Information Concealing Games

Saswati Sarkar *, Eitan Altman †, Rachid El-Azouzi ‡, Yezejael Hayel ‡
* Dept. of Electrical and Systems Eng., Univ. of Pennsylvania, 200 S. 33rd Str., Philadelphia 19104, USA
† INRIA, Centre Sophia-Antipolis, 2004 Route des Lucioles, 06902 Sophia-Antipolis Cedex, France
‡ Univesité d'Avignon, 339, Chemin des Meinajaries, Agroparc BP 1228, 84911 Avignon Cedex 9, France

***Abstract*— Consider the situation in which a decision maker (Actor) has to decide which of several available resources to use in the presence of an adversary (called Controller) that can prevent the Actor of receiving information on the state of some of the resources. The Controller has a limitation on the amount of information it can conceal. What information should it deny from the the Actor? How should the Actor choose a resource as a function of the statistics of the states of the resources and of the non-concealed information on the state of the others. We formulate this problem as a non-zero sum game and transform it into an equivalent zero-sum game. We then propose ways to compute the most harmful behavior of the Controller as well as the best choice of a resource for the Actor, and analyse their complexity. We identify cases in which the exact solution is computationally intractable, and provide approximate solutions with polynomial complexity. We present many motivating examples and explore numerically the performance of the approximations.**

I. Introduction

Exchange of information among different entities forms the basis of most technological advances in the information era and also of social interactions. Several seminal advances in communication systems have lead to schemes that maximize the rate of exchange of information. An aspect that has received somewhat less attention, and is as important, is that of designing a framework for deciding what information should be revealed and what should be concealed during exchange of information among different entities so as to maximize their utilities. The main challenge towards developing such a framework is that oftentimes such decisions depend on the objective for exchange of information, and hence can only be determined on a case by case basis. The contribution of this paper is to develop a rigorous framework to study such questions.

We consider a system with two adversarial entities. The system's state is a random vector of dimension n. At a given time the first entity (*Controller*) has complete information about the state of the system; it can then hide some information from the second entity. It is assumed however that it must reveal a certain "minimum" amount of the information. It can choose the nature of the information it reveals subject to satisfying the above constraint. The second entity (*Actor*) takes certain actions based on the information it acquires about the system, and the actions are associated with certain utilities for both the Controller and the Actor which also depend on the state of the system. The same actions and states fetch different utilities for the Controller and Actor, but usually when one entity has a high utility the other has a low utility. We devise a framework that enables the Controller to decide the information it would conceal so as to maximize its own utility, and the Actor to determine its actions based on the information it has about the system so as to again maximize its utility.

A. Motivation and Challenge

a) Cognitive radio networks: A transmitter has access to n channels, whose qualities constitute the state of the system. It needs to select one channel for transmission, and the transmission quality of the selected channel determines the rate of successful transmission. Hence, the transmitter probes the channels in order to assess their qualities before it transmits any packet. A malicious entity (jammer) seeks to reduce the rate of successful transmission. Jamming is often done by generating signals that interfere with the sender's communication; however the jammer may be able to deteriorate the transmission rate much more by preventing the transmitter from learning the states of the channels. We thus consider the case where the jammer blocks the probe packets and not the actual transmission. We assume that the jammer knows the quality of the channels and can block the probes in at most k channels. Hence, the states of at least $n-k$ channels must be revealed to the transmitter. The transmitter selects the channel after it learns about the states of the channels the jammer reveals. Note that it may or may not select a channel whose state has been revealed since the fact that the jammer has concealed the state of a channel may indicate that the transmission quality of the corresponding channel is good. The rate of successful transmission attained by the transmitter determines the utility of the transmitter and the jammer. The ICG will enable the jammer (Controller) to optimally determine which channels it would conceal, and the transmitter (Actor) to select the channel.

b) Query resolution networks: Consider a client that needs to locate a desired information. It queries some data bases to determine which of them has the information. The responses constitute the state of the system and specify the probability with which the requested information is present in the data base (as the search in response to such preliminary queries may not be comprehensive and also the information may be dated). The responses reach the node through a gateway that has a malicious entity which blocks some of the responses in order to undermine the information location service. The client needs to determine which database it would request the information from based on the responses to its query, and again it may choose one it received a response from or one it did not receive a response from (the latter may happen if the responses it receives reveal low probabilities).

The utility of the client and the malicious entity depends on the probability that the client obtains the information it is interested in. The ICG we described will enable the malicious entity (Controller) to optimally determine which responses it would suppress and the client (Actor) to determine which database it would query.

c) Buyer-Seller authentication in e-commerce: Consider an e-commerce system where a buyer and a seller are bargaining. The authentication process between them proceeds in two stages. The buyer has n pieces of information using which he can authenticate himself to the seller. He reveals limited information about k of these pieces using which the seller can complete the first stage of the authentication successfully if the buyer is who he claims to be (e.g., using some proof verification methods). Next, the seller identifies himself to the buyer, and subsequently asks about complete information for one of the n pieces which may or may not be among those that the buyer initially selects. The buyer provides the requested information and the authentication is successful if again he is who he claims to be. This two-stage authentication process allows each entity to identify himself once he has some (albeit incomplete) information about the other participant. Now, the complete information the buyer reveals about any one piece in the authenticating process may allow the seller to acquire more information about the buyer than that required for mere authentication, e.g., information about his previous transactions with other merchants, etc. This will for example allow him to bargain more effectively with the buyer once the authentication is successful. Now, the different pieces of information the buyer possesses about himself reveals different amount of information about him, and the buyer must select the k pieces in the first stage so as to minimize the additional information he finally reveals to the seller. The seller must subsequently select the piece in the second stage to acquire maximum possible information about the buyer. The ICG will enable the buyer (Controller) and the seller (Actor) to attain their respective objectives by optimally selecting the pieces in question.

d) Gambling: Consider a gambling game in which two gamblers have a common collection of N cards each of which can have one of m colors. They randomly select a number for each card and write the chosen number on one side of the corresponding card. Subsequently, they put all cards in a bin, and the second gambler draws n cards randomly from the bin without observing the numbers on them. The first gambler then observes the colors and the numbers of the cards drawn and tells the second the numbers and the colors of k of these cards, and only the colors of the rest of the cards. The second gambler needs to select one of these n cards (either a card whose number it knows or one whose number it does not know), and the first pays him an amount that equals the number on the selected card (if this number is negative then the second pays the first). The first gambler (the Controller) needs to select the k cards so as to minimize the amount it pays, and the second needs to select a card so as to maximize the amount it receives.

e) Security systems: Consider a corrupt employee who sells secrets about the company's security system to some burglars. The building in which the company is located has n gates, and the employee knows the offered or in order to conceal his collusion in the event of an enquiry the employee informs the burglar information about only k of these gates. He also decides to select the gates whose information he reveals so as to minimize the probability that the break in is successful since if there is a successful break-in a comprehensive enquiry is likely to be launched. The burglar can break in through one among the n gates, and selects this gate based on the information he obtains from the employee so as to maximize his chance of success. In both these examples, the ICG we described will enable the Controller (first gambler or employee) and the Actor (second gambler or burgler) to attain their objectives by making appropriate selections.

An important challenge in designing a framework for modeling ICG, which incidentally recurs in all the above examples, is that the Actor can learn about the system from not only the information that the Controller reveals, but also from the information that the Controller conceals since the fact that an information has been concealed may provide important insight about its nature. Thus, the Actor must determine its optimal action so as to exploit the information contained in both of the above, and the Controller must determine what it should reveal considering that the Actor will learn from both the above.

B. Contribution and Related work

Our first contribution is to formulate the ICG as a stochastic leader-follower game (Sec II-A) and to transforom it into an equivalent zero-sum game (Sec III-A). This allows the equilibrium of the original game to be computed using a linear program (LP). The equivalence is surprising as (i) the Controller and Actor have different amount of information about the system, and (ii) Unlike standard leader-follower games, randomization is needed here. Our second contribution consists of proposing various computation algorithms, complexity analysis and approximations.

A related zero-sum repeated game of revealing/concealing information has been considered by Aumann and Maschler at [1]. There too, one player has full state infnormation and the other has only partial state information (i.e. which game is played). In contrast to our model, the informed player does not directly control the information available to the other player. It is by observing the actions of the informed player that the other one obtains some state information. Hiding information has also been studied as a tool for authentication of images or voice files. For a rich survey on this type of information hiding, see [2].

II. A Mathematical Framework

We next formulate the ICG as a stochastic leader-follower game and develop appropriate solution concepts for such games (Sec II-A).

A. Terminologies and Solution Concepts

We start by modeling the ICG as a stochastic leader-follower game between two players: the Controller and the Actor. We describe the game in both the normal form as well as in the strategic form. Let $\mathcal{N} = \{1, \ldots, n\}$.

System state: The state of the system is an n-dimensional vector $\vec{X}$ whose entries take values in $\mathcal{K} = \{0, ..., K-1\}$. The state space is $\mathcal{K}^n$. The random variables corresponding to the components of the state vector may be dependent and can be described by a joint probability distribution β.

Information of the Controller: The Controller knows the system state vector $\vec{X}$, and thereby has full information.

Actions of the Controller: The controller conceals the values of at most k components of the system state vector from the actor; it decides which components it would conceal based on its information. Thus, the controller's action is a subset of $\mathcal{N}$ with cardinality k or lower. Note that each such action determines a sub-vector of the system state, with size $n-k$ or more, that the actor observes. Let $\mathcal{A}_c(\vec{x})$ denote the set of all such sub-vectors when the controller's information (i.e., the system state vector) is $\vec{x}$, and $\mathcal{A}_c = \cup_{\vec{x} \in \mathcal{K}^n} \mathcal{A}_c(\vec{x})$.

Information of the Actor: The Actor knows the states of those components of the system state vector which the Controller does not conceal. Specifically, if c be the action taken by the Controller and the system state is $\vec{x}$, then the Actor's information $\vec{y}$ consists of the sub-vector of $\vec{x}$ with components in $N \setminus c$. Therefore, from its information $\vec{y}$, the Actor knows the Controller's action (i.e., the subset of components $a(\vec{y})$ the Controller conceals). Let $\mathcal{I}_a$ be the set of all possible informations of the Actor about the entries of the system state vector. It consists of at least $|\mathcal{A}_c| \times K^{n-k}$ elements.

Actions of the Actor: The Actor selects one of the components of the system state vector. Thus, its action is an integer l where $1 \leq l \leq n$. Thus, $\mathcal{N}$ is the set of all actions of the Actor.

Payoff function: If a component of the system state vector has value i, then the expected utility associated with that component is $r(i)$ such that $r(0) < r(1) < \ldots < r(K-1)$. If the system state is $\vec{x}$, and the Actor selects component l, then the payoff for the Actor is $r(x_l)$.

Strategies: *Behavioral strategy* of a player is a function from its information set to the set of probability measures over its action space. More precisely, the Controller can decide randomly which components to conceal based on the system state vector, and the Actor can randomly select a component based on the revealed sub-vector. Let u (v, resp.) be a behavioral policy of the Controller (Actor, resp.). Then, $u(\vec{x})$ ($v(\vec{y})$, resp.) is the probability distribution used by the Controller (Actor, resp.) for selecting its actions when its information is $\vec{x}$ ($\vec{y}$, resp.). Specifically, $u(\vec{x})_{\vec{y}}$ ($v(\vec{y})_i$, resp.) is the probability with which the Controller (Actor, resp.) conceals the sub-vector $\vec{y} \in \mathcal{A}_c(\vec{x})$ (selects the component $i \in \mathcal{N}$, resp.) when its information is $\vec{x}$ ($\vec{y}$, resp.). Let $\mathcal{U}$ ($\mathcal{V}$, resp.) be the set of behavioral strategies for the Controller (Actor, resp.).

Pure strategies: Let $\mathcal{U}^p \subset \mathcal{U}$ ($\mathcal{V}^p \subset \mathcal{V}$, resp.) be the set of pure (deterministic) behavioral policies for the Controller (Actor, resp.). A Controller's pure policy is a function from $\mathcal{K}^n$ to $\mathcal{A}_c$. An Actor's pure policy is a function from $\mathcal{I}_a$ to $\mathcal{N}$.

Mixed strategies: A mixed strategy of a player is a probability measure over its pure policies. Let $\mathcal{U}^M$ ($\mathcal{V}^M$, resp.) be the set of mixed strategies for Controller (Actor, resp.).

Note that behavioral and mixed are alternate representations of the randomized policies of the two players.

Probability space: A joint probability distribution β for the system state vector and strategies u and v for the Controller and Actor, resp., define a probability $P_\beta^{u,v}$ measure over the state, actions and informations of the two players. Let $E_\beta^{u,v}$ be the corresponding expectation operator.

Utility of the Actor: The Actor's utility is its expected payoff conditioned on its information, and is therefore a function of its information. Specifically, when the Actor's information is $\vec{y}$, the Controller and the Actor use (behavioral or mixed) strategies u and v resp., and the joint probability distribution of the system states is β, the Actor's utility $J_a^{u,v,\beta}(\vec{y})$ is given by $J_a^{\beta,u,v}(\vec{y}) = E_\beta^{u,v}[r(X_B)|\vec{Y}_a = \vec{y}]$, where $\vec{Y}_a$ is the random information of the Actor, X_i is the random state of the ith component of the system state vector, B is the action of the Actor. Hence, X_B is the random state of the component which is chosen possibly in a random way by the Actor.

Utility of the Controller: The Controller's utility is the negative of the expected payoff of the Actor conditioned on the Controller's information, and is therefore again a function of the Controller's information. Specifically, when the system state vector is $\vec{x}$, and the Controller and the Actor use (behavioral or mixed) strategies u and v resp., the Controller's utility $J_c^{u,v}(\vec{x})$ is given by $J_c^{u,v}(\vec{x}) = -E^{u,v}[r(x_B)|\vec{X} = \vec{x}]$, where $\vec{X}$ is the random system state vector, x_B is the Bth component of $\vec{x}$, B is (potentially random) action of the Actor when the system state is $\vec{x}$ and the Controller and the Actor use (behavioral or mixed) strategies u and v. Note that β is not used explicitly in computing the above expectation. If however u, v depend on β, the value of this expectation may depend on β.

Controller's and Actor's goals: The Controller and the Actor seek to maximize their respective utilities $J_c^{u,v}(\vec{x})$, $J_a^{\beta,u,v}(\vec{y})$ for all values of their respective informations $\vec{x}, \vec{y}$.

Since the Controller's and Actor's utilities are functions and not numbers, we can not use Nash equilibrium as a solution concept (unless we define an ordering between vectors). We however use related solution concepts, that of, *point-wise Nash equilibrium*, which we define next.

Definition 2.1: Let u^* and v^* be behavioral or mixed strategies of the Controller and Actor resp.. Then (u^*, v^*) is a point-wise Nash equilibrium if the following two conditions hold:

- for each system state vector $\vec{x}$ such that $\beta(\vec{x}) > 0$, $u^*(\vec{x})$ is a best response of the Controller against v^* of the Actor, i.e., $u^*(\vec{x})$ maximizes $J_c^{u,v^*}(\vec{x})$ among all strategies u of the Controller, and
- for each information $\vec{y}$ of the Actor which occurs with positive probability under β, u^*, $v^*(\vec{y})$ is a best response of the Actor against u^* of the Controller, i.e., $v^*(\vec{x})$ maximizes $J_a^{\beta,u^*,v}(\vec{y})$ among all strategies v of the Actor.

B. Elucidating examples

We now elucidate the above terminologies using the examples in Sec I-A.

In cognitive radio networks the system state vector constitutes the states of the channels, each channel can be in K

states, and $r(i)$ is the expected rate of successful transmission of the transmitter (Actor) when it transmits in a channel that is in state i. The jammer's (Controller's) action is to conceal the states some ($\leq k$) channels and the transmitter's action is to select a channel for transmission. An example of pure behavioral strategy of the jammer (denoted as *Greedy for Controller* or GC), is to conceal the channels with k-best states, that is, those with k-best expected rates of successful transmission (ties are broken in any pre-determined order). An example of pure policy of the transmitter, (denoted as *Best Among Revealed for Actor* or BRA), is to select the channel that has the highest state among the revealed channels (ties are broken in any pre-determined order). An example of behavioral policy of the jammer that is not pure is to conceal the states of as many channels that are in state $K-1$ as possible (subject to concealing the states of at most k channels), and if fewer than k channels are in state $K-1$ then select the remaining channels whose states are to be concealed uniformly among the channels that are not in state $K-1$. An example behavioral policy of the transmitter (denoted as *Uniform among Concealed for Actor* or UCA) that is not pure is to select a channel for transmission uniformly among those whose states are concealed. Next, $J_c^{u,v}(\vec{x})$ is the negative of the expected rate of successful transmission of the transmitter when the channel state vector is $\vec{x}$ and the jammer and transmitter use policies u, v resp. Also, $J_a^{\beta,u,v}(\vec{y})$ is the expected rate of successful transmission of the transmitter when the jammer reveals $\vec{y}$ to the transmitter, jammer and transmitter use policies u, v resp. and the joint distribution of the channel state vector is β. For example, let u be GC and v be UCA. Then $J_c^{u,v}(\vec{x}) = -\frac{\max S\subseteq N, |S|=k \sum_{i\in S} x_i}{k}$, and $J_a^{\beta,u,v}(\vec{y}) = \frac{\sum_{i\in a(\vec{y})} \mathbf{E}(X_i|\vec{y})}{k}$ (note that the conditional expectation in the latter depends on β). If the transmitter uses BRA, GC is the best response of the jammer, and if the state processes of the channels are independent and identically distributed, UCA is the best response of the transmitter against the GC policy of the jammer .

In the authentication example for e-commerce, the seller (Actor) may have different bargaining powers associated with different informations it can learn about the buyer (Controller), and the buyer may not know the seller's bargaining power associated with any piece even though he knows the details about the piece. This is because different sellers may have access to different data bases and therefore may be able to extract different amount of additional information about the buyer from the same content. The buyer may however know the expected bargaining power of the seller associated with each piece of information. This scenario can be modelled by assuming that each different piece of information of the buyer can be in one of K states, and the bargaining power associated with a particular state, say i, of a piece of information is a random variable whose expectation $r(i)$ is known to both the buyer and the seller. The system state vector consists the states of the n pieces of informations the buyer has about himself. The buyer knows the system state vector (note that the knowledge of the state of a piece of information implies that the buyer knows the expected and not the exact value of the bargaining power associated with that piece). The action of the buyer is to reveal limited information about some $(n-k)$ pieces of information in the first stage of the authentication: the seller can only determine $\vec{y}$ the states of these pieces of information from the limited information the buyer reveals (since although he knows what databases he can search he does not know the details about any of these pieces). The seller's action is to select one piece for which it requests details. Next, $J_c^{u,v}(\vec{x})$ is the negative of the expected bargaining power of the seller when the system state vector is $\vec{x}$ and the buyer and the seller use policies u, v resp. Also, $J_a^{\beta,u,v}(\vec{y})$ is the expected bargaining power of the seller when it observes $\vec{y}$ in the first stage, the buyer and seller use policies u, v resp. and the joint distribution of the system state vector is β.

In the gambling game, β can be obtained from the distribution that is simultaneously used to draw the random numbers, and K is the cardinality of the support set of this original distribution. Note that the random numbers drawn may be negative; we enumerate them using K positive integers, and each such enumeration constitutes the state of a card. Thus, each card has K possible states, and $r(i)$ is the number associated with the ith state. The system state vector consists the random numbers on the cards drawn by the second gambler (Actor), and is known only to the first. The action of the first gambler (Controller) is to reveal the states of some ($\geq n-k$) of these cards, which constitutes the information $\vec{y}$ for the second and the second gambler's action is to select one card for examination of the number among those that it selected initially. Next, $J_c^{u,v}(\vec{x})$ is the negative of the expectation of the random number on the card the second finally (potentially randomly) selects for examination when the system state vector is $\vec{x}$ and the gamblers use policies u, v resp. Also, $J_a^{\beta,u,v}(\vec{y})$ is the expectation of the number on the card the second finally selects for examination, when it observes $\vec{y}$, the gamblers use policies u, v resp. and the joint distribution of the system state vector is β.

The query resolution network and the security systems are similar to the cognitive radio network. In the former, the system state vector constitutes the states of the databases, each database can be in K states, and $r(i)$ is the probability that the information sought is in a database that is in state i. In the latter, the system state vector constitutes the states of the gates (e.g., the number of guards at each gate), each gate can be in K states, each state represents a level of efficacy of the security system at the gate and $r(i)$ is the probability that the burglar will successfully break in through a gate in state i.

C. Counter-intuitive properties of the Nash equilibrium

We now demonstrate that the point-wise Nash equilibrium exhibits several counter-intuitive properties which suggests that the point-wise Nash equilibrium may not always consist of simple policies that can be represented in closed form. This in turn motivates the design of efficient frameworks for computing the point-wise Nash equilibrium, which is the focus of the next two sections.

Consider the GC policy for the Controller (Sec II-B). Intuitively, it seems that GC minimizes the efficacy of the Actor and therefore there always exists some policy v^* for

the Actor such that (GC, v^*) constitutes a point-wise nash equilibrium. The following lemma shows that this intuition is unfounded, even when the joint probability distribution β is such that the state processes for different components are mutually independent and identically distributed (i.e., even when all channels are i.i.d. in cognitive radio networks).

Lemma 2.1: [4] There may not exist any policy v^* for the Actor such that (GC, v^*) constitutes a point-wise Nash equilibrium, even in systems where the state processes for different components are mutually independent and identically distributed.

We provide numerical evidence in Section V to the Lemma.

Next, consider a simple policy " Statistically Best for Actor" (SBA) for the Actor under which it decides its action without exploiting any knowledge of the Controller's policy. Specifically it selects the component i for which the expectation of the utility $(r(X_i))$ conditioned on the states of channels whose states have been revealed is the maximum under β (it uses only β and not the Controller's policy in determining the above conditional expectation). For example, when the state processes of all components are mutually independent, $K = 2$ (i.e., each component has 2 states), if the state of a component that is in state 1 has been revealed, SBA selects the component and otherwise SBA selects the component for which the expected reward is the maximum under the prior distribution β. It may seem that at least in simple special cases, i.e., when $K = 2$, there always exists some policy u^* for the Actor such that (u^*, SBA) constitutes a point-wise Nash equilibrium. The following lemma shows that such intuition is founded.

Lemma 2.2: [4] There may not exist any policy u^* for the Controller such that (u^*, SBA) constitutes a point-wise Nash equilibrium, even in systems where the state processes for different components are mutually independent and $K = 2$.

III. A COMPUTATIONAL FRAMEWORK FOR POINT-WISE NASH EQUILIBRIUM

The ICG is not a two-person zero-sum game as the arguments of the Controller's and Actor's utility functions have different dimensions, and hence the sum of these functions is not well-defined. Nevertheless, we identify a zero-sum game such that a policy pair (u, v) constitutes a point-wise Nash equilibrium in the original game if and only if it constitutes a saddle-point in the equivalent game.

A. An equivalent two-person zero-sum game

Definition 3.1: Consider a game with two players: the Controller and the Actor. The action of each player now is to select one of its pure behavioral policies in the stochastic leader-follower game described in the previous section. When the two players select policies u, v resp., the utility of the Actor under the joint probability distribution β for the system states is given by

$$R_\beta^{u,v} = E_\beta^{u,v}[r(X_B)] = \sum_{\vec{x}\in\mathcal{K}^n} \beta(\vec{x}) E_\beta^{u,v}[r(x_B)|\vec{X} = \vec{x}]. \quad (1)$$

where B is the action of the transmitter under policies u, v and random system state vector $\vec{X}$. The Actor seeks to maximize its utility and the Controller seeks to minimize the Actor's utility. The game is clearly a two-person zero-sum game. Clearly,

$$R_\beta^{u,v} = -\sum_{\vec{x}\in\mathcal{K}^n} \beta(\vec{x}) J_c^{u,v}(\vec{x}) \ \forall\ u, v, \beta, \quad (2)$$

$$\text{and } R_\beta^{u,v} = \sum_{\vec{y}\in\mathcal{K}^n} \Pr^{\beta,u}(\vec{y}) J_a^{\beta,u,v}(\vec{y}) \ \forall\ u, v, \beta. \quad (3)$$

Definition 3.2: Let $\mathcal{U}$ and $\mathcal{V}$ resp. be the set of behavioral strategies of the Controller and Actor in the two-person zero-sum game. The upper and lower values, $\overline{R}_\beta, \underline{R}_\beta$ of the above game are

$$\overline{R}_\beta = \inf_{u\in\mathcal{U}} \sup_{v\in\mathcal{V}} R_\beta^{u,v} \qquad \underline{R}_\beta = \sup_{v\in\mathcal{V}} \inf_{u\in\mathcal{U}} R_\beta^{u,v}.$$

For any $u^* \in \mathcal{U}$ and $v^* \in \mathcal{V}$ we have

$$\inf_{u\in\mathcal{U}} R_\beta^{u,v^*} \le \underline{R}_\beta \le R_\beta^{u^*,v^*} \le \overline{R}_\beta \le \sup_{v\in\mathcal{V}} R_\beta^{u^*,v}. \quad (4)$$

Definition 3.3: If $\inf_{u\in\mathcal{U}} R_\beta^{u,v^*} = \sup_{v\in\mathcal{V}} R_\beta^{u^*,v}$ for some $u^* \in \mathcal{U}$ and $v^* \in \mathcal{V}$, then all inequalities in (4) hold with equality and (u^*, v^*) are called saddle point policies, u^* is the saddle-point policy of the Controller, v^* is the saddle-point policy of the Actor, and $R_\beta^{u^*,v^*}$ is denoted as the value of the game.

Two-person zero-sum games with finitely many pure strategies for each player, have a saddle point within the mixed strategies which can be computed using linear programs (LPs) (see footnote at Section III-B) For each player, there is a one-to-one correspondence between the class of its behavioral and its mixed strategies [3] such that for any policy of the other player, the expected utility under the mixed strategy and the equivalent behavioral one is the same. Thus, a saddle point exists within the behavioral policies as well.

Note that a pure policy for any player in this game is to select a particular action which corresponds to a specific pure policy in the original game. Thus, there is a one-to-one correspondence between the sets of pure policies for each player in the two games such that for each pure policy for a player in a game the corresponding pure policy for the same player in the other game takes the same actions if presented with the same information. Since a mixed-policy for any player in any game is a probability distribution over the pure policies, there is a similar one-to-one correspondence between the sets of mixed policies for each player in the two games. Thus, it follows from the previous paragraph that there is a one-to-one correspondence between the sets of mixed policies in the zero-sum game and behavioral policies in the original game, such that for each behavioral (mixed) policy for any given player in the original (zero-sum) game, the corresponding mixed (behavioral) policy for the same player in the zero-sum (original) game has the same utility [3]. It follows using same arguments that similar correspondence exists between the sets of the behavioral policies in the two games. Thus, for notational simplicity, we use the same notations (e.g., u, v, etc.) to denote the mixed or behavioral policies in both games.

Theorem 3.1: [4] A mixed or behavioral policy pair (u^*, v^*) is a point-wise Nash equilibrium in the original game if and only if the corresponding mixed or behavioral policy pair (u^*, v^*) is a saddle point pair in the two-person zero-sum game.

Corollary 3.1: [4] A point-wise Nash-equilibrium (u^*, v^*) exists in the original game.

B. Computation of the saddle point

Saddle point of zero-sum games with finite number of strategies can be computed using LPs whose number of variables is approximately the number of pure strategies of a player and the number of constraints equal the number of pure strategies of the other player.[1] This may sound quite encouraging at first since solving LPs has polynomial complexity as a function of the number of decision variables and constraints. Neverthless, the computation is intractable due to the huge number of pure strategies N_c of the Controller and N_a of the Actor, given by[2]

$$\boxed{N_c = \left(\sum_{i=0}^{k} \binom{n}{i}\right)^{K^n} \quad \text{and} \quad N_a = n^{\sum_{i=0}^{k} \binom{n}{i} K^{n-i}}.} \tag{5}$$

Simplifying, the number of pure strategies of the Controller (Actor, resp.) in the original game is at least $\binom{n}{k}^{K^n}$ ($n^{\min(\binom{n}{(n/2)}, K^{n/2})}$, resp.).

We shall obtain alternative LPs whose *computation times are polynomials in* $(K^n + k)\binom{n}{k}$. Henceforth, u (v, resp.) are the behavioral policies of the Controller (Actor resp.).

1) Saddle point for the Controller: The following LP obtains a saddle-point policy for the Controller.

LP-CONTROLLER: Min $\sum_{\vec{y} \in \mathcal{A}_c} z(\vec{y})$ *s.t.*

$$\begin{aligned} z(\vec{y}) &\geq \sum_{\vec{x}: \vec{y} \in \mathcal{A}_c(\vec{x})} \beta(\vec{x}) r(x_i) u(\vec{x})_{\vec{y}} \\ &\qquad \forall\ i \in \mathcal{N}, \vec{y} \in \mathcal{A}_c \\ \sum_{\vec{y} \in \mathcal{A}_c(\vec{x})} u(\vec{x})_{\vec{y}} &= 1 \quad \text{for all } \vec{x} \in \mathcal{K}^n \\ u(\vec{x})_{\vec{y}} &\geq 0 \quad \forall\ \ \vec{x} \in \mathcal{K}^n, \vec{y} \in \mathcal{A}_c(\vec{x}) \end{aligned}$$

Theorem 3.2: [4] The solution $\{u(\vec{x})_{\vec{y}}\}_{\vec{y} \in \mathcal{A}_c(\vec{x}), \vec{x} \in \mathcal{K}^n}$ of LP-CONTROLLER constitutes the saddle-point policy u^* for the Controller.

[1]For example, consider a matrix game whose entries are $R^{u,v}$ where player I (minimizing) chooses a row u and player 2 (maximizing) chooses a column v. A saddle point policy for player 2 is obtained by maximizing $\left(\inf_u R^{u,v}\right)$. Then for any u, the value z is smaller than or equal to $\max_v \left(R^{u,v}\right)$. Moreover the value is the largest constant with this property. The LP is thus $\max_{p \in \Delta(\mathcal{V}^P), z} z$ *s.t.* $z \leq \sum_{v \in \mathcal{V}^P} p(v) R^{u,v},\ \forall u \in \mathcal{U}^P$. ($\Delta(\mathcal{V}^P)$ are all probability measures over $\mathcal{V}^P$).

[2](5) is obtained as follows.

- The Controller's information has K^n possible values, and for each such information it can choose $\sum_{i=0}^{k} \binom{n}{i}$ actions (note that $\sum_{i=0}^{k} \binom{n}{i}$ is the number of subsets of the components of cardinality at most k).
- The Actor's information has $\sum_{i=0}^{k} \binom{n}{i} K^{n-i}$ possible values, and for each such information it can choose n actions.

The following corollary proves an intuitive property of saddle point policies of the Controller, and will help reduce the number of variables of LP-CONTROLLER.

Corollary 3.2: [4] There exists a saddle-point policy u^* of the Controller in which it always conceals the states of k components.

Now, consider the following definition.

Definition 3.4: Let $\mathcal{A}_{c,k} = \{\vec{y} : |a(\vec{y})| = k, \vec{y} \in \mathcal{A}_c\}$ and $\mathcal{A}_{c,k}(\vec{x}) = \mathcal{A}_{c,k} \cap \mathcal{A}_{c,k}(\vec{x})$.

Due to Corollary 3.2, we only need to consider the variables $z(\vec{y})$ such that $|a(\vec{y})| = k$. Also, since for any $\vec{y}$ and $\vec{x}$ such that $\vec{y} \in \mathcal{A}_c(\vec{x})$, $x_i = y_i$ for any $i \in \mathcal{N} \setminus a(\vec{y})$, for any $\vec{y}$, $i \in \mathcal{N} \setminus a(\vec{y})$, and $y_i < \max_{j \in \mathcal{N} \setminus a(\vec{y})} y_j$, the value of the right hand side of the lower bound constraint in LP-CONTROLLER is less than or equal to that for $\vec{y}$, $l \in \mathcal{N} \setminus a(\vec{y})$, and $y_l = \max_{j \in \mathcal{N} \setminus a(\vec{y})} y_j$ irrespective of the choice of β, u. Thus, these constraints can be ignored as well, and LP-CONTROLLER can be described as follows.

LP-CONTROLLER: Minimize $\sum_{\vec{y} \in \mathcal{A}_c} z(\vec{y})$ *s.t.*

$$\begin{aligned} z(\vec{y}) &\geq \max_{i \in \mathcal{N} \setminus a(\vec{y})} r(y_i) \sum_{\vec{x}: \vec{y} \in \mathcal{A}_c(\vec{x})} \beta(\vec{x}) u(\vec{x})_{\vec{y}} \\ &\qquad \forall\ \vec{y} \in \mathcal{A}_{c,k} \\ z(\vec{y}) &\geq \sum_{\vec{x}: \vec{y} \in \mathcal{A}_c(\vec{x})} \beta(\vec{x}) r(x_i) u(\vec{x})_{\vec{y}} \\ &\qquad \forall\ i \in a(\vec{y}), \vec{y} \in \mathcal{A}_{c,k}, \\ \sum_{\vec{y} \in \mathcal{A}_{c,k}(\vec{x})} u(\vec{x})_{\vec{y}} &= 1 \quad \text{for all } \vec{x} \in \mathcal{K}^n \\ u(\vec{x})_{\vec{y}} &\geq 0 \quad \forall\ \ \vec{x} \in \mathcal{K}^n, \vec{y} \in \mathcal{A}_{c,k}(\vec{x}) \end{aligned}$$

Henceforth, we will use this description of LP-CONTROLLER. Note that LP-CONTROLLER has $K^n \binom{n}{k}$ variables and $(k+1)\binom{n}{k} + K^n + K^n \binom{n}{k}$ constraints. Thus, the computation time of this LP is polynomial in $(K^n + k)\binom{n}{k}$.

2) Saddle point for the Actor: The following LP obtains a saddle-point policy for the Actor.

LP-ACTOR: Maximize $\sum_{\vec{x} \in \mathcal{K}^n} \beta(\vec{x}) z(\vec{x})$

$$\begin{aligned} z(\vec{x}) &\leq \textstyle\sum_{i \in \mathcal{N}} v(\vec{y})_i r(x_i)\ \forall \vec{y} \in \mathcal{A}_c(\vec{x}), \vec{x} \in \mathcal{K}^n \\ v(\vec{y})_j &\geq 0 \ \ \forall\ \ \vec{y}, j \in \mathcal{N} \\ \textstyle\sum_{j \in \mathcal{N}} v(\vec{y})_j &= 1 \ \ \forall\ \vec{y} \in \mathcal{A}_c \end{aligned}$$

Theorem 3.3: [4] The solution $\{v(\vec{y})_i\}_{i \in \mathcal{N}, \vec{y} \in \mathcal{A}_c}$ of LP-ACTOR constitutes the saddle-point policy v^* for the Actor.

Definition 3.5: A policy $v \in \mathcal{V}$ of an Actor is said to be *sensible* if it never selects a component whose state has been revealed and which is in a state that is lower than the highest state among the states of all components whose states have been revealed (i.e., $v(\vec{y})_i = 0$ if $i \notin a(\vec{y})$ and $y_i \neq \max_{j \in \mathcal{N} \setminus a(\vec{y})} y_j$).

Observation 1: $R_\beta^{u,v^1} = R_\beta^{u,v^2}$ for any $u \in \mathcal{U}, v^1, v^2 \in \mathcal{V}$ such that $v^1(\vec{y})_i = v^2(\vec{y})_i$ for any $i \in a(\vec{y})$ and $\sum_{i: i \notin a(\vec{y}), y_i = j} v^1(\vec{y})_i = \sum_{i: i \notin a(\vec{y}), y_i = j} v^2(\vec{y})_i$ for each $j \in \{0, \ldots, K-1\}$.

The following corollary proves an intuitive property of saddle point policies of the Actor, and will help reduce the

number of variables of LP-ACTOR.

Corollary 3.3: [4] There exists a sensible saddle-point policy v^* of the Actor.

Due to Corollaries 3.2 and 3.3 and the above observation, we only consider variables $v(\vec{y})$ such that $|a(\vec{y})| = k$ and need to determine the components $v(\vec{y})_j$ such that $j \in a(\vec{y}) \cup j \in (\mathcal{N} \setminus a(\vec{y})) \cap \{l : y_l = \max_{m \in \mathcal{N} \setminus a(\vec{y})} y_m\}$. Thus, LP-ACTOR can be re-written as follows.

LP-ACTOR: Maximize $\sum_{\vec{x} \in \mathcal{K}^n} \beta(\vec{x}) z(\vec{x})$

$$
\begin{aligned}
z(\vec{x}) &\leq \left(1 - \sum_{i \in a(\vec{y})} v(\vec{y})_i\right) \max_{i \in \mathcal{N} \setminus a(\vec{y})} r(y_i) \\
&+ \sum_{i \in a(\vec{y})} v(\vec{y})_i r(x_i) \ \forall \ \vec{y} \in \mathcal{A}_{c,k}(\vec{x}), \vec{x} \in \mathcal{K}^n \\
v(\vec{y})_j &\geq 0, \quad \forall \ j \in a(\vec{y}), \ \vec{y} \ \in \mathcal{A}_{c,k} \\
\sum_{j \in a(\vec{y})} v(\vec{y})_j &\leq 1, \quad \forall \ \vec{y} \in \mathcal{A}_{c,k}
\end{aligned}
$$

Henceforth, we consider the above description for LP-ACTOR. Thus, LP-ACTOR effectively has $K^n + k\binom{n}{k}$ variables and $(K^n + k + 2)\binom{n}{k}$ constraints. Thus, the computation time of LP-ACTOR is polynomial in $(K^n + k)\binom{n}{k}$.

IV. PERFORMANCE GUARANTEES USING POLYNOMIAL TIME COMPUTATION

We have proved that the saddle-point policies can be obtained by solving LPs whose number of variables is exponential in n and polynomial in K. The saddle points can thus be computed for moderate values of n but not for large n. We therefore focus on obtaining provable performance guarantees using polynomial time computable policies. We first consider the important special case where the system consists of few classes of components such that all components in each class are statistically identical and the number of states K is small. We prove that the saddle point policies can be computed in polynomial time in such systems and obtain approximating policies computable in almost linear time.

A. Polynomial time computation in systems with constant number of classes and of states

We first formally define the notion of classes of components.

Definition 4.1: Let $\vec{x}^{i,j} \in \mathcal{K}^n$ be obtained by interchanging the ith and the jth components of $\vec{x} \in \mathcal{K}^n$. Let $\vec{y}^{i,j} \in \mathcal{A}_c$ be obtained as follows: (a) if $i, j \notin a(\vec{y})$ $a(\vec{y}^{i,j}) = a(\vec{y})$, $y_i^{i,j} = y_j$, $y_j^{i,j} = y_i$, $y_l^{i,j} = y_l$, $l \notin a(\vec{y}) \cup \{i, j\}$ (b) if $i \in a(\vec{y})$, $j \notin a(\vec{y})$, then $a(\vec{y}^{i,j}) = a(\vec{y}) \cup \{j\} \setminus \{i\}$, $y_i^{i,j} = y_j$, $y_l^{i,j} = y_l$, $l \notin a(\vec{y}^{i,j}) \cup \{i\}$, (c) if $i \notin a(\vec{y})$, $j \in a(\vec{y})$, then $a(\vec{y}^{i,j}) = a(\vec{y}) \cup \{i\} \setminus \{j\}$, $y_j^{i,j} = y_i$, $y_l^{i,j} = y_l$, $l \notin a(\vec{y}^{i,j}) \cup \{j\}$, (d) $\vec{y}^{i,j} = \vec{y}$, otherwise.

Definition 4.2: Components i, j are said to be in the same class if $\beta(\vec{x}) = \beta(\vec{x}^{i,j})$ for all $\vec{x} \in \mathcal{K}^n$. Note that the membership in the same class constitutes an equivalence relation and hence the classes constitute a partition of $\mathcal{N}$. Let the system consist of M classes, where $1 \leq M \leq n$. The classes are numbered as $1, \ldots, M$, and n_i components are in class i where $\sum_{i=1}^{M} n_i = n$. Let $a(\vec{y}, i)$ be the set of components in class i that have been concealed when the Actor's information is $\vec{y}$. Note that $a(\vec{y}) = \cup_{i=1}^{M} a(\vec{y}, i)$.

We shall take advantage of the fact that systems that have large number of components often have small or moderate number of classes of components and states. We first present a key property of systems with arbitrary number of classes of components.

1) Symmetry among components in the same class:

Definition 4.3: Let u, v be behavioral policies of the Controller and Actor resp. and $i, j \in \mathcal{N}$. The mirror image w.r.t (i, j) of the policy u (v, resp.), $u^{i,j}$ ($v^{i,j}$, resp.) is a policy obtained as follows: $u^{i,j}(\vec{x})_{\vec{y}} = u(\vec{x}^{i,j})_{\vec{y}^{i,j}}$ ($v^{i,j}(\vec{y})_i = v(\vec{y}^{i,j})_j$ and $v^{i,j}(\vec{y})_j = v(\vec{y}^{i,j})_i$, resp.).

Intuitively, $u^{i,j}$ ($v^{i,j}$, resp.) treat i as j and j as i.

Definition 4.4: A policy $u \in \mathcal{U}$ ($v \in \mathcal{V}$, resp.) is said to be symmetric w.r.t. (i, j) if $u = u^{i,j}$ ($v = v^{i,j}$, resp.). A policy $u \in \mathcal{U}$ ($v \in \mathcal{V}$, resp.) is said to be symmetric if it is symmetric w.r.t. each pair of components that are in the same class. Let $\mathcal{U}^s \subset \mathcal{U}$ and $\mathcal{V}^s \subset \mathcal{V}$ be the classes of all symmetric policies of the Controller and Actor resp..

The following theorem shows the existence of a symmetric saddle-point.

Theorem 4.1: [4] There exists a symmetric policy $u \in \mathcal{U}^s$ ($v \in \mathcal{V}^s$, resp.) for the Controller (Actor, resp.) such that u (v, resp.) constitutes a saddle-point of the Controller (Actor, resp.).

2) Additional Terminologies:

Definition 4.5: Let $\mathbf{l}(\vec{x})$ be a matrix with M rows and K columns and entries in $0, \ldots, n$ such that $l(\vec{x})_{i,j}$ is the number of components of $\vec{x}$ that are in class i and state j. Let $\mathcal{L} = \{\mathbf{l} : \mathbf{l}(\vec{x}) = \mathbf{l}, \vec{x} \in \mathcal{K}^n\}$. Let $\mathbf{m}(\vec{y})$ be a matrix with M rows and K columns with entries in $0, \ldots, a(\vec{y})$ such that $m(\vec{y})_{i,j}$ is the number of components of $\vec{y}$ that are in class i and state j. Let $\mathcal{M}_{\vec{x}} = \{\mathbf{m} : \mathbf{m}(\vec{y}) = \mathbf{m}, \vec{y} \in \mathcal{A}_{c,k}(\vec{x})\}$. Note that $\mathcal{M}_{\vec{x}^1} = \mathcal{M}_{\vec{x}^2}$ if $\mathbf{l}(\vec{x}^1) = \mathbf{l}(\vec{x}^2)$. Let $\mathcal{M}_{\mathbf{l}} = \cup_{\vec{x} \in \mathcal{K}^n, \mathbf{l}(\vec{x}) = \mathbf{l}} \mathcal{M}_{\vec{x}}$. Let $\mathcal{M} = \cup_{\mathbf{l} \in \mathcal{L}} \mathcal{M}_{\mathbf{l}}$.

With slight abuse of notation, we have used $\mathbf{l}, \mathbf{m}$ to denote both the functions and the values of the functions as well - the implication of specific usages are clear from the context.

Note that (a) $|\{\vec{y} : \mathbf{m}(\vec{y}) = \mathbf{m}, \vec{y} \in \mathcal{A}_{c,k}(\vec{x})\}|$ depends on $\vec{x}$ only through $\mathbf{l}(\vec{x})$. and (b) $|\{\vec{x} : \mathbf{l}(\vec{x}) = \mathbf{l}, \vec{y} \in \mathcal{A}_c(\vec{x})\}|$ depends on $\vec{y}$ only through $\mathbf{m}(\vec{y})$. Thus, we can introduce the following definitions.

Definition 4.6: Let $\Theta_1(\mathbf{l}, \mathbf{m})$ denote for one (representative) $\vec{x}$ such that $\mathbf{l}(\vec{x}) = \mathbf{l}$ the number of $\vec{y}$ in $\mathcal{A}_{c,k}(\vec{x})$ such that $\mathbf{m}(\vec{y}) = \mathbf{m}$. Let $\Theta_2(\mathbf{l}, \mathbf{m})$ denote is the number of system state vectors $\vec{x}$ such that (a) $\mathbf{l}(\vec{x}) = \mathbf{l}$ and (b) $\vec{y} \in \mathcal{A}_c(\vec{x})$ for one (representative) $\vec{y}$ such that $\mathbf{m}(\vec{y}) = \mathbf{m}$. Let $\Theta_3(\mathbf{m}) = |\vec{y} \in \mathcal{A}_{c,k} : \mathbf{m}(\vec{y}) = \mathbf{m}|$, and $\Theta_4(\mathbf{l}) = |\vec{x} \in \mathcal{K}^n : \mathbf{l}(\vec{x}) = \mathbf{l}|$.

Note that both $\Theta_2(\mathbf{l}, \mathbf{m})\Theta_3(\mathbf{m})$ and $\Theta_1(\mathbf{l}, \mathbf{m})\Theta_4(\mathbf{l})$ constitute the number of tuples $(\vec{x}, \vec{y})$ such that $\vec{x} \in \mathcal{K}^n, \vec{y} \in \mathcal{A}_{c,k}(\vec{x})$ and $\mathbf{l}(\vec{x}) = \mathbf{l}, \mathbf{m}(\vec{y}) = \mathbf{m}$. Thus, $\Theta_2(\mathbf{l}, \mathbf{m})\Theta_3(\mathbf{m}) = \Theta_1(\mathbf{l}, \mathbf{m})\Theta_4(\mathbf{l})$

Definition 4.7: Let $R_1(\mathbf{m}) = \max_{j:m_{i,j}>0} r(j)$, and set

$$R_2(\mathbf{l},\mathbf{m},i) \quad = \quad \sum_{j=0}^{K-1} r(j)\frac{l_{i,j}-m_{i,j}}{n_i-\sum_{j=0}^{K-1} m_{i,j}}.$$

Note that $R_1(\mathbf{m})$ is the expected reward the Actor obtains when its information is $\vec{y}$ such that $\mathbf{m}(\vec{y}) = \mathbf{m}$ and it selects a component whose state has been revealed and whose state is the highest among those of the components whose states have been revealed. Also, $R_2(\mathbf{l},\mathbf{m},i)$ is the expected reward the Actor obtains when its information is $\vec{y}$ such that $m(\vec{y}) = \mathbf{m}$, the system state is $\vec{x}$ such that $\mathbf{l}(\vec{x}) = \mathbf{l}$ and it selects a component of class i uniformly among $a(\vec{y},i)$.

Definition 4.8: Let $\mathcal{C}(\mathbf{m})$, $1 \leq |\mathcal{C}(\mathbf{m})| \leq \min(k,M)$, be the set of classes for which at least one component's state has been concealed when the Actor's information $\vec{y}$ is such that $\mathbf{m}(\vec{y}) = \mathbf{m}$. Let $\Phi(\mathbf{m},i)$ be the number of components of class i that have been concealed when the Actor's information $\vec{y}$ is such that $\mathbf{m}(\vec{y}) = \mathbf{m}$. Note that $\Phi(\mathbf{m},i) = \sum_{j=0}^{K-1} m_{i,j}$, and $|\mathcal{C}(\mathbf{m})| = \sum_{i=1}^{M} \min\left(\Phi(\mathbf{m},i),1\right)$.

Finally, note that since $\beta(\vec{x}) = \beta(\vec{x}^{i,j})$ for all i,j that are in the same class, $\beta(\vec{x}^1) = \beta(\vec{x}^2)$ if $\mathbf{l}(\vec{x}^1) = \mathbf{l}(\vec{x}^2)$.

Definition 4.9: Let $\beta'(\mathbf{l})$ denote $\beta(\vec{x})$ for some (representative) $\vec{x} \in \mathcal{K}^n$ such that $\mathbf{l}(\vec{x}^1) = \mathbf{l}$, and $\beta''(\mathbf{l}) = \Theta_4(\vec{l})\beta'(\vec{l})$.

3) Polynomial time computation of saddle point policy of Controller for constant K, M: We now consider the simplification of LP-CONTROLLER.

Note that u is symmetric if and only if $u(\vec{x}^1)_{\vec{y}^1} = u(\vec{x}^2)_{\vec{y}^2}$ whenever the following conditions hold: (a) $\mathbf{l}(\vec{x}^1) = \mathbf{l}(\vec{x}^2)$, (b) $\mathbf{m}(\vec{y}^1) = \mathbf{m}(\vec{y}^2)$ (c) $\vec{y}^1 \in \mathcal{A}_c(\vec{x}^1)$, $\vec{y}^2 \in \mathcal{A}_c(\vec{x}^2)$. Let $u'(\mathbf{l})_{\mathbf{m}}$ denote $u(\vec{x})_{\vec{y}}$ for some (representative) $\vec{x} \in \mathcal{K}^n, \vec{y} \in \mathcal{A}_{c,k}(\vec{x})$ such that $\mathbf{l}(\vec{x}) = \mathbf{l}$, $\mathbf{m}(\vec{y}) = \mathbf{m}$. Thus, each $u \in \mathcal{U}^s$ is uniquely described by $u^s(\mathbf{l})_{\mathbf{m}}$ where $u^s(\mathbf{l})_{\mathbf{m}} = \Theta_1(\mathbf{l},\mathbf{m})u'(\mathbf{l})_{\mathbf{m}}$.

We now state LP-CONTROLLER-CLASS that computes $\{u^s(\mathbf{l})_{\mathbf{m}}\}$ for the symmetric saddle point strategy of the Controller.

LP-CONTROLLER-CLASS: $Min \sum_{\mathbf{m}\in\mathcal{M}} \eta(\vec{m})$ $s.t.$

$$\begin{aligned}
\forall\ \mathbf{m} \in \mathcal{M},\ \eta(\mathbf{m}) &\geq R_1(\mathbf{m}\in\mathcal{M}) \sum_{\mathbf{l}:\mathbf{m}\in\mathcal{M}_\mathbf{l}} \beta''(\mathbf{l}) u^s(\mathbf{l})_{\mathbf{m}} \\
\eta(\mathbf{m}) &\geq \sum_{\mathbf{l}:\mathbf{m}\in\mathcal{M}_\mathbf{l}} \beta''(\mathbf{l}) u(\mathbf{l})_{\mathbf{m}} R_2(\mathbf{l},\mathbf{m},i) \\
&\qquad \forall\ \mathbf{m}\in\mathcal{M}, i \in \mathcal{C}(\mathbf{m}) \\
\textstyle\sum_{\mathbf{m}\in\mathcal{M}_\mathbf{l}} u(\mathbf{l})_{\mathbf{m}} &= 1 \quad \text{for all } \mathbf{l} \in \mathcal{L} \\
u(\mathbf{l})_{\mathbf{m}} &\geq 0 \quad \forall\ \ \mathbf{m} \in \mathcal{M}_\mathbf{l}, \mathbf{l} \in \mathcal{L}.
\end{aligned}$$

Theorem 4.2: [4] The solution $\{u^s(\mathbf{l})_{\mathbf{m}}\}_{\mathbf{m}\in\mathcal{M}_\mathbf{l},\mathbf{l}\in\mathcal{L}}$ of LP-CONTROLLER-CLASS constitutes a policy u^* for the Controller such that $u^* \in \mathcal{U}^s$ and u^* is a saddle-point policy for the Controller.

Thus, LP-CONTROLLER-CLASS has $O(n^{2KM})$ variables and $O(n^{2KM})$ constraints. Thus, the computation time of LP-CONTROLLER-CLASS is polynomial in n and exponential in K, M, and hence polynomial in n if K, M are constants.

4) Polynomial time computation of saddle point policy of Actor for constant K, M: We now consider the computation of a symmetric saddle point strategy for the Actor. Note that the Actor's policy v is symmetric if and only if $v(\vec{y}^1)_i = v(\vec{y}^2)_j$ whenever the following conditions hold: (a) $\mathbf{m}(\vec{y}^1) = \mathbf{m}(\vec{y}^2)$ (b) i,j are in the same class, and (b) either (i) $i \in a(\vec{y}^1), j \in a(\vec{y}^2)$, or (ii) $i \notin a(\vec{y}^1), j \notin a(\vec{y}^2), y^1_i = y^2_j$.

Consider a $\mathbf{m} \in \mathcal{M}$ and a class $i \in \mathcal{C}(\mathbf{m})$. Then, let $v'(\mathbf{m})_i$ be the probability with which a symmetric policy v selects one (representative) component, say j, that is in class i and has been concealed, when the Actor's information state is a (representative) $\vec{y}$ such that $\mathbf{m}(\vec{y}) = \mathbf{m}$ (i.e., $v'(\mathbf{m})_i = v(\vec{y})_j$). Let $v^s(\mathbf{m})_j = \Phi(\mathbf{m},j)v'(\mathbf{m})_j$, $j \in \mathcal{C}(\mathbf{m})$, be the total probability with which a symmetric policy $v \in \mathcal{V}^s$ of the Actor selects a component which is in class j and whose state has been concealed, when the Actor's information state is a (representative) $\vec{y}$ such that $\mathbf{m}(\vec{y}) = \mathbf{m}$. Thus, v selects a component whose state has been revealed with probability $1 - \sum_{j\in\mathcal{C}(\vec{y})} v^s(\mathbf{m}(\vec{y}))_j$. From Corollary 3.3 it is sufficient to consider only sensible policies. Thus, from Observation 1, $v^s(\mathbf{m})_j$, $j \in \mathcal{C}(\mathbf{m})$ uniquely specify a symmetric saddle point strategy $v \in \mathcal{V}^s$. We prove that the following LP-ACTOR-CLASS, computes the above.

LP-ACTOR-CLASS: Maximize $\sum_{\mathbf{l}\in\mathcal{L}} \beta''(\mathbf{l})\eta(\mathbf{l})$ $s.t.$

$$\begin{aligned}
\eta(\mathbf{l}) &\leq (1-\textstyle\sum_{i\in\mathcal{C}(\mathbf{m})} v^s(\mathbf{m})_i R_1(\mathbf{m}) \\
&\quad + \textstyle\sum_{i\in\mathcal{C}(\mathbf{m})} v^s(\mathbf{m})_i R_2(\mathbf{l},\mathbf{m},i) \\
&\quad \forall \mathbf{m} \in \mathcal{M}_\mathbf{l}, \mathbf{l} \in \mathcal{L} \\
v^s(\mathbf{m})_i &\geq 0 \ \ \forall\ i \in \mathcal{C}(\mathbf{m}), \mathbf{m} \in \mathcal{M} \\
\textstyle\sum_{i\in\mathcal{C}(\mathbf{m})} v(\mathbf{m})_i &\leq 1 \ \ \forall\ \mathbf{m} \in \mathcal{M}
\end{aligned}$$

Theorem 4.3: [4] The solution $\{v^s(\mathbf{m})_j\}_{\mathbf{m}\in\mathcal{M},j\in\mathcal{C}(\mathbf{m})}$ of LP-ACTOR-CLASS constitutes a policy v^* for the Actor such that $v^* \in \mathcal{V}^s$ and v^* is a saddle-point policy for the Actor.

LP-ACTOR-CLASS has $O(n^{KM})$ variables and $O(n^{2KM})$ constraints. Thus, the computation time of LP-ACTOR-CLASS is polynomial in n and exponential in K, M.

B. Approximation guarantees using polynomial time computable policies for arbitrary systems

Saddle point strategies can be computed in polynomial time when either n is a constant (using LP-CONTROLLER or LP-ACTOR) or K, M are constants (using LP-CONTROLLER-CLASS or LP-ACTOR-CLASS). The computation however becomes intractable when two or more of these parameters are large. We now prove that simple linear ($O(n)$) or almost linear ($O(n\log n)+K$) time computable policies can provably approximate the saddle point policies.

C. Approximation guarantees using a linear time computable policy for the Actor

Consider a symmetric sensible policy, denoted as $v^{\text{UNIFORM}} \in \mathcal{V}^s$, for the Actor that (a) selects each concealed class and a revealed component with equal probabilities, i.e., $v^{s,\text{UNIFORM}}(\mathbf{m})_i = 1/\left(|\mathcal{C}(\mathbf{m})|+1\right)$ for each $\mathbf{m} \in \mathcal{M}$, $i \in \mathcal{C}(\mathbf{m})$. This uniquely describes any symmetric sensible policy since a symmetric policy selects uniformly among the concealed components in each class and a sensible policy selects only a revealed component with the highest state whenever it selects a revealed component. Clearly, the Actor

needs $O(n)$ time and memory to select a component using this policy. We now present the main result of this section.

Theorem 4.4: [4] For any β, k, n, K, M,

$$\inf_{u\in\mathcal{U}} R_\beta^{u,v^{\text{UNIFORM}}} \geq \frac{1}{\min(k,M)+1} \sup_{v\in\mathcal{V}} \inf_{u\in\mathcal{U}} R_\beta^{u,v}.$$

For $K = 2$, the approximation ratio can be improved slightly. In this case, for a symmetric sensible saddle point strategy v of the Actor, $\sum_{i\in\mathcal{C}(\mathbf{m})} v^s(\mathbf{m})_i = 1$ if all revealed components are in state 0 and $\sum_{i\in\mathcal{C}(\mathbf{m})} v^s(\mathbf{m})_i = 0$ otherwise. Using the above, it follows that the Actor's policy that selects (a) a component in state 1 if the state of at least one such component is revealed and (b) each concealed class with equal probability, otherwise, attains a $1/\min(k,M)$ approximation ratio.

D. Approximation guarantees using an almost linear time computable policy for the Controller

Consider the *Greedy for Controller* policy of the Controller where it conceals the components with k highest states and breaks ties randomly and uniformly. We denote this policy as u^{GC}. Clearly, $u^{\text{GC}} \in \mathcal{U}^s$. Note that the Controller needs $O(n\log n + K)$ time and $O(n)$ memory to decide its actions using this policy.

Theorem 4.5: [4] For any β, k, n, K, M,

$$\sup_{v\in\mathcal{V}} R_\beta^{u^{\text{GC}},v} \leq (k+1) \inf_{u\in\mathcal{U}} \sup_{v\in\mathcal{V}} R_\beta^{u,v}.$$

For any β, k, n, K such that $M = 1$, that is, all components are statistically identical,

$$\sup_{v\in\mathcal{V}} R_\beta^{u^{\text{GC}},v} \leq 2 \inf_{u\in\mathcal{U}} \sup_{v\in\mathcal{V}} R_\beta^{u,v}.$$

Note that when $K = 2$ the approximation factor turns out to be k (instead of $k+1$) for arbitrary β, k, n, M, see [4].

V. NUMERICAL EXAMPLE

Through a simple example we shall illustrate several points: (i) we show that GC is in general not a saddle-point strategy for the Controller, (ii) We illustrate the bound obtained with GC in Theorem 4.5, (iii) We show symmetrical statements for the Uniform policy for the Actor, (iv) Investigate tightness.

We consider $n = 3$ channels, $K = 3$ states per channel and $M = 1$ single class (i.e. all channels are symmetric). We have taken $k = 2$ (two channels are concealed). The probability distribution β for each channel is given by the following vector: $\beta(\overrightarrow{x}) = (1/3, 1/3, 1/3)$. The reward function considered is the following vector: $r = (0, x, 1)$, where $x = r(1)$ is the reward when the channel is in state 1. We let x vary and compare numerically in figure 1 the following:

- value of the game
- the performance when the Actor uses the uniform strategy and Controller plays optimally against it,
- the performance when the Controller uses the GC strategy and the Actor plays optimally against it,
- The bounds in Theorem 4.4
- The bounds in Theorem 4.5

The figure confirms that the GC is not a saddle-point point for the Controller and that the Uniform strategy is not a saddle-point for the Actor. However both these strategies are seen to perform well and to guarantee a performance close to the value of the ICG. The bounds given bby Theorems 4.4 and 4.5 are indeed seen to hold. The optimal strategy of the sender turned out to choose the unconcealed channel with probability 1 for $x = 0.7$ or larger. It chooses with probability 1 a concealed channel for $x = 0.3$ or loss. For $x = 0.5$ the sender's policy was to randomize between the two (it chose the unconcealed channel with probabilitly 0.4).

VI. CONCLUSIONS AND OPEN QUESTIONS

We have studied a leader-follower game where the actions of the leader (Controller) determine the information available to the follower (Actor). By concealing information, the leader degrades the performance of the follower that attempts to choose one out of several resources with the best state among all. We have provided a rich body of computation and appproximation tools for that problem along with mathematical foundations and complexity analysis.

The question of tightness of the approximation guarantees is only partially solved. The approximation bound for the uniform policy of the Actor is indeed tight [4]. The question regarding the tightness of the approximation ratio obtained for the Greedy policy of the Controller remains open. Other open problems include establishing that the computation of the saddle point policies is NP-hard, and determining whether the approximation factors can be substantially improved while using polynomial time computation. We plan to extend our study to the stochastic game framework in which the states can change in time according to some Markov structure.

REFERENCES

[1] R. J. Aumann and M. Maschler. Repeated games with incomplete information. avec la collaboration de R. Stearns. M.I.T. Press, 1995

[2] Fabien A. P. Petitcolas, Ross J. Anderson and Markus G. Kuhn, "Infnormation Hiding — A Survey" *Proceedings of the IEEE, special Issue on protection of multimedia content*, **87**(7):1062–1078, July 1999.

[3] Harold W. Kuhn, "Extensive games and the problem of information", *Contributions to the to the Theory of Games*, eds. H.W.Kuhn and A.Tucker, Vol. 2, Princeton University Press, pp. 193-216, 1953

[4] S. Sarkar, E. Altman, R. El-Azouzi and Y. Hayel, "Information concealing games in communication networks", Full version, available at http://www-sop.inria.fr/mistral/personnel/Eitan.Altman/ntkgame.html

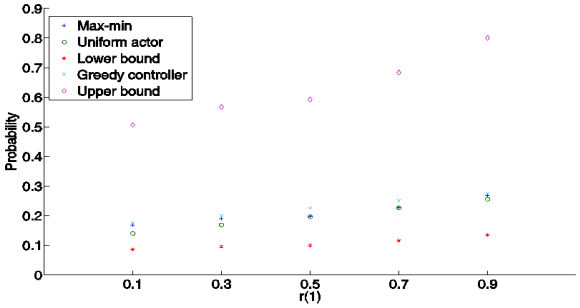

Fig. 1. Approximation performance of the uniform policy compared to the optimal.