Computing Nash equilibria and best response strategies in two-person zero-sum sequential games

Elvis D. DOHMATOB

Abstract

In this manuscript, we consider the problem of computing a best response against an opponent's realization plan in two-person sequential games. The proposed algorithm for solving the corresponding contrained convex-optimization problem, derives from the primal-dual scheme of A. Chambolle and T. Pock. Our algorithm is simple: all resolvent operators can be effectively computed in closed-form, using only elementary algebraic operations.

Index Terms

game theory; two-person sequential game; Nash equilibria; best response; convex-optimization; contraint; primal-dual scheme

I. Introduction

A. Notation and Terminology

To begin, let us introduce some technical notation and terminology we will be using in this paper. The reader should lookup any standard textbook (for example [1]) on convex optimization for a tutorial introduction to these notions. Viz,

- $\mathcal{P}(S)$: set of all the subsets of S;
- $\mathcal{P}_k(S)$: set of all the subsets of S which have exactly k elements;
- $\mathbb{R}^{m \times n}$: space of all m-by-p real matrices;
- $E^{-1}(U)$: pre-image of $U \subset \mathbb{R}^n$ under a matrix $E \in \mathbb{R}^{p,n}$, namely the set $\{x \in \mathbb{R}^n \mid Ex \in U\}$;
- A^T: transpose of a matrix A;
- $\begin{bmatrix} A \\ B \end{bmatrix}$: vertical stacking of two matrices A and B;
- v_j: jth component of a vector v;
- $u^T v$: dot product of two vectors u and v;
- $\mathbb{R}^n_+$: the n-dimensional nonnegative orthant;
- $(x)_+$: component-wise maximum of a vector x and 0;
- i_C: indicator function of a convex set C;
- Π_C: euclidean projector onto a convex set C;
- $\|K\|_2$: matrix 2-norm of a matrix K;
- *l.s.c.p.c*: acronym for adjective *lower semi-continuous proper convex*;
- f^*: Fenchel transform (a.k.a convex conjugate) of a *l.s.c.p.c* function f;
- $(1 + \sigma \partial f)^{-1}$: resolvent (a.k.a proximal) operator of a *l.s.c.p.c* function f, for a given stepsize $\sigma > 0$

B. Generating the Texas Hold'em Poker game tree

Let $a := (a_1, a_2, a_3, a_4) \in \mathbb{R} \times \mathbb{R} \times \mathbb{R}^2 \times \mathbb{R}^2$ hold the accounting info, where:

- $a_1 \in \mathbb{R}$ is the running bet size,
- $a_2 \in \mathbb{R}$ is the running total amount in the pot,
- $a_3 \in \mathbb{R}^2$ is a vector showing what each player has put into the pot sofar, and
- $a_4 \in \mathbb{R}^2$ is a vector showing running capital of each player.

Let $ok(a, L)$ be a flag which is true iff the bet size a_1 has not exceeded the limit L and each player's running credit is nonnegative. For any round $t \in \mathbb{N}$, let $p(t)$ be the player to begin and $\Sigma(t)$ be the set of all signals that can be emitted by the chance player.

Definition 1. *A signal is simply any part of the information needed at shutdown to score the player's hands.*

Remark 1. *An emitted signal can be fully observable (for example, community cards), or only partially observable (for example when private / hole / pocket cards are delt, players can only see their own hole cards given to them; they can't see another player's hole cards).*

$$\gamma(t, T, a, L) = \begin{cases} \mathcal{P}_1(\Sigma(t)), & \textit{if } (t < T) \wedge ok(a, L) \wedge (p(t) = 0); \\ \{\boldsymbol{F}_{p(t)}\} \cup \boldsymbol{C}_{p(t)} \gamma(t+1, T, a, L) \cup \boldsymbol{K}_{p(t)} \lambda(t, T, g(a, \boldsymbol{K}_{p(t)}), L), & \textit{if } (t < T) \wedge ok(a, L) \wedge (p(t) > 0); \\ \emptyset, & \textit{if } t \geq T \vee \neg ok(a, L). \end{cases} \tag{1}$$

where the auxiliary function λ is defined by:

$$\left.\begin{aligned} \lambda(t, T, a, L) &:= \cup_{n=0}^{\infty} \alpha_1^{(n)}(t, T, a, L) \cup \cup_{n=0}^{\infty} \alpha_2^{(n)}(t, T, a, L) \\ \alpha_1^{(n)}(t, T, a, L) &:= (\boldsymbol{R}_{p(t)'} \boldsymbol{R}_{p(t)})^n \mathcal{P}_1(\{\boldsymbol{F}_{p(t)'}, \boldsymbol{K}_{p(t)'}\}) \gamma(t+1, T, g(a, (\boldsymbol{R}_{p(t)'} \boldsymbol{R}_{p(t)})^n \mathcal{P}_1(\{\boldsymbol{F}_{p(t)'}, \boldsymbol{K}_{p(t)'}\})), L) \\ \alpha_2^{(n)}(t, T, a, L) &:= (\boldsymbol{R}_{p(t)'} \boldsymbol{R}_{p(t)})^n \boldsymbol{R}_{p(t)'} \mathcal{P}_1(\{\boldsymbol{F}_{p(t)}, \boldsymbol{K}_{p(t)}\}) \gamma(t+1, T, g(a, (\boldsymbol{R}_{p(t)'} \boldsymbol{R}_{p(t)})^n \boldsymbol{R}_{p(t)'} \mathcal{P}_1(\{\boldsymbol{F}_{p(t)}, \boldsymbol{K}_{p(t)}\})), L) \end{aligned}\right\} \tag{2}$$

C. Statement of the problem

Consider a two-person sequential game[1] in sequential-form (See for example, [2] and [3] for theory on sequential-form representation), and let $A \in \mathbb{R}^{m,n}$ be our payoff matrix. We will be referring to the other player as "the opponent". We are interested in the problem of finding a best response strategy $x^* \in \mathbb{R}^n_+ \cap E^{-1}(\{e\})$, given a fixed behavioral strategy y_0 for the opponent, where E is a sparse p-by-n matrix whose entries are -1, 0, or $+1$, and $(1, 0, 0, ..., 0) =: e \in \mathbb{R}^p$. Recall that E and e encode linear constraints on our "admissible" realization plans[2] x, in the sequential form representation of the game. In the language of convex-optimization, one can readily give the following saddle-point formulation for this problem. Viz,

$$\underset{x \in \mathbb{R}^n_+ \cap E^{-1}(\{e\})}{min} \quad -y_0^T A x \tag{3}$$

Definition 2. *A solution x^* to problem* (3) *is called a best response strategy against the opponent's fixed strategy y_0.*

Remark 2. *Of course y_0 is not necessarily an optimal strategy for the opponent. In case it is, the pair (x^*, y_0) is a Nash equilibrium for the game.*

Remark 3. *In practice, A, E, and F are very sparse. This sparsity should be thoroughly exploited[3] by a solver for problem* (3)*.*

In section II, we give a brief overview of existing methods for solving (3)*. We elaborate our proposed algorithm in section III.*

II. Related work

Pending...

III. The proposed algorithm

In this section we present the algorithm which is the purpose of this paper, namely an algorithm for solving (3)*. Our algorithm (Alg.1) is a use-case of the generic primal-dual algorithm of A. Chambolle and T. Pock, namely Algorithm 1 of [4].*

A. Derivation of the algorithm

First observe that (1) can be re-written in the form:

$$\underset{x \in \mathbb{R}^n}{min} \; f(-Ax) + g(x) \tag{4}$$

where:

$$\left.\begin{aligned} g &:= i_{E^{-1}(\{e\})} + i_{\mathbb{R}^m_+} \\ f &: z \mapsto y_0^T z \end{aligned}\right\} \tag{5}$$

Finally, the primal-dual formulation of this problem (which can be easily obtained using Fenchel-Rockafellar duality, for example) is:

$$\underset{x \in \mathbb{R}^n}{min} \; \underset{y \in \mathbb{R}^m}{max} \; y^T(-A)x_0 + g(x) - f^*(y) \tag{6}$$

Now, let's observe that $i_{E^{-1}(\{e\})}(x) = \underset{\zeta \in \mathbb{R}^p}{max} \; \zeta^T(e - Ex)$, and so one can re-write (2) as:

$$\underset{x \in \mathbb{R}^n}{min} \; \underset{y \in \mathbb{R}^m, \zeta \in \mathbb{R}^p}{max} \; \begin{bmatrix} y \\ \zeta \end{bmatrix}^T Kx + G(x) - F^*(y, \zeta) \tag{7}$$

where we've defined:

$$\left.\begin{aligned} K &:= -\begin{bmatrix} A \\ E \end{bmatrix} \\ G &:= g - i_{E^{-1}(\{b\})} = i_{\mathbb{R}^n_+} \\ F^* &: (y, \zeta) \mapsto (f^*(y) - \zeta^T e) = (i_{\{y_0\}}(y) - e^T\zeta) \end{aligned}\right\} \tag{8}$$

Furthermore, one easily checks that F^* and G are l.s.c.p.c with resolvent (a.k.a proximity) operators given by the simple formulae:

$$\left.\begin{aligned} (1 + \tau\partial G)^{-1} &= \Pi_{\mathbb{R}^n_+} : x \mapsto (x)_+ \\ (1 + \sigma\partial F^*)^{-1} &: (y, \zeta) \mapsto (y_0, \zeta + \sigma e) \end{aligned}\right\} \tag{9}$$

Using these ingredients, we derive the primal-dual algorithm given in Alg.1, for solving (7)*, and thus* (3)*.*

Remark 4. *Equation* (7) *is in the form of equation (2) in [4] if we take $X := \mathbb{R}^n$ and $Y := \mathbb{R}^m \times \mathbb{R}^p$.*

Remark 5. *Neither G nor F^* is strongly convex and so Alg.1 cannot be accelerated in the sense of algorithm 39 of [4].*

[1] In a sequential game, players take turns in play, one after the other, as opposed to simultaneous play.
[2] In games in sequential form, the terms "strategy" and "realization plan" mean thesame thing.
[3] For example when multiplying vectors with these matrices.

Algorithm 1: Primal-dual algorithm for computing best response against opponent's fixed realization plan y_0

Given Tolerance $\epsilon > 0$.
Initialize $\tilde{x^{(0)}} = x^{(0)} \in \mathbb{R}^n$; $\zeta^{(0)} \in \mathbb{R}^q$; $\tau, \sigma > 0$ s.t. $\tau\sigma\|K\|_2^2 < 1$; $k = 0$.
Precompute $\eta_0 \leftarrow \tau A^T y_0$
repeat

$$\begin{aligned} \zeta^{(k+1)} &\leftarrow \zeta^{(k)} - \sigma\big(e - E\tilde{x}^{(k)}\big) \\ x^{(k+1)} &\leftarrow \big(x^{(k)} + \eta_0 + E^T\zeta^{(k+1)}\big)_+ \\ \tilde{x}^{(k+1)} &\leftarrow 2x^{(k+1)} - x^{(k)} \\ k &\leftarrow k+1 \end{aligned}$$

until $\frac{\|x^{(k+1)}-x^{(k)}\|_2^2}{2\tau} + \frac{\|\zeta^{(k+1)}-\zeta^{(k)}\|_2^2}{2\sigma} < \epsilon$;
return $x^{(k)}$, $-y_0^T A x^{(k)}$

B. Convergence analysis of the algorithm

Pending...

Remark 6. *The derivation above reveals that in equation* (3) *above, if the constraint "$x \geq 0$" is replaced by a constraint "$x \in C$" (thus obtaining a new problem) where C is a convex set , then we simply need to replace the operator "$(.)_+$" with "Π_C" in the equations to obtain a corresponding algorithm. Of course, this is because $i_C^* = \Pi_C$.*

Acknowledgments: *Pending...*

REFERENCES

[1] *S. Boyd and L. Vandenberghe,* Convex Optimization. *Cambridge University Press, 2004.*
[2] *B. Von Stengel, "Efficient computation of behavior strategies,"* Games and Economic Behavior*, vol. 14, no. 2, pp. 220–246, 1996.*
[3] *D. Koller, N. Megiddo, and B. Von Stengel, "Efficient computation of equilibria for extensive two-person games,"* Games and Economic Behavior*, vol. 14, no. 2, pp. 247–259, 1996.*
[4] *A. Chambolle and T. Pock, "A first-order primal-dual algorithm for convex problems with applications to imaging,"* J. Math. Imaging Vis.*, vol. 40, p. 120, 2011.*