Incentives in Effort Games

(Short Paper)

Yoram Bachrach Jeffrey S. Rosenschein
School of Engineering and Computer Science
The Hebrew University of Jerusalem, Israel
{yori, jeff}@cs.huji.ac.il

ABSTRACT

We consider *Effort Games*, a game theoretic model of cooperation in open environments, which is a variant of the principal-agent problem from economic theory. In our multiagent domain, a common project depends on various tasks; achieving certain subsets of the tasks completes the project successfully, while others do not. The probability of achieving a task is higher when the agent in charge of it exerts effort, at a certain cost for that agent. A central authority, called the principal, attempts to incentivize agents to exert effort, but can only reward agents based on the success of the entire project.

We model this domain as a normal form game, where the payoffs for each strategy profile are defined based on the different probabilities of achieving each task and on the boolean function that defines which task subsets complete the project and which do not. We view this boolean function as a simple coalitional game, and call this game the underlying coalitional game. We show that finding the minimal reward that induces an agent to exert effort is at least as hard computationally as finding the Banzhaf power index in the underlying coalitional game, so this problem is #P-hard in general.

We also show that in a certain restricted domain, where the underlying coalitional game is a unanimity weighted voting game with certain properties, it is possible to solve all of the above problems in polynomial time.

1. INTRODUCTION

The computational aspects of many game theoretic concepts have been thoroughly studied in recent years. A key issue in many such domains is constructing a proper reward scheme to achieve the desired behavior of self-interested agents.

In this paper, we deal with the computational complexity of finding a reward scheme in *effort games* in open environments. In our model, the mechanism's purpose is to incentivize agents to exert effort on a common project; each agent is in charge of a task, and can increase the probability that this task will be completed successfully, at a certain cost to that agent. We assume that the interested party, called the *principal*, cannot observe the agents' decisions about whether to expend effort, or the results of the individual tasks. The only information available to the principal is the overall result of the entire project.

Cite as: Incentives in Effort Games (Short Paper), Yoram Bachrach and Jeffrey S. Rosenschein, *Proc. of 7th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2008)*, Padgham, Parkes, Müller and Parsons (eds.), May, 12-16., 2008, Estoril, Portugal, pp. 1557-1560.

1.1 Preliminaries

We consider the effort game model using game theoretic tools. We use the solution concept of *iterated elimination of dominated strategies*, where strictly dominated strategies are removed from the game and no longer have any effect on future dominance relations. The analysis also relies on viewing the project as a cooperative game.

The heart of an effort game is a boolean function that decides which task subsets complete the project successfully, and which task subsets do not. Our results depend on viewing this boolean function as a simple coalitional game.

Definition 1. *A coalitional game is a domain that consists of a set of tasks, T, and a characteristic function mapping any subset of the tasks to a real value $v : 2^T \to \mathbb{R}$, indicating the total utility of a project that achieves exactly these tasks.*

In a *simple* coalitional game, v only gets values of 0 or 1 ($v : 2^I \to \{0, 1\}$). We say a subset $C \subset T$ *wins* if $v(T) = 1$, and say it *loses* if $v(C) = 0$. We denote the set of all subsets of tasks that win the simple game as $T_{win} = \{T' \subset T | v(T') = 1\}$. A task t is *critical* in a winning subset C if the task's removal from that coalition makes it lose: $v(C) = 1$, $v(C \setminus \{t\}) = 0$. A game is *increasing* if for all subsets $C' \subset C \subset T$ we have $v(C') \leq v(C)$. We will assume achieving more tasks is always better for the project, so games are increasing. Thus, if a certain subset of tasks $C \subset T$ wins, every superset of C also wins.

We also consider the restricted case of effort voting games. A weighted voting game is a well-known game-theoretic model of cooperation in political bodies; each agent has a weight, and a coalition of agents wins the game if the sum of the weights of its members exceeds a certain threshold.

Definition 2. *A weighted voting game is a simple coalitional game with tasks (agents) $T = (t_1, \dots, t_n)$, a vector of weights $w = (w_1, \dots, w_n)$ and a threshold q. We say t_i has the weight w_i. Given a coalition $C \subseteq T$ we denote the weight of the coalition $w(C) = \sum_{i \in \{i | t_i \in C\}} w_i$. A coalition C wins the game (so $v(C) = 1$) if $w(C) \geq q$, and loses the game (so $v(C) = 0$) if $w(C) < q$.*

A question that arises in the context of simple games, and especially in weighted voting games, is that of measuring the influence a certain task (player) has on the outcome of the game. One approach to measuring this notion is power indices, and specifically the Banzhaf power index.

DEFINITION 3. *The Banzhaf index is a power index that depends on the number of coalitions in which an agent is critical, out of all possible coalitions. It is given by* $\beta(v) = (\beta_1(v), \ldots, \beta_n(v))$ *where*

$$\beta_i(v) = \frac{1}{2^{n-1}} \sum_{S \subset T | i \in S} [v(S) - v(S \setminus \{i\})].$$

1.2 The Effort Game Model

We now define effort game models and related problems.

DEFINITION 4. *An effort game domain is a domain that consists of the following: A set of n agents,* $I = \{a_1, \ldots, a_n\}$*; a set of n tasks,* $T = \{t_1, \ldots, t_n\}$*; a simple coalitional game G with task set T, such that* $|T| = n$*, and with the value function* $v : 2^T \to \{0, 1\}$*; a set of success probability pairs* $(\alpha_1, \beta_1), (\alpha_2, \beta_2), \ldots, (\alpha_n, \beta_n)$ *so that* $\alpha_i, \beta_i \in \mathbb{R}$*, and that* $0 \leq \alpha_i \leq \beta_i \leq 1$*; a set of effort exertion costs* $c_1, \ldots, c_n \in \mathbb{R}$*, so that* $c_i > 0$.

Informally, this domain is interpreted as follows. A joint project depends on the completion of certain tasks. Achieving some of the subsets of tasks completes the project successfully, and some fail, as determined by the simple game G. An agent a_i is responsible for each such task t_i. That agent may exert effort, which gives the task a probability of β_i of being completed. However, exerting effort costs that agent a certain utility c_i. If the agent shirks (does not exert effort) the agent does not incur the cost c_i, and the task has a lower probability α_i of being completed.

We now formally describe the domain. Each of the agents $a_i \in I$ is responsible for the task $t_i \in T$. For each coalition of agents $C \subset I$, we denote the set of tasks owned by these agents $T(C) = \{t_i \in T | a_i \in C\}$. We say the common project is successful if a subset of tasks $T' \subset T$ is achieved, so that $v(T') = 1$ (so T' is winning in G). Each agent can either choose to exert effort or shirk. In the effort games model, we assume the tasks succeed or fail *independently* of one another. If a_i exerts effort, task t_i is completed with probability β_i. If it does not exert effort (and shirks instead), task t_i is completed with a lower probability $\alpha_i < \beta_i$. However, each agent has a cost for exerting effort, $c_i > 0$, which is deducted from the utility obtained by the agent. Suppose the agents in C exert effort, and that the agents in $I \setminus C$ do not. Given C we know the probability that each task is completed. If $a_i \in C$ then t_i is completed with probability β_i; if $a_i \notin C$, t_i is completed with probability α_i.

Given the coalition of agents that contribute effort, C, we denote the probability that a certain task t_i is completed as $p_i(C)$, defined as follows:

$$p_i(C) = \begin{cases} \beta_i & \text{if } t_i \in C \\ \alpha_i & \text{if } t_i \notin C \end{cases}$$

Consider a subset of tasks $T' \subset T$. Given the coalition C of agents that exert effort, we can calculate the probability that exactly the tasks in T' are the ones achieved: $\Pr_C(T') = \prod_{t_i \in T'} p_i(C) \cdot \prod_{t_i \notin T'} (1 - p_i(C))$. We can also calculate the probability that *any* winning subset of tasks is achieved, and denote this by $Pr_C(Win) = \sum_{T_w \in T_{win}} Pr_C(T_w)$.

In our model we have a central authority (called the *principal*) interested in successfully completing the common project. The principal attempts to make sure a certain subset of agents exerts effort, and needs to reward the agents so that they will exert effort despite their cost of doing so. He thus designs a *reward scheme*, but attempts to minimize his costs.

Let $C \subset I$ be the coalition of agents that have exerted effort, and T' be the set of achieved tasks. On the one hand, T' may not contain all the tasks of the agents that have exerted effort, since if $\beta_i < 1$, a task has a probability of failing even when the agent exerts effort. On the other hand, T' may contain some tasks for which agents did not exert effort, since if $\alpha_i > 0$, a task has a probability of succeeding even if an agent does not exert effort. We assume that the principal knows whether the project succeeded or not (whether $v(T') = 1$ or $v(T') = 0$), knows c_i, α_i, β_i of all agents, but does not know whether an agent a_i has exerted effort (so $a_i \in C$) or shirked (so $a_i \notin C$). Thus the principal cannot reward only those agents that have exerted effort. It can only promise each agent a_i a certain reward r_i if the project succeeds, and a reward of 0 if it does not. The principal can choose among various reward vectors $r = (r_1, \ldots, r_n)$.

Given the reward vector $r = (r_1, \ldots, r_n)$, and given that the agents that exert effort are $C \subset I$, a_i's expected reward is $e_i(C) = \sum_{T_w \in T_{win}} \Pr_C(T') \cdot r_i$. Agent a_i has a cost c_i of exerting effort. It can choose between two strategies—exert effort, or shirk. Exerting effort increases the expected reward, but has a cost c_i. If a_i shirks he does not incur the cost c_i, but his expected reward is smaller. The effort game is the normal form game obtained due to a certain reward vector r chosen by the principal.

DEFINITION 5. *An Effort Game is the normal form game* $G_e(r)$ *defined on the above domain with a simple coalitional game* G *and a reward vector* $r = (r_1, \ldots, r_n)$*, as follows.*

In $G_e(r)$ *agent* a_i *has two strategies:* $S_i = \{exert, shirk\}$*. Denote by* Σ *the set of all strategy profiles* $\Sigma = S_1 \times \ldots \times S_n$*. Given a strategy profile* $\sigma = (s_1, \ldots s_n) \in \Sigma$*, we denote the coalition of agents that exert effort in* σ *by* $C_\sigma = \{a_i \in I | s_i = exert\}$*. To fully define the game, we must also define the payoff function of each agent* $F_i : \Sigma \to \mathbb{R}$*. The payoffs depend on the reward vector* $r = (r_1, \ldots, r_n)$*: the payoff of each agent in strategy profile* σ *is his expected reward minus the cost of the effort exerted. Thus:*

$$F_i(\sigma) = \begin{cases} e_i(C_\sigma) - c_i & \text{if } s_i = exert \\ e_i(C_\sigma) & \text{if } s_i = shirk \end{cases}$$

G_e *depends on* r*, so we denote it as* $G_e(r)$.

Given a simple coalitional game G, each reward vector r defines a different effort game $G_e(r)$. In this domain, the principal may want to make sure a certain subset of the agents exert effort, and it is up to him to choose a reward vector that achieves this, under certain assumptions on the rational behavior of these agents. The strategies used by the agents are determined by a certain *game theoretic solution concept*. Such solution concepts typically define different possible strategy profiles. A reward vector that guarantees that a certain coalition C' exerts effort, under a certain solution concept, is an incentive inducing scheme for C'.[1]

Although there may be many possible incentive inducing reward vectors, the principal is self-interested, and attempts to minimize the total rewards it pays, $\sum_{i=1}^{n} r_i$.

[1]Several papers regarding "combinatorial agency" [1] have focused on a Nash equilibrium domain. We survey some of this work in Section 3. In this paper, we focus on a dominant strategy implementation, and on an iterated elimination of dominant strategy implementation.

Definition 6. *An Iterated Elimination of Dominated Strategies Incentive Inducing Scheme for C' is a reward vector $r = (r_1, \dots, r_n)$, such that in the effort game $G_e(r)$, after any sequence of eliminating dominated strategies, for any $a_i \in C'$, the only remaining strategy for a_i is to exert effort.*

We define the following relation regarding effort exertion in strategy profiles. Let D be an effort game domain, and $r = (r_1, \dots, r_n)$ be a reward vector. Let $\sigma_1, \sigma_2 \in \Sigma$ be two strategy profiles in $G_e(r)$, so $\sigma_1 = (s_{1,1}, s_{1,2}, \dots, s_{1,n})$ and $\sigma_2 = (s_{2,1}, s_{2,2}, \dots, s_{2,n})$. We say σ_1 is *more exerting* than σ_2, and denote $\sigma_1 >_e \sigma_2$, if the following holds: all the agents that exert effort in σ_2 also exert effort in σ_1, and at least one agent that exerts effort in σ_1 does not exert effort in σ_2. We can prove the following:

Theorem 1. *Let D be an effort game domain, $r = (r_1, \dots, r_n)$ be a reward vector, and a_i be an agent in that domain. Let $\sigma_1, \sigma_2 \in \Sigma$ be two strategy profiles in $G_e(r)$ so that $\sigma_1 >_e \sigma_2$, and so that $\sigma_{1,i} = \sigma_{2,i}$. Then $e_i(C_{\sigma_1}) \geq e_i(C_{\sigma_2})$.*

Proofs are omitted due to lack of space.

Corollary 1. *If $\sigma_1, \sigma_2 \in \Sigma$ are strategy profiles such that for all $a_j \neq a_i$ we have that $\sigma_{1,j} = \sigma_{2,j}$, and that $\sigma_{1,i} = exert$ and $\sigma_{2,i} = shirk$, then for all j we have that $\sum_{T_w \in T_{win}} \Pr_{C_{\sigma_1}}(T_w) \geq \sum_{T_w \in T_{win}} \Pr_{C_{\sigma_2}}(T_w)$. If σ_1, σ_2 are strategy profiles such that $\sigma_1 >_e \sigma_2$ and such that $\sigma_{1,i} = \sigma_{2,i}$, then $\sum_{T_w \in T_{win}} \Pr_{C_{\sigma_1}}(T_w) \geq \sum_{T_w \in T_{win}} \Pr_{C_{\sigma_2}}(T_w)$.*

2. THE COMPLEXITY OF INCENTIVES

Given an effort game, agents naturally consider whether they should exert effort, and the principal naturally considers how it should incentivize a certain subset of the agents to exert effort, while minimizing the sum of rewards it must give. We now consider an effort game domain D, with agents $I = \{a_1, \dots, a_n\}$, where a_i is responsible for task t_i. The underlying coalitional game is G, with the value function $v : 2^T \to \{0, 1\}$. The set of success probabilities is $(\alpha_1, \beta_1), \dots, (\alpha_n, \beta_n)$, and the effort exertion costs are $c_1, \dots, c_n$. Several natural problems arise:

1. *DSE* — Dominant Strategy Exert: Given $G_e(r)$, is "exert" a dominant strategy for a_i?
2. *IEE* — Iterated Elimination Exert: Given $G_e(r)$, is "exert" the *only* remaining strategy for a_i after iterated elimination of dominated strategies? This means that if Σ' is the set of strategy profiles remaining after a sequence of iterated eliminations, then for any strategy profile $\sigma' = (\sigma'_1, \dots, \sigma'_n) \in \Sigma'$ we have $\sigma'_i = exert$.
3. *IE-INI* — Iterated Elimination Inducing Incentives: Given D, compute an iterated elimination of dominated strategies incentive inducing scheme $r = (r_1, \dots, r_n)$ for C (see Definition 6).

The computational complexity of these problems is investigated in the following sections.

2.1 Rewards and the Banzhaf Power Index

We now show the relation between the complexity of the above problems, and power indices in the underlying coalitional game.

Theorem 2. *Let D be an effort game domain, where for a_i we have $\alpha_i = 0$ and $\beta_i = 1$, and for all $a_j \neq a_i$ we have $\alpha_j = \beta_j = \frac{1}{2}$. Let $r = (r_1, \dots, r_n)$ be a reward vector, so that for a_i exerting effort is a dominant strategy in $G_e(r)$. Then, $r_i > \frac{c_i}{\beta_i(v)}$ (where $\beta_i(v)$ is the Banzhaf power index of t_i in the underlying coalitional game G, with the value function v).*

Consider an effort game domain D, the reward vector r, and the resulting effort game $G_e(r)$. We show that *minimal-DSE*, testing whether a certain reward vector is the minimal reward vector that makes exerting effort a dominant strategy for a certain agent a_i, is at least as hard computationally as calculating the Banzhaf power index in the underlying coalitional game G. Since calculating the Banzhaf index is #P-hard in various domains (see Section 3),[2] this shows that, in general, minimal-DSE is also #P-hard. The proof for the next theorem is based on showing that being able to answer *minimal-DSE* queries in polynomial time allows calculating the Banzhaf index in polynomial time (a reduction from Banzhaf to minimal-DSE).

Theorem 3. *Minimal-DSE is at least as hard computationally as calculating the Banzhaf power index of its underlying coalitional game G.*

One domain of coalitional games where calculating the Banzhaf power index is known to be NP-hard is weighted voting games [5]. Thus, in an effort game where the underlying coalitional game is a weighted voting game, it is NP-hard to test if exerting effort is a dominant strategy.

2.2 Inducing Incentives in Unanimity Weighted Voting Games

We now consider a restricted class of effort games in weighted voting domains, and show how to find a dominant strategy incentive inducing scheme, or an iterated elimination of dominated strategies incentive inducing scheme. In our domain, voters decide on a course of action using weighted voting. Each voter has a weight, and a decision passes if the total weight of agents that vote for it exceeds a certain threshold. We consider a restricted case where the voters may have different weights (voter i has weight w_i), but the quota is so high that the decision only passes when *all* agents vote for it. Such games are called *unanimity voting games*, and in such games the quota for passing the decision is $q = \sum_{i=1}^{n} w_i$. Thus, in this restricted setting, all voters have equal power (so the Banzhaf power index is the same for all the agents, even though they have different weights).

Suppose each voter has a probability of α to vote in favor of the decision. An agent a_i may increase the probability of voter v_i voting in favor of the decision to $\beta > \alpha$, at a certain cost of exerting this effort, c_i. In our domain, we will assume the effort exertion cost is proportional to the voter's weight, so $c_i = w_i$. Consider a principal that wants all the agents to exert effort. We model this situation as an effort game domain D. The underlying coalitional game G

[2]The complexity of calculating the power index depends on the representation of the game. Most of these hardness results assume a polynomial algorithm that returns the value of any coalition (or gives a concise representation which allows doing so), and show that in some domains, even with access to such an algorithm, calculating the power index is computationally hard.

has the tasks $T = (t_1, \ldots, t_n)$. A coalition of tasks $C \subseteq T$ wins in G if it contains all the tasks and loses otherwise, so $v(T) = 1$, and for all $C \neq T$ we have $v(C) = 0$. The success probability pairs are identical for all tasks: $(\alpha_1 = \alpha, \beta_1 = \beta), (\alpha_2 = \alpha, \beta_2 = \beta), \ldots, (\alpha_n = \alpha, \beta_n = \beta)$. Agent a_i is in charge of t_i, and the effort exertion costs are the weights in the underlying weighted voting game, so $c_i = w_i$.

Consider the above effort game domain D. Given a reward vector $r = (r_1, \ldots, r_n)$ we get the effort game $G_e(r)$. We show how to compute both a dominant strategy incentive inducing scheme (D-INI) and an iterated elimination of dominated strategies incentive inducing scheme (IE-INI) in this domain. We also show how to find such an IE-INI vector that minimizes $\sum_{i=1}^{n} r_i$. We first show how to calculate the minimal reward that makes exerting effort a dominant strategy for a_i.

Lemma 1. *If $r_i > \frac{c_i}{\alpha^{n-1} \cdot (\beta - \alpha)}$, then exerting effort is a dominant strategy for a_i.*

The above lemma allows us to solve D-INI in this domain in polynomial time—we have a simple formula for the minimal reward vector r which is a dominant strategy incentive inducing scheme.

Corollary 2. *D-INI is in P for the effort game in the above specific weighted voting domain. The following reward vector is a dominant strategy incentive inducing scheme: $r^* = (\frac{c_1}{\alpha^{n-1} \cdot (\beta - \alpha)}, \ldots, \frac{c_n}{\alpha^{n-1} \cdot (\beta - \alpha)})$.*

We now consider IE-INI, computing an iterated elimination of dominated strategies incentive inducing scheme in this domain. We show that such a scheme can significantly reduce the total rewards $\sum_{i=1}^{n} r_i$. We suggest an IE-INI procedure for this domain.

Due to Lemma 1, if $r_i > \frac{c_i}{\alpha^{n-1} \cdot (\beta - \alpha)}$, then exerting effort is a dominant strategy for a_i. Thus, after one step of elimination of dominated strategies, a_i is sure to exert effort. Consider a_j, who knows a_i would exert effort.

Lemma 2. *Let a_i, a_j be two agents,and r_i, r_j be their rewards so that $r_i > \frac{c_i}{\alpha^{n-1} \cdot (\beta - \alpha)}$ and that $r_j > \frac{c_j}{\alpha^{n-2} \cdot \beta \cdot (\beta - \alpha)}$. Then under iterated elimination of dominated strategies, the only remaining strategy for both a_i and a_j is to exert effort.*

We now consider an iterated elimination of dominated strategies incentive inducing scheme (IE-INI). First choose an ordering of the agents; then go through the agents in that order, and find the minimal reward required to make exerting effort a dominant strategy for each of them, given that its predecessors exert effort as well. Denote by π a permutation (reordering) of the agents. We denote the set of all such permutations Π. $\pi(i)$ is the location of a_i in the new ordering of the agents. In the generated reward scheme, the strategy "shirk" for $a_{\pi(i)}$ is eliminated during round i of strategy elimination.

Theorem 4. *IE-INI is in P for the effort game in the abovementioned specific weighted voting domain. For any reordering of the agents $\pi \in \Pi$, a reward vector r_π, where for all agents a_i we have: $r_{\pi(i)} > \frac{c_{\pi(i)}}{\alpha^{n-\pi(i)} \cdot \beta^{\pi(i)-1} \cdot (\beta - \alpha)}$, is an iterated elimination incentive inducing scheme.*

We note that each reordering π of the agents results in a different reward vector. The principal is interested in minimizing $\sum_{i=1}^{n} r_i$. We show that the best ordering is according to the agents' exertion costs $c_i = w_i$.

Lemma 3. *The reward vector r_π that minimizes $\sum_{i=1}^{n} r_i$ is achieved by sorting agents by their weights $w_i = c_i$, from smallest to biggest.*

We have thus shown that in this domain we can find an IE-INI by sorting the agents according to their weights (and thus costs of exerting efforts), and then using the equation from Theorem 4 to construct the reward vector.

3. RELATED WORK

The computational complexity of iterated elimination of dominated strategies has been studied in [6, 4]. The Banzhaf index originated in [3]. [7] showed that calculating the Banzhaf index in weighted voting games is NP-complete, and [2] showed that calculating it in *network flow games* is #P-complete.

A model similar to ours appeared in [8]. However, it defined a very *restricted* effort game, where only the grand coalition I wins, where α is the same for all agents, and where a task is always completed when the agent in charge exerts effort, so $\beta = 1$. For that domain, it shows an easily calculable reward vector which is an iterative elimination of dominated strategy implementation. [8] focused on the economics of discrimination, whereas this paper concentrates on the computational features of effort games. Another model similar to ours is given in [1]. However, [1] focused on a Nash Equilibrium, while this paper focuses on the stronger notion of a dominant strategy equilibrium and on iterated elimination of dominated strategies equilibrium.

4. ACKNOWLEDGMENT

This work was partially supported by Israel Science Foundation grant #898/05.

5. REFERENCES

[1] M. Babaioff, M. Feldman, and N. Nisan. Combinatorial agency. In *Proceedings of the 7th ACM Conference on Electronic Commerce (EC 2006)*, Ann Arbor, Michigan, 2006.

[2] Y. Bachrach and J. S. Rosenschein. Computing the Banzhaf power index in network flow games. In *The Sixth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2007)*, pages 323–329, Honolulu, May 2007.

[3] J. F. Banzhaf. Weighted voting doesn't work: a mathematical analysis. *Rutgers Law Review*, 19:317–343, 1965.

[4] V. Conitzer and T. Sandholm. Complexity of (iterated) dominance. In *Proceedings of the 6th ACM Conference on Electronic Commerce (EC 2005)*, pages 88–97, Vancouver, 2005.

[5] X. Deng and C. H. Papadimitriou. On the complexity of cooperative solution concepts. *Math. Oper. Res.*, 19(2):257–266, 1994.

[6] I. Gilboa, E. Kalai, and E. Zemel. The complexity of eliminating dominated strategies. *MOR: Mathematics of Operations Research*, 18, 1993.

[7] Y. Matsui and T. Matsui. NP-completeness for calculating power indices of weighted majority games. *Theoretical Computer Science*, 263(1–2):305–310, 2001.

[8] E. Winter. Incentives and discrimination. *American Economic Review*, 94(3):764–773, 2004.