \eta$, where $\eta > 0$. Because of the sign of the constant, it turns out that directions at which the optimal pair does not change are still redundant. In particular, if we have a clockwise sequence of directions λ, λ', and λ'' at which u are the optimal payoffs in state s and η is the optimal penalty, then

$$\begin{aligned}
\lambda' \cdot v &= \frac{\alpha\lambda + (1-\alpha)\lambda''}{\|\alpha\lambda + (1-\alpha)\lambda''\|} \cdot v \\
&\leq \frac{1}{\|\alpha\lambda + (1-\alpha)\lambda''\|} \left[\alpha(\lambda \cdot u - \eta) + (1-\alpha)(\lambda'' \cdot u - \eta)\right] \\
&= \lambda' \cdot u - \frac{1}{\|\alpha\lambda + (1-\alpha)\lambda''\|}\eta \\
&\leq \lambda' \cdot u - \eta,
\end{aligned}$$

since $\|\alpha\lambda + (1-\alpha)\lambda''\| \leq 1$. As a result, we can simply intersect the half-spaces at legitimate test directions to compute $\widetilde{B}^\epsilon$. We therefore have:

Theorem 2. *Suppose that $N = 2$, $\mathbf{A}(\mathbf{W})$ is non-empty valued, and B^ϵ sub-generates at $\mathbf{W}$. Then the previously described procedure terminates in at most $2\overline{L}\,\overline{M}$ substitutions and runtime $O(\overline{L}\,\overline{M}^2)$. If there are no legitimate test directions at $\mathbf{u}^0$, then $\widetilde{B}(\mathbf{W})(s) = \{\mathbf{u}^0(s)\}$*

for all s. Otherwise,

$$\widetilde{B}^{\epsilon}(\mathbf{W})(s) = \{v | \lambda^k \cdot v \leq \lambda^k \cdot \mathbf{u}^k(s) \; \forall k = 1, \ldots, K\}. \tag{11}$$

References

Blackwell, D. (1962): "Discrete dynamic programming," *The Annals of Mathematical Statistics*, 719–726.

Dantzig, G. B. and M. N. Thapa (2006): *Linear programming 1: introduction*, Springer Science & Business Media.