

Randomized Sampling for Large Zero-Sum Games*

Shaunak D. Bopardikar[†] Alessandro Borri João Hespanha Maria Prandini
Maria D. Di Benedetto

September 1, 2010

Keywords: Game theory, Randomized algorithms, Zero-Sum Games, Optimization

Abstract

This paper addresses the solution of large zero-sum matrix games using randomized methods. We formalize a procedure, termed as *sampled saddle point (SSP)*, by which a player can compute mixed policies that, with a high probability, are security policies against an adversary playing the same game and who is also using the SSP procedure. The computational savings result from solving stochastically sampled subgames that are much smaller than the original game. We provide two methodologies and determine how large the subgames should be to guarantee the desired high probability. The first methodology provides a game-independent bound on the size of the subgames that can be computed a priori. The second methodology is useful when computation limitations prevent a player from satisfying the first game-independent bound and provides a high-probability bound on how much the outcome of the game can violate the precomputed security level. We also analyze the effect of mismatch between the distributions used by the two players to obtain their respective subgames using the SSP procedure, and extend the previous bounds based on the distributions used by the players. Finally, we demonstrate the usefulness of these results in solving a hide-and-seek game that is known to exhibit exponential complexity.

1 Introduction

While a large number of robust design problems can be formulated as zero-sum matrix games, in practice, such games lead to extremely large — often infinite — matrices. This is the case in combinatorial problems, where the decision makers are faced with a number of possible options that increases exponentially with the size of the problem; for example, in path planning problems where the number of paths increases combinatorially with the number of points (cf. [4]). Large zero-sum matrix games also arise in partial information feedback games (cf. [10]) wherein optimal strategies

*This material is based upon work supported in part by ARO MURI Grant number W911NF0910553, and in part by the Center of Excellence for Research DEWS, University of L’Aquila, Italy. A preliminary version of this work entitled “Randomized Sampling for Large Zero-Sum Games” will be presented at the 2010 IEEE Conference on Decision and Control, Atlanta, GA, USA.

[†]Shaunak D. Bopardikar and João Hespanha are with the Department of Electrical and Computer Engineering, University of California at Santa Barbara, CA, USA, {sdbopardikar,hespanha}@ece.ucsb.edu. Alessandro Borri and Maria D. Di Benedetto are with the Department of Electrical Engineering, University of L’Aquila, Italy {alessandro.borri,mariadomenica.dibenedetto}@univaq.it. Maria Prandini is with the Dipartimento di Elettronica e Informazione of the Politecnico di Milano, prandini@elet.polimi.it.

are functions of the players' actions and thus the number of strategies grows exponentially with the size of the players' action spaces.

Inspired by the use of randomized approaches to solve optimization problems, we consider an approach to solve very large zero-sum matrix games by using randomized sampling. Each player reduces her search space by taking a random sample of the available actions to construct a much smaller version of the original game. Players then solve these smaller games and utilize the saddle-point policies so obtained against each other. We call this procedure the *sampled saddle-point* (SSP) algorithm. Since each player only considers a small submatrix of the original game, the two players typically consider very different submatrices. Therefore, the saddle-point policies obtained by this process will generally not be security policies for the whole game. This means that a player can obtain an outcome that is strictly worse than the value computed based on her submatrix. However, we show that this happens with low probability if the size of the submatrix is sufficiently large.

In this framework, a reasonable notion of security policy for a player is that the outcome of the game should not unpleasantly surprise the player with high probability. In particular, one wants the outcome of the game not to be much larger than what the minimizer expects or much smaller than what the maximizer expects, based on the computation of the value of her submatrix. In this paper, we analyze this sampling procedure for zero-sum games and provide conditions under which it leads to a security policy with high probability.

Related Work

Two-player zero-sum matrix games have been studied extensively over the past decades (cf. textbook by [3]). The classical Mini-Max theorem (cf. [19]) guarantees the existence of an optimal pair of strategies for the two players, each of which is a security policy for the corresponding player. However, when the matrix is of large size, the computation of the optimal strategies involves solving optimization problems with a large number of variables and constraints.

A probabilistic approach has proven to be computationally efficient in evaluating games with large sizes. Using probabilistic analysis, the existence of simple, near-optimal strategies over a subset with logarithmically smaller size of the original matrix game was established in [12]. A popular method to solve win-lose type of multi-stage or dynamic games is to evaluate the root of a game tree, in which every node is alternately an *AND* and an *OR* operation, while the leaves have a value of either 0 or 1. [13] present randomized algorithms to evaluate such game trees more efficiently than using deterministic algorithms.

Randomized methods have also been successful in providing efficient solutions to complex control design problems with probabilistic guarantees. [11] adopt a probabilistic approach to show the existence of randomized algorithms with polynomial complexity to solve complex robust stability analysis problems. [14] propose a randomized method to determine the minimum number of samples that provide a probabilistic guarantee of the level of worst-case controller performance. In [16, 17, 18], the authors demonstrate the use of randomized algorithms in statistical learning theory to solve control design problems and a number of well known complex problems in matrix theory. In [6, 7, 9], the authors introduce the *scenario approach* to solve convex optimization problems with an infinite number of constraints, and discuss possible applications of the approach to systems and control. The results in these papers are instrumental to establish several of the results of this paper.

Statistical learning theory (cf. [15]) provides a framework to address optimization in infinite dimensions. Using these tools, [1] consider semi-infinite optimization problems under uncertainty in which the objective function is possibly non-convex. In [2], the authors provide an improvement in sample-size bounds and also over the bounds on the scenario approach for convex optimization. We use these results to provide bounds on the size of the subgames for probabilistic security.

Contributions

The contributions of this paper are four-fold. First, based on results from the scenario approach in [6, 7, 2], we show that when the sizes of the subgames solved by each player are sufficiently large, the SSP algorithm provides security policies for both players with some pre-specified high probability $1 - \delta$. When the subgames are chosen by the players from identical distributions, the bounds on the sizes of the subgames are *game independent* and are computable a-priori. Not surprisingly, they grow with the desired confidence level $1 - \delta$. However, they are *independent of the size of the original matrix game*, which could, in fact, be even infinite and not even have a value.

Second, we propose a methodology that provides an a-posteriori, high-probability bound on the deviation of the game outcome from the pre-computed security level. In particular, regardless of the size of the subgames solved by each player, we provide a high probability bound on how much a player can expect the outcome of the game to violate the security value computed based on the submatrix used to determine her saddle-point equilibrium. This bound is computed after a player selects and solves her subgame.

Third, we consider a mismatch between the distributions used by the two players to sample their submatrices. We extend the previous bounds to obtain general bounds that depend on the mismatch in the distributions used by the players. These general bounds turn out to be non-conservative when the mismatch is associated with policy domination in the matrix game. For example, suppose the minimizer finds that a set of columns X would always lead to a larger outcome than the set of columns Y , then the minimizer secures herself by sampling policies out of X instead of Y . We show that the bounds derived in the two methodologies also provide the same probabilistic guarantees in the case of policy domination.

Fourth and finally, we apply the methodologies to solve a hide-and-seek game, in which one player hides a treasure in one of N points and the other player searches for the treasure by visiting each of the points. This is formalized as a zero-sum game in which the player that hides the treasure wants to maximize the distance that the other player needs to travel until the treasure is found. To determine the optimal strategy for this game, it is required to solve a matrix game whose size is $N \times N!$. Thus, exact solutions to this problem require computation that scales exponentially with the number of points N . Our approach is *independent of the size of the game* and therefore the size of the matrix plays no role in the amount of computation required by it. This is possible because each player concentrates on a subset of her action set, and probabilistic guarantees rather than deterministic guarantees on the quality of the solution with respect to the actual game outcome are provided.

As compared to the preliminary conference version (cf. [5]), this paper presents new results on mismatch in the distributions used by the players to construct the subgames. Other differences include usage of sample size bounds from [2], which provide an improvement on the sizes of the subgames as compared to those in the conference version.

Organization

This paper is organized as follows. The problem formulation and the sampled saddle-point algorithm are presented in Section 2. Bounds on probabilistic guarantees when the two players use identical distributions to sample the matrix, are established in Section 3. These bounds are extended in Section 4 to allow mismatch between the distributions. Finally, we demonstrate the procedure applied to the hide-and-seek problem in Section 5.

2 Sampled Saddle-Point Algorithm

Consider a zero-sum matrix game defined by an $M \times N$ matrix A , in which player P_1 is the minimizer and selects rows and player P_2 is the maximizer and selects columns. We are interested in problems for which the matrix A is too large to permit the computation of mixed saddle-points and therefore the players are forced to consider only submatrices of A to select their policies. This scenario motivates the following procedure, which we henceforth call the *sampled saddle-point (SSP) algorithm*.

1. Each player P_k , $k \in \{1, 2\}$ randomly selects m_k rows and n_k columns of A , which she uses to construct a $m_k \times n_k$ submatrix A_k of A . Denoting by $\mathcal{B}^{k \times \ell}$ the set of $k \times \ell$ left-stochastic $(0, 1)$ -matrices (i.e., matrices whose entries belong to the set $\{0, 1\}$ and whose columns add up to one), we can express the process of constructing each submatrix A_k by randomly selecting two random matrices $\Gamma_k \in \mathcal{B}^{M \times m_k}$ and $\Pi_k \in \mathcal{B}^{N \times n_k}$ and then computing the product:

$$A_k = \Gamma_k' A \Pi_k.$$

2. Each player P_k , $k \in \{1, 2\}$ computes the mixed security value and the corresponding security policy for her submatrix A_k :

$$\begin{aligned} \bar{V}(A_1) &= \max_{z \in \mathcal{S}_{n_1}} y_1^{*'} A_1 z = \min_{y \in \mathcal{S}_{m_1}} \max_{z \in \mathcal{S}_{n_1}} y' A_1 z \\ \underline{V}(A_2) &= \min_{y \in \mathcal{S}_{m_2}} y' A_2 z_2^* = \max_{z \in \mathcal{S}_{n_2}} \min_{y \in \mathcal{S}_{m_2}} y' A_2 z \end{aligned}$$

where y_1^* (resp. z_2^*) is a mixed security policy for P_1 (resp. P_2) in the submatrix game A_1 (resp. A_2). \mathcal{S}_{m_k} and \mathcal{S}_{n_k} denote the probability simplexes of appropriate dimensions. We call $\bar{V}(A_1)$ and $\underline{V}(A_2)$ the *sampled security values of the game* for P_1 and P_2 , respectively.

3. Player P_1 selects a row according to the distribution y_1^* , whereas P_2 selects a column according to the distribution z_2^* , which correspond to the following policies for the original game: $y^* := \Gamma_1 y_1^*$, $z^* := \Pi_2 z_2^*$ and the following game outcome:

$$y^{*'} A z^* = y_1^{*'} \Gamma_1' A \Pi_2 z_2^*.$$

We call y^* and z^* the *sampled security policies* for players P_1 and P_2 , respectively.

We say that *the SSP algorithm is ϵ -secure for player P_1 with confidence $1 - \delta$* if

$$P_{\Gamma_1, \Pi_1, \Gamma_2, \Pi_2} (y^{*'} A z^* \leq \bar{V}(A_1) + \epsilon) \geq 1 - \delta. \quad (1)$$

Here and in the sequel, we use a subscript in the probability measure P to emphasize which random variables define the event that is being measured. In essence, condition (1) states that the

probability that the outcome of the game will violate P_1 's sampled security value by more than ϵ is smaller than δ . Similarly, we say that *the SSP algorithm is ϵ -secure for player P_2 with confidence $1 - \delta$* if

$$P_{\Gamma_1, \Pi_1, \Gamma_2, \Pi_2} (y^{*'} Az^* \geq \underline{V}(A_2) - \epsilon) \geq 1 - \delta. \quad (2)$$

The previous definitions guarantee that the two players will be surprised with low probability when playing with policies obtained from a one-shot solution to the SSP algorithm. However, no specific guarantee is given to the inherent safety of the policies/values obtained using this algorithm. So, e.g., suppose that player P_1 computes y^* once using the SSP algorithm and then plays this policy multiple times against a sequence of policies z^* that P_2 obtained by running the SSP algorithm multiple times, P_1 could conceivably be surprised many more times that one would expect for a low value of δ . This would happen if she was “unlucky” and got a particular (low probability) y^* that is particularly bad or a value $\bar{V}(A_1)$ that is particularly optimistic. To avoid this scenario, we introduce notions of security that refer to specific policies/values: We say that *a policy y^* with value $\bar{V}(A_1)$ is ϵ -secure for player P_1 with confidence $1 - \delta$* if

$$P_{\Gamma_2, \Pi_2} (y^{*'} Az^* \leq \bar{V}(A_1) + \epsilon \mid y^*, \bar{V}(A_1)) \geq 1 - \delta \quad (3)$$

and that *a policy z^* with value $\underline{V}(A_2)$ is ϵ -secure for player P_2 with confidence $1 - \delta$* if

$$P_{\Gamma_1, \Pi_1} (y^{*'} Az^* \geq \underline{V}(A_2) - \epsilon \mid z^*, \underline{V}(A_1)) \geq 1 - \delta. \quad (4)$$

Note that the subscripts in the probability measure now only include the matrices corresponding to random extractions by the other player, since the probability guarantees are given for specific policies and values of one player.

So far, we have not specified the joint distribution of the row/column extraction matrices $\Gamma_1, \Gamma_2, \Pi_1, \Pi_2$, but this distribution will clearly affect the outcomes of the algorithm. In the context of noncooperative games, one should presume the extractions of the two players to be independent of each other. For simplicity, we will further assume that players extract rows and columns independently, as stated in the following assumption:

Assumption 2.1 (Independence) *The four random matrices $\Gamma_1, \Pi_1, \Gamma_2, \Pi_2$ are statistically independent.* ■

Remark 2.1 (Non-matrix games) The results in this paper do not depend on the fact that the original game is a matrix game. They extend trivially to any cost-function $J(u, d)$, $u \in \mathcal{U}$, $d \in \mathcal{D}$ where \mathcal{U} and \mathcal{D} denote the sets of policies for the minimizer and maximizer, respectively. In fact, it is not even necessary that the original game has saddle-point policies since all that the SSP algorithm uses is the fact that when we take finite samples of the sets of policies, we obtain finite matrix games. ■

Remark 2.2 (Non-unique security policies) When the matrices A_1 and A_2 have multiple security policies, the SSP algorithm does not specify *which* of these should be used to define the sampled security policies. However, the choice of security policy may have a significant effect on the value of the probabilities in (1) and (2). So, any useful probabilistic guarantee for ϵ -security should hold independently of which security policy is used in the SSP algorithm. This is the case of all the results in this paper. ■

3 Bounds for Probabilistic Guarantees

In this section, we present theoretical bounds on the size of the sub-matrices that guarantee that the SSP algorithm and the resulting policies are probabilistically secure. The results of this section focus on the case when the players sample their submatrices from identical distributions. This assumption will be relaxed in Section 4.

3.1 Game-Independent Bounds

The main result of this section provides a bound on the size of the submatrices for the players that guarantees ϵ -security with $\epsilon = 0$.

Theorem 3.1 (Game independent bounds) *Suppose that Assumption 2.1 holds. Then*

1. *If Π_1 and Π_2 have identically distributed columns and*

$$n_1 = \left\lceil \frac{m_1 + 1}{\delta} - 1 \right\rceil \bar{n}_2 \quad (5)$$

for some $\bar{n}_2 \geq n_2$, then the SSP algorithm is $\epsilon = 0$ -secure for P_1 with confidence $1 - \delta$. If one further increases n_1 to satisfy

$$n_1 = \left\lceil \frac{1}{\delta} \left(\ln \frac{1}{\beta} + m_1 + \sqrt{2m_1 \ln \frac{1}{\beta}} \right) \right\rceil \bar{n}_2 \quad (6)$$

for some $\beta \in (0, 1)$, then, with probability¹ higher than $1 - \beta$, the policy y^ with value $\bar{V}(A_1)$ is $\epsilon = 0$ -secure for P_1 with confidence $1 - \delta$.*

2. *If Γ_1 and Γ_2 have identically distributed columns and*

$$m_2 = \left\lceil \frac{n_2 + 1}{\delta} - 1 \right\rceil \bar{m}_1 \quad (7)$$

for some $\bar{m}_1 \geq m_1$, then the SSP algorithm is $\epsilon = 0$ -secure for P_2 with confidence $1 - \delta$. If one further increases m_2 to satisfy

$$m_2 = \left\lceil \frac{1}{\delta} \left(\ln \frac{1}{\beta} + n_2 + \sqrt{2n_2 \ln \frac{1}{\beta}} \right) \right\rceil \bar{m}_1 \quad (8)$$

for some $\beta \in (0, 1)$, then, with probability² higher than $1 - \beta$, the policy z^ with value $\underline{V}(A_2)$ is $\epsilon = 0$ -secure for P_2 with confidence $1 - \delta$. ■*

In words, this results states that it is always possible to guarantee $\epsilon = 0$ -security for P_1 , if she constructs her submatrix A_1 utilizing a sufficiently large number of columns. In particular, she always needs to choose a number of columns n_1 larger than the number of columns n_2 that P_2 is considering for her mixed policies (cf. (5) and (6)). The additional number of columns that P_1 needs to consider is a function of the number m_1 of rows that P_1 wants to consider for her mixed policy and the desired confidence levels. The result for P_2 is analogous.

¹The confidence level β for P_1 refers solely to the extraction of the matrix Π_1 and holds for any given matrix Γ_1 .

²The confidence level β for P_2 refers solely to the extraction of the matrix Γ_2 and holds for any given matrix Π_2 .

The confidence level $1 - \beta$ that appears in the bound (6) for P_1 's policy-specific probabilistic guarantees refers to the probability that the bound fails altogether due to an “unfortunate” sample used by P_1 to compute the policy y^* . However, note that only the logarithm of the confidence level β appears in bounds regarding the security of y^* and z^* . One can therefore make β extremely small with a relatively small additional computational cost.

In the probabilistic guarantees provided by Theorem 3.1 with (5), the confidence $1 - \delta$ refers to the extraction of all the row/column matrices $\Gamma_1, \Gamma_2, \Pi_1, \Pi_2$ as in (1). However, for the probabilistic guaranteed with (6), the confidence $1 - \delta$ refers to the extraction of Γ_2, Π_2 as in (3), whereas the confidence $1 - \beta$ refers solely to the extraction of the matrix Π_1 and holds for any given matrix Γ_1 (as shown in the proof).

Remark 3.2 (P_1 's knowledge of n_2) According to Theorem 3.1, for player P_1 to enjoy guaranteed $\epsilon = 0$ -security with confidence $1 - \delta$, she must know an upper bound \bar{n}_2 on the number of columns that P_2 is using to construct her submatrix A_2 . Even if P_1 does not know \bar{n}_2 precisely and, e.g., underestimates \bar{n}_2 by a certain percentage, then (5) and (6) are still useful in that they predict that the performance degradation in the confidence level δ should grow proportionately. This is because the bounds in (5) and (6) essentially scale with \bar{n}_2/δ . Analogous comments can be made on (7) and (8) and on P_2 's knowledge of m_1 . ■

Now, suppose that P_1 restricts herself to pure policies instead of mixed policies. Then, a bound similar to (6) can be established for a pure security policy y_p^* with a pure security level $\bar{V}_p(A_1)$. We now present the result for P_1 . Analogous result can be stated for P_2 .

Theorem 3.3 (Bound with pure policies) *Suppose that Assumption 2.1 holds. If Π_1 and Π_2 have identically distributed columns and*

$$n_1 = \left\lceil \frac{1}{\delta} \ln \frac{m_1 + 1}{\beta} \right\rceil \bar{n}_2,$$

for some $\beta \in (0, 1)$ and for some $\bar{n}_2 \geq n_2$, then with probability higher than $1 - \beta$, the pure policy y_p^* with the pure security value $\bar{V}_p(A_1)$ is $\epsilon = 0$ -secure for P_1 with confidence $1 - \delta$.

In terms of computational complexity, Theorem 3.3 provides significant improvement over the linear dependence in m_1 (cf. (6)). However, the corresponding pure security level $\bar{V}_p(A_1)$ could be much higher than the one in mixed policies $\bar{V}(A_1)$. Thus, pure policies are useful to consider only if faced with computational difficulties.

We now present the proof of Theorem 3.1.

Proof of Theorem 3.1: We only prove the statement 1, since the proof of statement 2 can be obtained by symmetry. From the definition of the security value $\bar{V}(A_1)$,

$$\begin{aligned} \bar{V}(A_1) &= \min_{y \in \mathcal{S}_{m_1}} \max_{z \in \mathcal{S}_{n_1}} y' \Gamma_1' A \Pi_1 z \\ &= \min_{y \in \mathcal{S}_{m_1}} \max_{j \in \{1, \dots, n_1\}} y' \Gamma_1' A \Pi_1 e_j \\ &= \min_{\theta \in \Theta} \left\{ v : y' \Gamma_1' A \Pi_1 e_j \leq v, \forall j \in \{1, \dots, n_1\} \right\}, \end{aligned} \tag{9}$$

where e_j denotes the j th element of the canonical basis of \mathbb{R}^{n_1} , $\theta := (y, v)$, and $\Theta := \mathcal{S}_{m_1} \times \mathbb{R}$.

Since n_1 is an integer multiple of \bar{n}_2 , i.e., $n_1 = K\bar{n}_2$ with $K = \left\lceil \frac{m_1+1}{\delta} - 1 \right\rceil$, we can take the $K\bar{n}_2$ columns of $\Pi_1 \in \mathcal{B}^{N \times K\bar{n}_2}$ to construct K i.i.d. matrices $\Delta_1, \Delta_2, \dots, \Delta_K$, each in the set $\mathcal{B}^{N \times \bar{n}_2}$. If we then define the function

$$f(\theta, \Delta) = \max_{j \in \{1, \dots, \bar{n}_2\}} y' \Gamma_1' A \Delta e_j - v,$$

$\forall \theta := (y, v) \in \Theta$, $\Delta \in \mathcal{B}^{N \times \bar{n}_2}$, we can rewrite (9) as

$$\bar{V}(A_1) = \min_{\theta \in \Theta} \left\{ v : f(\theta, \Delta_i) \leq 0, \forall i \in \{1, \dots, K\} \right\},$$

Let the minimum above be achieved for some $\theta^* = (y_1^*, \bar{V}(A_1))$. For any given realization of the matrix Γ_1 (which is independent of the Δ_i by Assumption 2.1) we conclude from [7, Proposition 3] that the (conditional) probability that another matrix Δ sampled independently from the same distribution as the Δ_i satisfies the constraint $f(\theta^*, \Delta) \leq 0$ can be lower-bounded as follows:

$$\mathbb{P}_{\Pi_1, \Delta} (f(\theta^*, \Delta) \leq 0 \mid \Gamma_1) \geq \frac{K - m_1}{K + 1} \geq 1 - \delta, \quad (10)$$

where the second inequality is a consequence of (5). Using the definition of f and θ^* , we can re-write (10) as

$$\mathbb{P}_{\Pi_1, \Delta} (y_1^{*'} \Gamma_1' A \Delta e_j \leq \bar{V}(A_1), \forall j \in \{1, \dots, \bar{n}_2\} \mid \Gamma_1) \geq 1 - \delta.$$

Since $n_2 \leq \bar{n}_2$, we further conclude that

$$\mathbb{P}_{\Pi_1, \Delta} (y_1^{*'} \Gamma_1' A \Delta e_j \leq \bar{V}(A_1), \forall j \in \{1, \dots, n_2\} \mid \Gamma_1) \geq 1 - \delta.$$

Under Assumption 2.1, when the columns of Π_1 and Π_2 are identically distributed, the matrix consisting of the first n_2 columns of Δ can be viewed as the matrix Π_2 and we conclude from the inequality above that

$$\mathbb{P}_{\Pi_1, \Pi_2} (y_1^{*'} \Gamma_1' A \Pi_2 e_j \leq \bar{V}(A_1), \forall j \in \{1, \dots, n_2\} \mid \Gamma_1) \geq 1 - \delta.$$

Since

$$y_1^{*'} \Gamma_1' A \Pi_2 e_j \leq \bar{V}(A_1), \forall j \in \{1, \dots, n_2\} \Rightarrow y_1^{*'} \Gamma_1' A \Pi_2 z \leq \bar{V}(A_1), \forall z \in \mathcal{S}^{n_2},$$

we conclude that

$$\mathbb{P}_{\Pi_1, \Gamma_2, \Pi_2} (y_1^{*'} \Gamma_1' A \Pi_2 z^* \leq \bar{V}(A_1) \mid \Gamma_1) \geq 1 - \delta.$$

Since we have shown that this bound holds for an arbitrary realization of Γ_1 , it also holds for the unconditional probability, which shows that the SSP algorithm is $\epsilon = 0$ -secure for \mathbb{P}_1 with confidence $1 - \delta$.

If instead of applying [7, Proposition 3] we apply [2, Theorem 4], then using (6), we conclude that

$$\mathbb{P}_\Delta (f(\theta^*, \Delta) \leq 0 \mid \Gamma_1, \theta^*) \geq 1 - \delta$$

with probability higher than $1 - \beta$, where the confidence level $1 - \beta$ refers to the extraction of $\Pi_1 = [\Delta_1, \dots, \Delta_K]$ that defines θ^* (given Γ_1). The proof can now proceed exactly as before, but with (10) replaced by the inequality above, which now involves a probability conditioned to y^* and $\bar{V}(A_1)$. This shows that, with probability higher than $1 - \beta$, the policy y^* with value $\bar{V}(A_1)$ is $\epsilon = 0$ -secure for \mathbb{P}_1 with confidence $1 - \delta$. \blacksquare

The proof of Theorem 3.3 is analogous to the proof of Theorem 3.1, with the main difference being the choice of $K = \left\lceil \frac{1}{\delta} \ln \frac{m_1+1}{\beta} \right\rceil$, and the use of [2, Theorem 3] instead of [2, Theorem 4].

3.2 A-posteriori Probabilistic Guarantees

Suppose that, due to computational limitations, player P_1 cannot satisfy the bounds in Theorem 3.1 to obtain $\epsilon = 0$ -security for a given level of confidence $1 - \delta$. One option to overcome this difficulty would be to settle for a lower level of confidence until the bounds in Theorem 3.1 hold for a value of n_1 that is computationally acceptable for P_1 . However, one may desire to maintain the same high level of confidence, and instead accept a larger value for ϵ . In this section, we explore this option, which is not covered by Theorem 3.1. For brevity, we only consider the SSP algorithm from the perspective of P_1 .

Consider the following procedure for P_1 :

1. Pick values for m_1, n_1 and use the SSP algorithm to compute a sampled security policy y^* and the corresponding sampled security value $\bar{V}(A_1)$.
2. Using the column distribution of Π_1 , independently extract k_1 columns of A into a matrix $\bar{\Pi}_1 \in \mathcal{B}^{N \times k_1}$ and compute the row vector

$$\bar{v} := \max_{j \in \{1, \dots, k_1\}} y^{*'} A \bar{\Pi}_1 e_j, \quad (11)$$

where e_j denotes the j th element of the canonical basis of \mathbb{R}^{k_1} .

The following result provides an a-posteriori guarantee on this procedure.

Theorem 3.4 (A-posteriori bounds) *Suppose that Assumption 2.1 holds. If Π_1 and Π_2 have identically distributed columns and*

$$k_1 = \left\lceil \frac{1}{\delta} - 1 \right\rceil \bar{n}_2, \quad (12)$$

for some $\bar{n}_2 \geq n_2$, then the SSP algorithm is ϵ -secure for P_1 with confidence $1 - \delta$ for any

$$\epsilon \geq \bar{v} - \bar{V}(A_1). \quad (13)$$

If one further increases k_1 to satisfy

$$k_1 = \left\lceil \frac{1}{\delta} \ln \frac{1}{\beta} \right\rceil \bar{n}_2, \quad (14)$$

then, with probability higher than $1 - \beta$, the policy y^* with value $\bar{V}(A_1)$ is ϵ -secure for P_1 with confidence $1 - \delta$. ■

In the probabilistic guarantee provided by Theorem 3.4 with (12), the confidence $1 - \delta$ refers not only to the extraction of the row/column matrices $\Gamma_1, \Gamma_2, \Pi_1, \Pi_2$, but also to the test matrix $\bar{\Pi}_1$ since ϵ depends on it, i.e., (1) should be understood as

$$P_{\Gamma_1, \Pi_1, \Gamma_2, \Pi_2, \bar{\Pi}_1} (y^{*'} A z^* \leq \bar{V}(A_1) + \epsilon) \geq 1 - \delta. \quad (15)$$

For the probabilistic guarantee with (14), the confidence $1 - \delta$ refers to the extraction of Γ_2, Π_2 , i.e., (3) should be understood as

$$P_{\Gamma_2, \Pi_2} (y^{*'} A z^* \leq \bar{V}(A_1) + \epsilon \mid y^*, \bar{V}(A_1), \epsilon) \geq 1 - \delta$$

whereas the confidence $1 - \beta$ refers solely to the extraction of the matrix $\bar{\Pi}_1$.

Proof of Theorem 3.4: From the definition of \bar{v} and (13), we conclude that

$$\bar{V}(A_1) + \epsilon \geq \bar{v} = \max_{j \in \{1, \dots, K\bar{n}_2\}} y^{*'} A \bar{\Pi}_1 e_j, \quad (16)$$

where $K := \left\lceil \frac{1}{\delta} - 1 \right\rceil$. Partitioning the columns of $\bar{\Pi}_1 \in \mathcal{B}^{N \times K\bar{n}_2}$ to construct K i.i.d. matrices $\Delta_1, \Delta_2, \dots, \Delta_K$, each in the set $\mathcal{B}^{N \times \bar{n}_2}$ and defining

$$f(\Delta) = \max_{j \in \{1, \dots, \bar{n}_2\}} y^{*'} A \Delta e_j, \quad \forall \Delta \in \mathcal{B}^{N \times \bar{n}_2}$$

we can rewrite (16) as

$$\bar{V}(A_1) + \epsilon \geq \max_{i \in \{1, \dots, K\}} f(\Delta_i). \quad (17)$$

For any given realizations of y^* and $\bar{V}(A_1)$ (which are independent of the Δ_i), we conclude from [7, Proposition 4] that the (conditional) probability that another matrix Δ , sampled independently from the same distribution as the Δ_i , satisfies the constraint $f(\Delta) \leq \max_{i \in \{1, \dots, K\}} f(\Delta_i)$ can be lower-bounded as follows:

$$P_{\bar{\Pi}_1, \Delta} \left(f(\Delta) \leq \max_{i \in \{1, \dots, K\}} f(\Delta_i) \mid y^*, \bar{V}(A_1) \right) \geq \frac{K}{K+1} \geq 1 - \delta, \quad (18)$$

where the second inequality is a consequence of (12). From the definition of f and (17), we conclude from (18) that

$$P_{\bar{\Pi}_1, \Delta} \left(\max_{j \in \{1, \dots, \bar{n}_2\}} y^{*'} A \Delta e_j \leq \bar{V}(A_1) + \epsilon \mid y^*, \bar{V}(A_1) \right) \geq 1 - \delta,$$

and therefore

$$P_{\bar{\Pi}_1, \Delta} \left(y^{*'} A \Delta e_j \leq \bar{V}(A_1) + \epsilon, \forall j \in \{1, \dots, \bar{n}_2\} \mid y^*, \bar{V}(A_1) \right) \geq 1 - \delta.$$

Since $n_2 \leq \bar{n}_2$, we further conclude that

$$P_{\bar{\Pi}_1, \Delta} \left(y^{*'} A \Delta e_j \leq \bar{V}(A_1) + \epsilon, \forall j \in \{1, \dots, n_2\} \mid y^*, \bar{V}(A_1) \right) \geq 1 - \delta.$$

Under Assumption 2.1, when the columns of Π_1 and Π_2 are identically distributed, the matrix consisting of the first n_2 columns of Δ can be viewed as the matrix Π_2 and we conclude from the inequality above that

$$P_{\bar{\Pi}_1, \Pi_2} \left(y^{*'} A \Pi_2 e_j \leq \bar{V}(A_1) + \epsilon, \forall j \in \{1, \dots, n_2\} \mid y^*, \bar{V}(A_1) \right) \geq 1 - \delta.$$

Given that

$$y^{*'} A \Pi_2 e_j \leq \bar{V}(A_1) + \epsilon, \forall j \in \{1, \dots, n_2\} \Rightarrow y^{*'} A \Pi_2 z \leq \bar{V}(A_1) + \epsilon, \forall z \in \mathcal{S}^{n_2},$$

we get that

$$\mathbb{P}_{\Gamma_2, \Pi_2, \bar{\Pi}_1} \left(y^{*\prime} A \Pi_2 z_2^* \leq \bar{V}(A_1) + \epsilon \mid y^*, \bar{V}(A_1) \right) \geq 1 - \delta.$$

Since we have shown that this bound holds for arbitrary realizations of y^* and $\bar{V}(A_1)$, it also holds for the unconditional probability, from which (15) follows.

If instead of applying [7, Proposition 4] we use (14) and apply [8, Theorem 1], we conclude that

$$\mathbb{P}_\Delta \left(f(\Delta) \leq \max_{i \in \{1, \dots, K\}} f(\Delta_i) \mid y^*, \bar{V}(A_1), \epsilon \right) \geq 1 - \delta,$$

with probability higher than $1 - \beta$, where the confidence level $1 - \beta$ refers to the extraction of $\bar{\Pi}_1 = [\Delta_1, \dots, \Delta_K]$ that defines ϵ . The proof can now proceed exactly as before, but with (18) replaced by the inequality above that now involves a probability conditioned to y^* , $\bar{V}(A_1)$, and ϵ . This shows that, with probability higher than $1 - \beta$, the policy y^* with value $\bar{V}(A_1)$ is ϵ -secure for \mathbb{P}_1 with confidence $1 - \delta$. \blacksquare

4 Mismatch in the Sampling Distributions

In this section, we relax the assumption that both players sample from identical distributions to construct their respective submatrices. The proofs of the intermediate results, i.e., Propositions 4.1, 4.2 and Lemma 4.6, are presented in the Appendix.

We begin with a general result which extends [7, Proposition 3] to mismatched distributions. Consider a sequence of $K + 1$ i.i.d. random variables $\Delta, \Delta_1, \Delta_2, \dots, \Delta_K$ taking values in a set \mathcal{D} , with a probability measure $\mathbb{P}_\Delta(\cdot)$ ³. Further consider a convex function $f : \Theta \times \mathcal{D} \rightarrow \mathbb{R}$, where Θ is a convex subset of \mathbb{R}^{n_θ} . Define the random variable⁴

$$\theta^*(\Delta_1, \dots, \Delta_K) = \operatorname{argmin}_{\theta \in \Theta} \{c\theta : f(\theta, \Delta_i) \leq 0, \forall i = 1, \dots, K\}, \quad (19)$$

Consider now an additional independent random variable $\bar{\Delta} \in \mathcal{D}$ having a distinct probability measure $\mathbb{P}_{\bar{\Delta}}(\cdot)$. Then, the following result holds.

Proposition 4.1 *Suppose there exists non-negative numbers ϵ, μ such that*

$$\mathbb{P}_{\bar{\Delta}}(f(\theta, \bar{\Delta}) > \epsilon) \leq \mu \mathbb{P}_\Delta(f(\theta, \Delta) > 0), \quad \forall \theta \in \Theta. \quad (20)$$

Then, for θ^ defined as in (19), we have that*

$$\mathbb{P}_{\bar{\Delta}, \Delta_1, \dots, \Delta_K} \left(f(\theta^*(\Delta_1, \dots, \Delta_K), \bar{\Delta}) > \epsilon \right) \leq \frac{\mu n_\theta}{K + 1}. \quad (21)$$

This result holds with the choice of

$$\mu = \mu_{\text{mis}}(\epsilon) := \sup_{\theta \in \Theta} \frac{\mathbb{P}_{\bar{\Delta}}(f(\theta, \bar{\Delta}) > \epsilon)}{\mathbb{P}_\Delta(f(\theta, \Delta) > 0)}, \quad (22)$$

where $\mu_{\text{mis}}(\epsilon)$ could be viewed as a *mismatch parameter*. When the distribution of Δ is identical to that of $\bar{\Delta}$, (20) holds for $\epsilon = 0$, and with $\mu = \mu_{\text{mis}}(0) = 1$.

Next, we prove the following result which involves two levels of probability, and thus generalizes [2, Theorem 4].

³In what follows, we implicitly assume that all sets that appear as arguments of probability measures are measurable.

⁴This result holds independent of any rule used to select one out of several possible multiple minima.

Proposition 4.2 Given $\delta \in (0, 1)$, $\beta \in (0, 1)$ and any $\epsilon \geq 0$, with the number of i.i.d. random variables K satisfying

$$K \geq \left\lceil \mu_{\text{mis}}(\epsilon) \frac{1}{\delta} \left(\ln \frac{1}{\beta} + (n_\theta - 1) + \sqrt{2(n_\theta - 1) \ln \frac{1}{\beta}} \right) \right\rceil, \quad (23)$$

with probability (with respect to the measure \mathbb{P}_Δ) higher than $1 - \beta$,

$$\mathbb{P}_{\bar{\Delta}}(f(\theta^*, \bar{\Delta}) > \epsilon | \theta^*) \leq \delta.$$

4.1 Bounds with Mismatched Distributions

The analogue of Theorem 3.1 for the case of mismatched distributions is the following result.

Corollary 4.3 (Game independent bounds) Suppose that \mathbb{P}_1 and \mathbb{P}_2 sample the columns independently with distributions $\mathbb{P}_\Delta(\cdot)$ and $\mathbb{P}_{\bar{\Delta}}(\cdot)$, respectively. For any $\epsilon \geq 0$, determine $\mu_{\text{mis}}(\epsilon)$ using (22). If

$$n_1 = \left\lceil \mu_{\text{mis}}(\epsilon) \frac{m_1 + 1}{\delta} - 1 \right\rceil \bar{n}_2,$$

for some $\bar{n}_2 \geq n_2$, then the SSP algorithm is ϵ -secure with confidence $1 - \delta$ for the player \mathbb{P}_1 . If, for some $\beta \in (0, 1)$, one further increases n_1 to satisfy

$$n_1 = \left\lceil \mu_{\text{mis}}(\epsilon) \frac{1}{\delta} \left(\ln \frac{1}{\beta} + m_1 + \sqrt{2m_1 \ln \frac{1}{\beta}} \right) \right\rceil \bar{n}_2$$

then, with probability ⁵ higher than $1 - \beta$, the policy y^* with value $\bar{V}(A_1)$ is ϵ -secure for \mathbb{P}_1 with confidence $1 - \delta$.

Analogous result holds for \mathbb{P}_2 with the row distributions. The proof is identical to that of Theorem 3.1, with [7, Proposition 3] replaced by Proposition 4.1 and with [2, Theorem 4] replaced by Proposition 4.2.

For the a-posteriori procedure of Section 3.2, the analogue of Theorem 3.4 with mismatched distributions is the following result.

Corollary 4.4 (A-posteriori bounds) Suppose that \mathbb{P}_1 and \mathbb{P}_2 sample the columns independently with distributions $\mathbb{P}_\Delta(\cdot)$ and $\mathbb{P}_{\bar{\Delta}}(\cdot)$, respectively. With $\mu_{\text{mis}}(\bar{\epsilon})$ given by (22), for any $\bar{\epsilon} \geq 0$, if

$$k_1 = \left\lceil \mu_{\text{mis}}(\bar{\epsilon}) \frac{1}{\delta} - 1 \right\rceil \bar{n}_2,$$

for some $\bar{n}_2 \geq n_2$, then the SSP algorithm is ϵ -secure for \mathbb{P}_1 with confidence $1 - \delta$, for any

$$\epsilon \geq \max\{\bar{v} - \bar{V}(A_1), \bar{\epsilon}\}.$$

If, for some $\beta \in (0, 1)$, one further increases k_1 to satisfy

$$k_1 = \left\lceil \frac{\mu_{\text{mis}}(\bar{\epsilon})}{\delta} \ln \frac{1}{\beta} \right\rceil \bar{n}_2,$$

then, with probability higher than $1 - \beta$, the policy y^* with value $\bar{V}(A_1)$ is ϵ -secure for \mathbb{P}_1 with confidence $1 - \delta$. ■

⁵This probability is with respect to the measure $\mathbb{P}_\Delta(\cdot)$.

The proof is identical to that of Theorem 3.4, with [7, Proposition 4] replaced by Proposition 4.1 and with [8, Theorem 1] replaced by Proposition 4.2, and with $n_\theta = 1$.

Thus, when there is a mismatch in the sampling distributions of the players, the sizes of the subgames depend on the mismatch parameter $\mu_{\text{mis}}(\epsilon)$. The larger the mismatch, the higher is the size of subgames to be sampled for obtaining the same probabilistic level of security. Although these bounds may be conservative, they are still useful in the sense that if there is a sufficiently “small” mismatch between the distributions, then the sizes of the subgames may only be slightly higher than those in the case of identical distributions.

The results in this section become overly conservative when P_1 samples with low probability certain columns of the matrix A which violate the constraints in the optimization problem (19). In the following sub-section, we will see that under cases such as policy domination in the matrix game and with a specified perturbation in distributions of the players, the same bounds as in Section 3 guarantee a reasonable security level for both players.

4.2 Matrix games with Dominated policies

Consider a situation when P_1 is aware of any good policies that P_2 may apply to play the game. Then, P_1 should secure herself by playing against the superior policies of P_2 while constructing her submatrix A_1 . For example, suppose that the entries in column c_1 of A are all element-wise less than those in column c_2 . Then, P_1 should sample column c_1 with probability zero, column c_2 with probability equal to the original sum of the probability mass of c_1 and c_2 , and the remaining columns identical to that of the distribution used by P_2 to sample those columns.

This example motivates the case of policy domination which is the focus of this sub-section. We consider the following type of domination in matrix games.

Definition 1 (ϵ -Dominance in Matrix Games) *Given a $M \times N$ matrix A and an $\epsilon \geq 0$, a matrix $\Pi^* \in \mathcal{B}^{N \times n^*}$ is said to ϵ -dominate a disjoint matrix $\Pi \in \mathcal{B}^{N \times n}$, if there exists a strategy $z_{\text{dom}} \in \mathcal{S}_{n^*}$ such that, for every $j \in \{1, \dots, n\}$ and for every $i \in \{1, \dots, M\}$,*

$$r'_i A \Pi^* z_{\text{dom}} \leq r'_i A \Pi e_j + \epsilon,$$

where r_i and e_j denote the i th and the j th elements of the canonical basis of \mathbb{R}^M and of \mathbb{R}^n , respectively.

We now introduce the notion of two sampling distributions being ϵ -perturbed.

Definition 2 (ϵ -perturbed sampling) *Given that a matrix $\Pi^* \in \mathcal{B}^{N \times n^*}$ ϵ -dominates a disjoint matrix $\Pi \in \mathcal{B}^{N \times n}$, for some $\epsilon \geq 0$. The distributions $P_{\Pi_1}, P_{\Pi_2} : \{1, 2, \dots, N\} \rightarrow [0, 1]$ are ϵ -perturbed if*

1. *the distributions match outside of the dominating and the dominated columns of A , i.e.,*

$$P_{\Pi_1}(j) = P_{\Pi_2}(j),$$

for all j such that $e_j \notin \text{Range}(\Pi^) \cup \text{Range}(\Pi)$.*

2. *the distribution $P_{\Pi_1}(\cdot)$ is point-wise larger over the dominated columns of A , i.e.,*

$$P_{\Pi_2}(j) \leq P_{\Pi_1}(j),$$

for all j such that $e_j \in \text{Range}(\Pi)$.

Here, e_j denotes the j th element of the canonical basis of \mathbb{R}^N .

We now present the main result of this sub-section. For brevity, we will address only the game independent bounds here, since a similar proof technique can be used to derive analogous a-posteriori bounds. We present only the result for P_1 , since the result for P_2 is symmetric.

Theorem 4.5 (Domination) *Given the matrix A , suppose that for some $\epsilon \geq 0$, there exists a matrix $\Pi^* \in \mathcal{B}^{N \times n^*}$ which ϵ -dominates a disjoint matrix $\Pi \in \mathcal{B}^{N \times n}$. If the probability distributions of Π_1 and Π_2 are ϵ -perturbed, then, with*

$$n_1 = \left\lceil \frac{m_1 + 1}{\delta} - 1 \right\rceil \bar{n}_2,$$

for some $\bar{n}_2 \geq n_2$, the SSP algorithm is ϵ -secure for P_1 with confidence $1 - \delta$. If one further increases n_1 to satisfy

$$n_1 = \left\lceil \frac{1}{\delta} \left(\ln \frac{1}{\beta} + m_1 + \sqrt{2m_1 \ln \frac{1}{\beta}} \right) \right\rceil \bar{n}_2$$

for some $\beta \in (0, 1)$, then, with probability higher than $1 - \beta$, the policy y^* with value $\bar{V}(A_1)$ is ϵ -secure for P_1 with confidence $1 - \delta$.

To prove Theorem 4.5, we introduce a more abstract notion of ϵ -dominance and ϵ -perturbed measures.

Definition 3 (ϵ -dominance) *The value $D^* \in \mathcal{D}$ is said to ϵ -dominate a set $\mathcal{D}^* \subset \mathcal{D} \setminus \{D^*\}$ if*

$$f(\theta, D^*) \leq f(\theta, D) + \epsilon, \quad \forall \theta \in \Theta, D \in \mathcal{D}^*,$$

for some constant $\epsilon \geq 0$.

Definition 4 (ϵ -perturbed measures) *Given that D^* ϵ -dominates the set $\mathcal{D}^* \subset \mathcal{D} \setminus \{D^*\}$ for some $\epsilon \geq 0$. Then, the measures $P_{\Delta}(\cdot)$ and $P_{\bar{\Delta}}(\cdot)$ are ϵ -perturbed if*

1. the measures $P_{\Delta}(\cdot)$ and $P_{\bar{\Delta}}(\cdot)$ match outside $\{D^*\} \cup \mathcal{D}^*$ in the sense that

$$P_{\bar{\Delta}}(\mathcal{Q}) = P_{\Delta}(\mathcal{Q}),$$

for all measurable subsets \mathcal{Q} of $\mathcal{D} \setminus \{D^*\} \cup \mathcal{D}^*$.

2. The measure $P_{\Delta}(\cdot)$ is larger than $P_{\bar{\Delta}}(\cdot)$ inside \mathcal{D}^* in the sense that

$$P_{\bar{\Delta}}(\mathcal{Q}) \leq P_{\Delta}(\mathcal{Q}),$$

for all measurable subsets \mathcal{Q} of \mathcal{D}^* .

The following result states that when the two distributions differ because $P_{\bar{\Delta}}$ assigns less probability to dominated realizations, then (20) holds with $\mu = 1$, as if there was no mismatch between the distributions.

Lemma 4.6 *Suppose that D^* ϵ -dominates the set $\mathcal{D}^* \subset \mathcal{D} \setminus \{D^*\}$ for some $\epsilon \geq 0$, and that the measures $P_{\Delta}(\cdot)$ and $P_{\bar{\Delta}}(\cdot)$ are ϵ -perturbed. Then,*

1. Relation (21) holds with $\mu = 1$, and

2. Given $\delta \in (0, 1)$, $\beta \in (0, 1)$, and θ^* as defined in (19), if K satisfies

$$K \geq \left\lceil \frac{1}{\delta} \left(\ln \frac{1}{\beta} + (n_\theta - 1) + \sqrt{2(n_\theta - 1) \ln \frac{1}{\beta}} \right) \right\rceil,$$

with probability (with respect to the measure P_Δ) higher than $1 - \beta$,

$$P_{\bar{\Delta}}(f(\theta^*, \bar{\Delta}) > \epsilon | \theta^*) \leq \delta.$$

The proof of Theorem 4.5 is now exactly identical to that of Theorem 3.1, with [7, Proposition 3] replaced by statement 1 of Lemma 4.6 and with [2, Theorem 4] replaced by statement 2 of Lemma 4.6, and with $n_\theta = m_1 + 1$.

5 Example: Hide-and-seek matrix game

In this section, we apply the procedures from Sections 3 to a classic search problem through many points in a plane.

Consider a zero-sum game where P_1 hides a non-moving object (treasure) in one of N points $\{p_1, \dots, p_N\} \subset \mathbb{R}^2$ on the plane and P_2 has to find the treasure with minimum cost, by traveling from point to point until she finds it. The game is played over the set of mixed policies:

- P_1 chooses a probability distribution $z \in \mathcal{S}_N$ for the treasure over the N points, and
- P_2 chooses a probability distribution $y \in \mathcal{S}_M$ over the set $\mathcal{R} := \{r_j : j = 1, \dots, M\}$ of $M := N!$ routes that start at P_1 's initial position $p_0 \in \mathbb{R}^2$ and go through all possible permutations of the points.

Each route is assigned a cost equal to its total Euclidean length:

$$c(r_j) = \sum_{k=1}^N \|r_j(k) - r_j(k-1)\|,$$

where $r_j(0) := p_0$ and each subsequent $r_j(k) \in \mathbb{R}^2$, $k \in \{1, \dots, N\}$ denotes the k th point in route r_j . When P_1 chooses to hide the treasure at point i and P_2 selects route r_j , the outcome of the game is equal to the cost of route r_j from its initial point until the point p_i where the treasure lies. Namely,

$$A_{ij} = - \sum_{k=1}^{k_{ij}^*} \|r_j(k) - r_j(k-1)\|, \tag{24}$$

where the summation ends at the index k_{ij}^* for which $r_j(k_{ij}^*)$ corresponds to the point i where the treasure is hidden. The minus sign in (24) is needed to maintain consistency with the formulation in the first part of the paper, where P_1 is the minimizer. Indeed, P_1 hides the treasure to maximize the distance and therefore to minimize the entries of A .

The exact computation of the optimal mixed strategies is intractable because the size of the matrix A is $N \times N!$. However, the results in this paper regarding the SSP algorithm have a

computational complexity that is completely *independent of the size of the game*, which means that we can provide probabilistic guarantees for games with an arbitrarily large number of points.

In this particular game, only the player P_2 that chooses paths has a very large number of options ($M = N!$) so we can assume that both players consider all possible N locations where P_1 can hide the treasure (all rows of A), but randomly select only a small number of paths (columns of A) to construct their submatrices. This means that the player P_2 that selects the paths will never be surprised since she always considers all options for the actions of P_1 . However, the player P_1 that hides the treasure should respect the bounds provided by Theorems 3.1 and 3.4 to avoid unpleasant surprises.

In our numerical experiments, we considered N points distributed uniformly randomly in a square region. For a fixed value of \bar{n}_2 , β , and δ , we run Monte Carlo simulations of the procedure with the a-posteriori guarantees described in Section 3.2 using the bound in (14), and studied the outcome \bar{v} in (11) for increasing values of n_1 up to the corresponding a-priori bound (6), indicated by an arrow in Figure 1. Since \bar{v} is obtained through a randomized procedure, it is a random variable and takes different values in the different Monte Carlo simulations. In Figure 1, we show the dotted 90 (resp. dashed 50) percentile curve such that 90% (resp. 50%) of the realizations of \bar{v} were below this curve. We then repeated the experiments with the same values of δ and \bar{n}_2 , but using the a-posteriori bound in (12), and studied the outcome \bar{v} in (11) for increasing values of n_1 up to the corresponding a-priori bound (5). The obtained solid 90 (resp. dashed-dot 50) percentile curves are plotted in Figure 1.

We observe that all of these curves are reasonably "flat", implying that with the choice of n_1 that is a few orders of magnitude lower than the a-priori bound, one can obtain a security strategy that has a relatively small increase in the a-posteriori security level \bar{v} . For example, from Figure 1, we conclude that with a value of n_1 upto 40 times lower than the a-priori bound (6) needed for $\epsilon = 0$ -security of the policy y^* , in 90% (resp. 50%) of the simulations, the increase in the a-posteriori security level \bar{v} for the empirically obtained strategy y^* is at most 5 (resp. 3) units.

Figure 2 summarizes numerical results obtained with a higher value of \bar{n}_2 . Akin to Figure 1, we observe that all of the curves are reasonably flat and compared to Figure 1, the outcome \bar{v} increases on average, as is expected because player P_2 is allowed to select a larger number of columns and hence, to possibly achieve a larger value of the outcome of the game. (e.g., for $n_1 = 1000$, the 90 percentile curve is higher than in Figure 1 by at most 5 units).

Our numerical results indicate that by using much smaller number of column samples, significant computational savings can be obtained at the expense of a relatively small increase in the anticipated security level.

6 Conclusions and Future Directions

We addressed the solution of large zero-sum matrix games using randomized techniques. We provided a procedure by which each player samples a submatrix, computes mixed policies for the submatrix and uses the resulting optimal strategy to play against the other player. We proposed the notion of security policies and levels for each player, and derived a-priori game-independent bounds on the size of the submatrices that guarantees a security policy with high probability. We also presented an a-posteriori bound on how much the outcome of the game can violate the precomputed security level if the size of the submatrices do not satisfy the a-priori bounds. We

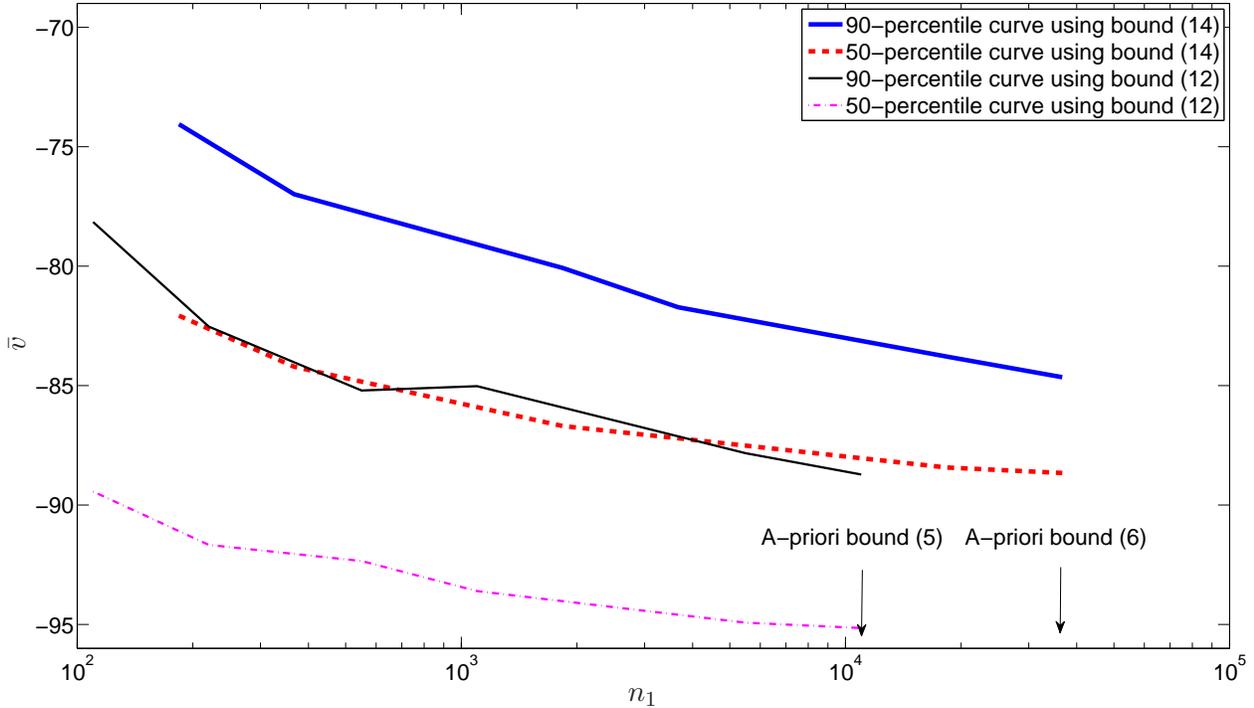


Figure 1: Numerically determined percentile values of the a-posteriori outcome \bar{v} (cf. Section 3.2) for different values of n_1 . In these experiments, the number of points is $N = 10$, side length of the square region is 50 units, $m_1 = \bar{n}_2 = 10$, $\delta = 0.01$, $\beta = 10^{-5}$, and the rows and the columns were drawn uniformly randomly.

then analyzed mismatch between the distributions used to obtain the subgames. We extended the theoretical bounds given the distributions used by the two players, and also analyzed the case of policy domination in matrix games. Finally, we applied the technique to solve a combinatorial hide-and-seek game.

This work suggests a number of exciting future directions of research. One promising direction is to explore incremental optimization techniques to reduce the bound on the size of the submatrices. Another direction being presently addressed is to extend the sampling procedure to dynamic or multi-stage games. Additionally, it would also be interesting to analyze closed-loop versions of the hide-and-seek game that involve the minimizer taking measurements of the location of the treasure as it moves from point to point.

References

- [1] T. Alamo, R. Tempo, and E. F. Camacho. Randomized strategies for probabilistic solutions of uncertain feasibility and optimization problems. *IEEE Transactions on Automatic Control*, 54(11):2545–2559, November 2009.
- [2] T. Alamo, R. Tempo, and A. Luque. On the sample complexity of randomized approaches to the analysis and design under uncertainty. In *American Control Conference*, pages 4671–4676, Baltimore, MD, USA, June–July 2010.

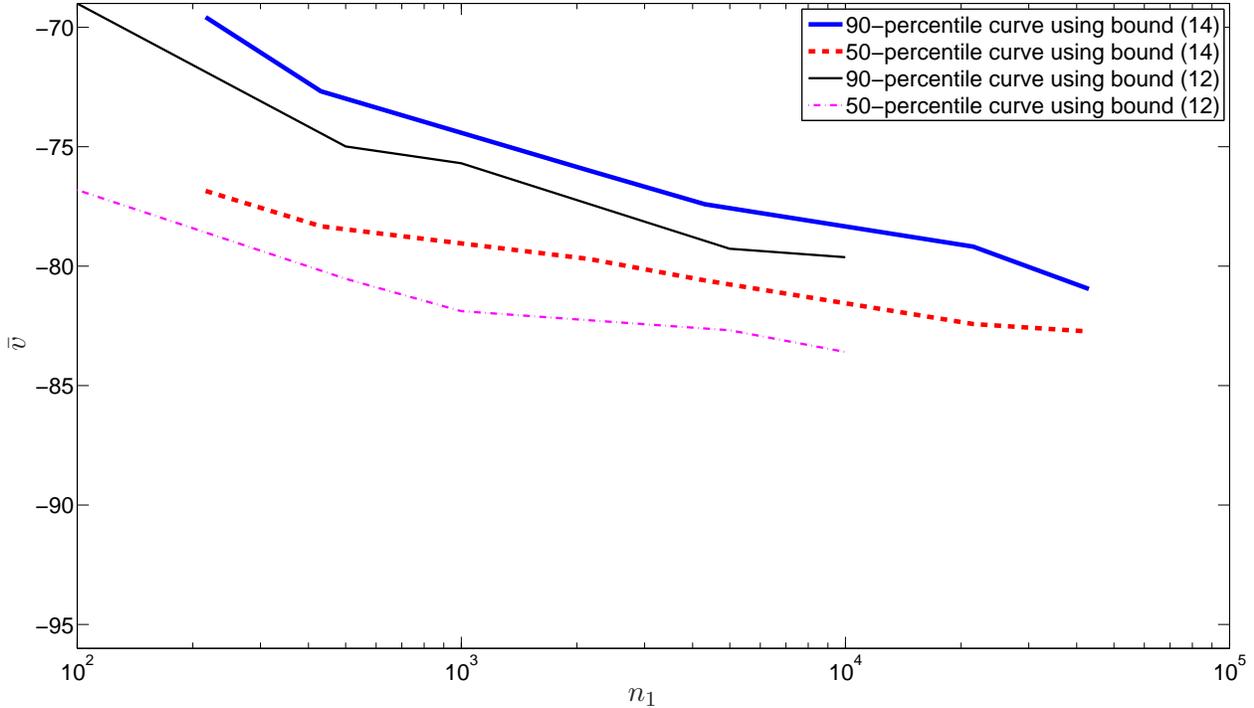


Figure 2: Numerically determined values of the a-posteriori outcome \bar{v} (cf. Section 3.2) for different values of n_1 . In these experiments, $N = 10$, $m_1 = 10$, $\bar{n}_2 = 1000$, $\delta = 0.01$, $\beta = 10^{-5}$, and the rows and the columns were drawn uniformly randomly.

- [3] T. Basar and G. J. Olsder. *Dynamic Non-Cooperative Game Theory*. SIAM, Philadelphia, PA, USA, 1999.
- [4] R. Bellman. Dynamic programming treatment of the traveling salesman problem. *J. Assoc. Comput. Mach.*, 9:61–63, 1962.
- [5] S. D. Bopardikar, A. Borri, J. P. Hespanha, M. Prandini, and M. D. Di Benedetto. Randomized sampling for large zero-sum games. In *IEEE Conference on Decision and Control*, Atlanta, GA, USA, Dec. 2010 (To appear).
- [6] G. C. Calafiore and M. C. Campi. The scenario approach to robust control design. *IEEE Transactions on Automatic Control*, 51(5):742–753, May 2006.
- [7] M. C. Campi and G. C. Calafiore. Notes on the scenario design approach. *IEEE Transactions on Automatic Control*, 54(2):382–385, February 2009.
- [8] M. C. Campi and S. Garatti. The exact feasibility of randomized solutions of robust convex programs. *SIAM Journal on Control and Optimization*, 19(3):1211–1230, 2008.
- [9] M. C. Campi, S. Garatti, and M. Prandini. The scenario approach for systems and control design. *Annual Reviews in Control*, 33(2):149–157, Dec. 2009.
- [10] J. P. Hespanha and M. Prandini. Nash equilibria in partial-information games on Markov chains. In *IEEE Conference on Decision and Control*, pages 2102–2107, Orlando, FL, USA, December 2001.

- [11] P. Khargonekar and A. Tikku. Randomized algorithms for robust control analysis and synthesis have polynomial complexity. In *Proceedings of the 35th IEEE Conference on Decision and Control*, volume 3, pages 3470–3475, Kobe, Japan, Dec. 1996.
- [12] R. J. Lipton and N. E. Young. Simple strategies for large zero-sum games with applications to complexity theory. In *Twenty-sixth annual ACM Symposium on Theory of Computing*, pages 734–740, 1994.
- [13] R. Motwani and P. Raghavan. *Randomized Algorithms*. Cambridge University Press, 1995.
- [14] R. Tempo, E. W. Bai, and F. Dabbene. Probabilistic robustness analysis: Explicit bounds for the minimum number of samples. *Systems and Control Letters*, 30(5):237–242, 1997.
- [15] V. Vapnik. *Statistical Learning Theory*. John Wiley, New York, 1998.
- [16] M. Vidyasagar. Statistical learning theory and randomized algorithms for control. *IEEE Control Systems Magazine*, 18(6):69–85, Dec. 1998.
- [17] M. Vidyasagar. Randomized algorithms for robust controller synthesis using statistical learning theory. *Automatica*, 37(10):1515–1528, October 2001.
- [18] M. Vidyasagar and V. D. Blondel. Probabilistic solutions to some NP–hard matrix problems. *Automatica*, 37(9):1397–1405, Sep. 2001.
- [19] J. Von Neumann. Zur theorie der gesellschaftsspiele. *Math. Annalen.*, 100:295–320, 1928.

Appendix

In this Appendix, we provide the proofs of intermediate results, viz. Propositions 4.1 and 4.2 and Lemma 4.6.

Proof of Proposition 4.1: For simplicity, we omit the explicit dependence of θ^* on $\Delta_1, \dots, \Delta_K$. We have

$$P_{\bar{\Delta}, \Delta_1, \dots, \Delta_K} (f(\theta^*, \bar{\Delta}) > \epsilon) = \mathbb{E}_{\Delta_1, \dots, \Delta_K} \left[P_{\bar{\Delta}} (f(\theta^*, \bar{\Delta}) > 0 | \theta^*) \right],$$

we can use (20) to conclude that

$$\begin{aligned} P_{\bar{\Delta}, \Delta_1, \dots, \Delta_K} (f(\theta^*, \bar{\Delta}) > \epsilon) &\leq \mu_{\text{mis}} \mathbb{E}_{\Delta_1, \dots, \Delta_K} \left[P_{\Delta} (f(\theta^*, \Delta) > 0 | \theta^*) \right] \\ &\leq \frac{\mu_{\text{mis}} n_{\theta}}{K + 1}, \end{aligned}$$

where the second inequality follows from [7, Proposition 3]. ■

Proof of Proposition 4.2: For any $\bar{\delta} > 0$, using [2, Theorem 4], we conclude that with K satisfying

$$K \geq \left\lceil \frac{1}{\bar{\delta}} \left(\ln \frac{1}{\beta} + (n_{\theta} - 1) + \sqrt{2(n_{\theta} - 1) \ln \frac{1}{\beta}} \right) \right\rceil,$$

with probability (with respect to the measure P_{Δ}) higher than $1 - \beta$,

$$P_{\Delta} (f(\theta, \Delta) > 0 | \theta^*) \leq \bar{\delta}.$$

Using Proposition 4.1, for any given $\epsilon \geq 0$,

$$P_{\bar{\Delta}}(f(\theta, \bar{\Delta}) > \epsilon | \theta^*) \leq \mu_{\text{mis}}(\epsilon) P_{\Delta}(f(\theta, \Delta) > 0 | \theta^*),$$

and setting $\bar{\delta} := \delta / \mu_{\text{mis}}(\epsilon)$, the claim is proved. \blacksquare

Proof of Lemma 4.6: We begin with the proof of statement 1. For a given value of $\theta \in \Theta$, consider the sets

$$\begin{aligned} \mathcal{D}_{\epsilon}(\theta) &:= \{D \in \mathcal{D} : f(\theta, D) > \epsilon\}, \\ \mathcal{D}(\theta) &:= \{D \in \mathcal{D} : f(\theta, D) > 0\}. \end{aligned}$$

Using these sets, we can expand

$$P_{\bar{\Delta}}(f(\theta, \bar{\Delta}) > \epsilon) = P_{\bar{\Delta}}(\mathcal{D}_{\epsilon}(\theta) \cap \{D^*\}) + P_{\bar{\Delta}}(\mathcal{D}_{\epsilon}(\theta) \cap \mathcal{D}^*) + P_{\bar{\Delta}}(\mathcal{D}_{\epsilon}(\theta) \setminus \{D^*\} \cup \mathcal{D}^*). \quad (25)$$

There arise two cases:

1. $D^* \notin \mathcal{D}_{\epsilon}(\theta)$: Then, (25) leads to

$$P_{\bar{\Delta}}(f(\theta, \bar{\Delta}) > \epsilon) = P_{\bar{\Delta}}(\mathcal{D}_{\epsilon}(\theta) \cap \mathcal{D}^*) + P_{\bar{\Delta}}(\mathcal{D}_{\epsilon}(\theta) \setminus \{D^*\} \cup \mathcal{D}^*).$$

Using both conditions of Definition 4, we conclude that

$$\begin{aligned} P_{\bar{\Delta}}(f(\theta, \bar{\Delta}) > \epsilon) &\leq P_{\Delta}(\mathcal{D}_{\epsilon}(\theta) \cap \mathcal{D}^*) + P_{\Delta}(\mathcal{D}_{\epsilon}(\theta) \setminus \{D^*\} \cup \mathcal{D}^*) \\ &= P_{\Delta}(f(\theta, \Delta) > \epsilon) \leq P_{\Delta}(f(\theta, \Delta) > 0), \end{aligned}$$

Proposition 4.1 completes the proof for this case.

2. If $D^* \in \mathcal{D}_{\epsilon}(\theta)$: Then, we have that

$$f(\theta, D^*) > \epsilon,$$

and since D^* ϵ -dominates \mathcal{D}^* , for every $D \in \mathcal{D}^*$, we have that

$$f(\theta, D) \geq f(\theta, D^*) - \epsilon > 0 \Rightarrow D \in \mathcal{D}(\theta),$$

This implies that $\mathcal{D}^* \subset \mathcal{D}(\theta)$. Similar to (25), we can write Using these sets, we can expand

$$\begin{aligned} P_{\bar{\Delta}}(f(\theta, \bar{\Delta}) > 0) &= P_{\bar{\Delta}}(\mathcal{D}(\theta) \cap \{D^*\}) + P_{\bar{\Delta}}(\mathcal{D}(\theta) \cap \mathcal{D}^*) + P_{\bar{\Delta}}(\mathcal{D}(\theta) \setminus \{D^*\} \cup \mathcal{D}^*) \\ &= P_{\bar{\Delta}}(\{D^*\}) + P_{\bar{\Delta}}(\mathcal{D}^*) + P_{\bar{\Delta}}(\mathcal{D}(\theta) \setminus \{D^*\} \cup \mathcal{D}^*), \end{aligned} \quad (26)$$

Thus,

$$\begin{aligned} P_{\bar{\Delta}}(f(\theta, \bar{\Delta}) > \epsilon) &\leq P_{\bar{\Delta}}(f(\theta, \bar{\Delta}) > 0) \\ &= 1 - P_{\bar{\Delta}}(\mathcal{D} \setminus \{D^*\} \cup \mathcal{D}^*) + P_{\bar{\Delta}}(\mathcal{D}(\theta) \setminus \{D^*\} \cup \mathcal{D}^*) \\ &= 1 - P_{\Delta}(\mathcal{D} \setminus \{D^*\} \cup \mathcal{D}^*) + P_{\Delta}(\mathcal{D}(\theta) \setminus \{D^*\} \cup \mathcal{D}^*), \end{aligned}$$

where the last inequality that allowed us to go from $P_{\bar{\Delta}}$ to P_{Δ} is a consequence of condition 1 of Definition 4. Taking the reverse steps following in the previous equation, we conclude that when $D^* \in \mathcal{D}_{\epsilon}(\theta)$, we have that

$$P_{\bar{\Delta}}(f(\theta, \bar{\Delta}) > \epsilon) \leq P_{\Delta}(f(\theta, \Delta) > 0).$$

Proposition 4.1 completes the proof also for this case, and thus, statement 1 is established.

To establish statement 2, we apply statement 1 given the value of θ^* in conjunction with [2, Theorem 4]. \blacksquare