Math. Log. Quart. **55**, No. 4, 1 – 21 (2009) / **DOI** 10.1002/malq.200910018

Effort Games and the Price of Myopia

Yoram Bachrach[∗1], **Michael Zuckerman**[∗∗1], and **Jeffrey S. Rosenschein**[∗∗∗1]

[1] School of Engineering and Computer Science, Hebrew University, Jerusalem, Israel

Received 28 September 2008, revised 15 February 2009, accepted 17 February 2009
Published online 20 July 2009

Key words Coalitional games, principal agent, effort games.
MSC (2000) 03D15, 69Q17, 91A12

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; carrying out certain subsets of the tasks completes the project successfully, while carrying out other subsets does not. The probability of carrying out 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 carrying out 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 suggest the *Price of Myopia* (PoM) as a measure of the influence the model of rationality has on the minimal payments the principal has to make in order to motivate the agents in such a domain to exert effort.

We consider the computational complexity of testing whether exerting effort is a dominant strategy for an agent, and of finding a reward strategy for this domain, using either a *dominant strategy equilibrium* or using *iterated elimination of dominated strategies*. We show these problems are generally #P-hard, and that they are at least as computationally hard as calculating the Banzhaf power index in the underlying coalitional game. We also show that in a certain restricted domain, where the underlying coalitional game is a weighted voting game with certain properties, it is possible to solve all of the above problems in polynomial time. We give bounds on PoM in weighted voting effort games, and provide simulation results regarding PoM in another restricted class of effort games, namely effort games played over Series-Parallel Graphs.

1 Introduction

The computational aspects of many game theoretic concepts have been thoroughly studied in recent years, as they are significant for many real-world domains, including auctions, voting, and electronic commerce. A key issue in many such domains is constructing a proper reward scheme to achieve the desired behavior of self-interested agents.

There are many such examples: auction theory is sometimes used to design auctions that maximize the revenue of the auctioneer; social choice theory attempts to design mechanisms that give agents incentives to truthfully reveal their preferences so that an optimal social choice is made; solution concepts in coalitional game theory are used to make sure that rewards are distributed in a fair manner, or so that the coalitions that are formed are stable. The computational aspects of such concepts are critical, since in order to use them in practice, one must find a tractable way to compute them.

1.1 Our Setting

In this paper, we deal with the computational complexity of finding a reward scheme in *effort games* in open environments. These games are a particular model of a principal-agent problem. In our model, the mechanism's

∗ Corresponding author: e-mail: yori@cs.huji.ac.il
∗∗ e-mail: michez@cs.huji.ac.il
∗∗∗ e-mail: jeff@cs.huji.ac.il

purpose is to incentivize agents to exert effort when working on a common project. Completing the project requires the performance of various tasks. Some of the subsets of the tasks are “winning” subsets – when these tasks are completed, the project succeeds; some subsets are “losing” subsets – when only these tasks are completed, the project fails.

A self-interested agent is in charge of each task. This agent can exert effort, which increases the probability that its task will be completed successfully. However, this exertion of effort has a certain cost for the agent. On the other hand, the task has a certain probability of being completed successfully even if the agent in charge of it does not exert any effort. A natural way to reward agents is based on the effort they have exerted. However, in open environments this information is sometimes not available to the mechanism. These environments are open in the sense that a mechanism can only get information about the success of an entire project, and cannot get information about the performance of agents in the various tasks required for this project, or whether they have exerted effort or not. Thus, the mechanism can only reward the agents based on the success of the entire project. We consider the computational complexity of finding a reward scheme, so that all agents have incentives to exert effort and so that the project is successful.

One example of such a situation is maintenance efforts. Consider, for example, a communication network with a source node and a target node, where agents are in charge of maintenance tasks for the links between the nodes of the network. If no maintenance effort is expended on a link, it has some probability $\alpha < 1$ of functioning, but if a maintenance effort is made for the link, it functions with a higher probability $\beta > \alpha$. Consider a mechanism in charge of sending some information between source and target. This mechanism only knows whether it succeeded in sending the information, but if that attempt fails, may not know which links failed or if they failed due to lack of maintenance. Examining the links to see which of them failed may be very costly, and even if the mechanism could know which links worked and which failed, it cannot tell for sure whether a link failed because it was poorly maintained.

Another example is voting domains, which we consider in Section 5. Consider a domain where a decision is made by a certain voting protocol, where only the result is published, and not the votes of the participants. Suppose that some outsider has a certain desired outcome x, and attempts to use several lobbying agents, each able to convince a certain voter to vote for the desired outcome. When such a lobbying agent exerts effort, her voter votes for x with a high probability β, but this effort has a certain cost for that agent. If that agent shirks (does not exert effort), her voter only has a probability of $\alpha < \beta$ to vote for x. The outsider wants to make sure some of the agents exert effort. However, she cannot tell which agent has exerted effort, or even what each voter voted for. She can only know whether x was chosen or not.

In the above examples, the interested party, called *the principal*, cannot observe the agents’ decisions about whether to expend effort. The only information available to it is the overall result of the project. However, the agents have a certain cost to exerting efforts, and require a certain reward as an incentive for making such an effort. The principal cannot offer any reward based on whether an agent has exerted effort or not, since it does not have that information. It can, however, offer the agents a certain reward conditional on the project’s being successful. Since agents have a certain capability of increasing the probability of obtaining this outcome, they may exert effort in order to get their reward. However, they would only do so if the expected reward from exerting effort is greater than the cost of that effort for them.

The principal thus offers a reward vector $r = (r_1, \ldots, r_n)$, where for all i, $r_i \geq 0$. If the project succeeds, the reward of agent a_i is r_i, and if the project fails then for all a_i the reward of a_i is 0. Consider a principal that wants to motivate a certain subset of the agents to exert effort.[1)] Of course, it can offer each agent in this subset a reward that is high enough to make sure it is worthwhile for them to exert effort no matter what the other agents do. However, we assume that the principal wants to minimize the total rewards given when the project succeeds, $\sum_{i=1}^{n} r_i$.

In this paper, we examine several domains that can be modeled as effort games. We begin with a simple voting domain where voters vote on a certain decision, under the weighted voting model. Each voter has a certain weight, and a decision is carried out if the sum of weights of the voters who vote for accepting the decision exceeds a certain threshold. In this domain, each agent is capable of affecting the vote of a certain voter, and may increase the probability that that voter would choose to accept the decision. However, affecting the voters

[1)] In some domains considered in this paper, the mechanism wants to motivate *all* the agents to exert effort. It is also possible that the mechanism would *choose* which agent subset to incentivize, based on a target function for which it wants to optimize.

is costly for the agents. The principal wishes to maximize the probability of accepting the decision. However, the principal cannot observe whether a certain agent has exerted its influence, or even what a certain voter has voted for. The principal can only observe the outcome of the entire voting procedure, whether the decision has been accepted or not. The principal can incentivize agents to exert their influence by promising each such agent a certain reward if the decision is accepted.

The voting effort game is described in Section 5. We examine several reward schemes for this domain, each with different assumptions regarding the rational behavior of the agents. We show that when we assume a simple model of rational behavior for the agents, motivating them to exert effort is quite costly for the principal. Surprisingly, when we assume a different model of rationality where the agents make more sophisticated decisions, it turns out to be cheaper for the principal to motivate them, and the total rewards they obtain are smaller. We define the Price of Myopia (PoM), which quantifies the ratio between the total costs for the principal under these two models of rationality.

Besides examining the weighted voting domain, we examine a particular extension of it, namely effort games over series parallel graphs (SPGs). SPGs allow us to represent more complex effort games. Voting effort games may be viewed as just one type of a connection, a series connection, in SPG effort games. SPG effort games also allow parallel connections, reflecting the fact that one agent can substitute for another agent. The SPG effort game model is discussed in Section 7, where we give some results of simulations regarding the costs of the principal in certain SPG effort games.

The most general form of effort games are general effort games, where we express the domain as a coalitional game. We define the general effort game model in Section 3.2, and analyze the computational complexity of finding out whether a certain reward vector makes it the dominant strategy of a certain agent to exert effort. We also analyze the computational complexity of constructing a reward vector that guarantees that a certain subset of agents exert effort.[2)]

2 Related Work

In this paper we consider *effort games*, which model domains where agents are working in teams, but may not exert effort when their decisions are unobservable. These domains have a moral hazard problem (where free-riding may cause the teams to function in a sub-optimal way), and a central authority in charge of a project may decide to reward agents depending only on whether the project has ended successfully or not. Such considerations are also discussed in [8].

A model very similar to ours appears in [15]. That paper considers a simple situation, where a set of identical agents are working on a common project. Each agent may exert effort or not. Similarly to our model, each agent is in charge of a certain task, which has a probability of $\beta = 1$ of succeeding when effort is exerted, and some probability α of succeeding even if the agent shirked. However, that paper defines a very *restricted* effort game, where in the underlying coalitional game the only winning coalition is the grand coalition I, where α is the same for all agents, and where a task is always completed when the agent in charge exerts effort, so $\beta = 1$. [15] considers a principal who attempts to incentivize all the agents to exert effort, and shows a certain, easily calculable, reward vector which is an iterative elimination of dominated strategies implementation. Since agents in that domain are identical, yet the rewards are different, it shows that a certain discrimination allows the principal to minimize her payments while still having the project succeed with a probability of 1. Our model of effort games extends that model, by allowing different subsets of the tasks to complete the project. The focus of [15] is on the economic question of discrimination in such settings, whereas our paper concentrates on the computational features of effort games, and their relation to power indices.

Effort games can be seen as a restricted case of the model defined in [1]. That work also considers a domain where a principal employs agents in a joint project. The common theme of this paper and [1] is that the actions taken by the agents affect the probability that the project is successful. However, the model of [1] is more general – it allows agents to take more than one action, and each strategy profile results in a certain distribution over the

[2)] The complexity results obviously depend on the representation of the function that decides which agent subsets are winning and which are losing (sometimes called the characteristic function, or coalitional function). The weighted voting domains and SPG games described here have very concise representations (for more details we refer the reader to the papers mentioned in the related work discussion, in Section 2). However, the hardness results given later show that certain problems in effort games are harder than computing the Banzhaf power index, and thus hold for *any* representation where it is hard to compute the Banzhaf power index.

possible outcomes for the joint project. Since we consider a restricted form, hardness results regarding the general domain may not apply to our domain, and indeed we are able to provide polynomially computable formulas for some restricted domains. One major difference between our model and the model of [1] is the choice of a solution concept. [1] focuses on a Nash Equilibrium, while the current paper focuses on the stronger notion of a dominant strategy equilibrium and on iterated elimination of dominated strategies equilibrium. Thus, our probabilistic model can be represented within "combinatorial agency", but we have different assumptions about the rational behavior of the agents, which are reflected in the solution concept used. These different solution concepts change the complexity of inducing incentives, so our results are very different from those in that stream of related work.

Another paper similar to ours in spirit is [11]. It considers an interested party who wishes to influence the behavior of agents in a game, without directly enforcing agent behavior. We also consider an interested party that commits to making a positive monetary transfer. However, the approach in [11] is very different from ours. It shows how an interested party can affect the agents' behavior by committing to a monetary transfer *when the agents behave in a certain desired way*. This means the reward is contingent on the strategies the agents have used. This promise of a monetary transfer can indeed affect the agents' behavior. However, in our paper we do not assume the interested party is capable of *directly monitoring* the agents' behavior, so the monetary transfer can only be based on the *final outcome* of the game, rather than the strategy profile the agents employed. Thus, our setting is a *principal-agent* setting. We also do not assume the outcome of the game is directly determined by the agents' behavior. Rather, the game has several possible outcomes, and the *probability* that each such outcome would occur depends on the strategy profile.

The effort game model defined in this paper relies on both cooperative and non-cooperative game theoretic models. Dominant strategy equilibria, Nash equilibria, and iterated elimination of strictly dominated strategies are all known solution concepts. Our model and the concept of the Price of Myopia (PoM) focus on iterated elimination of dominated strategies. The computational complexity of iterated elimination of dominated strategies has been less studied than the complexity of Nash equilibria. [7] studies the complexity of iteratively eliminating solutions. [9] examines elimination of dominated strategies in the restricted case of binmatrix games. [4] shows how to compute non-iterated dominance in polynomial time (through linear programming), but provides hardness results for testing whether there exists a path of iterated eliminations that eventually eliminates a given strategy. [4] also considers testing strategy dominance relations in Bayesian games.

In this paper we show the relation between schemes that induce agents to exert effort in various cooperative domains and power indices. The concept of power indices originated from work on cooperative game theory. One prominent solution concept for coalitional games is that of the Shapley value, due to a seminal paper by Shapley [13]. The Shapley-Shubik index [14], an application of the Shapley value [13], has been used to measure power in committees and politics. The Banzhaf power index originated in [3]. Several papers have dealt with the computational complexity of calculating power indices. [5] shows that computing the Shapley value in weighted majority games is #P-complete. [10] has shown that calculating the Banzhaf index in weighted voting games is NP-complete. [2] has considered the problem of calculating the Banzhaf power index in *network flow games*. In this coalitional game, which models a certain problem in network reliability, agents control links in a network flow graph, and a coalition wins if it manages to allow a certain flow between a source vertex and a target vertex. It was shown for that domain that calculating the Banzhaf index is #P-complete, but gave a polynomial algorithm for a certain restricted case.

A certain restricted case of effort games that we consider in this paper is Series-Parallel Graphs (SPGs). In [6] the authors apply their results on marginal contribution nets to derive an algorithm for computing the Shapley value and the Banzhaf index in SPGs. Their algorithm works on an arbitrary flow value in SPGs and is exponential in the flow value. However, when applied to connectivity games on SPGs (like the ones defined in Section 7), the algorithm is polynomial, since the flow value there is 1.

3 The Formal Model

3.1 Preliminaries

We now define several game theoretic notions required for defining effort games. The definition of effort games relies on the definition of both normal form games and simple coalitional games.

Definition 3.1 An *n-player normal form game* is given by a set of agents (players) $I = \{a_1, \dots, a_n\}$, and for each agent a_i a (finite) set of pure strategies S_i, and a utility (payoff) function $F_i : S_1 \times S_2 \times \dots \times S_n \longrightarrow \mathbb{R}$. $F_i(s_1, \dots, s_n)$ denotes a_i's utility when each player a_j plays strategy s_j.

For brevity, we denote the set of strategy profiles $\Sigma = S_1 \times S_2 \times \dots \times S_n$, and denote items in Σ as $\sigma \in \Sigma$ $(\sigma = (s_1, \dots, s_n)$, where $s_i \in S_i)$.

We also denote $\Sigma_{-i} = S_1 \times \dots \times S_{i-1} \times S_{i+1} \times \dots \times S_n$, and given an incomplete strategy profile $\sigma_{-i} = (s_1, \dots, s_{i-1}, s_{i+1}, \dots, s_n) \in \Sigma_{-i}$ we denote $(\sigma_{-i}, s_i) = (s_1, \dots, s_{i-1}, s_i, s_{i+1}, \dots s_n) \in \Sigma$.

Given a normal form game G, we say agent a_i's strategy $s_{\mathrm{x}} \in S_i$ strongly dominates $s_{\mathrm{y}} \in S_i$ if a_i's utility is higher when using s_{x} than when using s_{y}, no matter what strategies the other agents use.

Definition 3.2 Given a normal form game G, we say agent a_i's strategy $s_{\mathrm{x}} \in S_i$ *strictly dominates* $s_{\mathrm{y}} \in S_i$ if the following holds: For any incomplete strategy profile $\sigma_{-i} \in \Sigma_{-i}$ we have $F_i((\sigma_{-i}, s_{\mathrm{x}})) > F_i((\sigma_{-i}, s_{\mathrm{y}}))$.

Definition 3.3 Given a normal form game G, we say agent a_i's strategy s_{x} is a_i's *dominant strategy* if it dominates all other strategies $s_i \in S_i$.

Given a certain game, game theoretic solution concepts specify which outcomes are reasonable, under various assumptions of rationality and common knowledge. In this paper, we will deal with two solution concepts: dominant strategy equilibrium, and iterated elimination of dominated strategies. Given a certain game, an agent a_i may or may not have a dominant strategy. One known game theoretic solution concept is a dominant strategy equilibrium. This concept defines a *single* reasonable strategy profile (and derived payoffs) for certain games.

Definition 3.4 Given a normal form game G, we say a strategy profile $\sigma = (s_1, \dots, s_n) \in \Sigma$ is a *dominant strategy equilibrium* if for any agent a_i, strategy s_i is a dominant strategy for a_i.

Another solution concept is that of *iterated elimination of dominated strategies*. In iterated dominance, strictly dominated strategies are removed from the game, and no longer have any effect on future dominance relations. A certain strategy $s_{\mathrm{x}} \in S_i$ may not dominate $s_{\mathrm{y}} \in S_i$, because s_{y} performs better against $\sigma_a \in \Sigma_{-i}$. However, if one of the strategies in σ_a is removed, that profile is no longer applicable, so s_{x} may now dominate s_{y}. It is a well-known fact that if we remove dominated strategies until no more dominated strategies remain, in the end the remaining strategies for each player will be the same, regardless of the order in which strategies are removed [12]. Some authors refer to this as the fact that iterated (strict) dominance is path-independent.

The heart of a general effort game is a boolean function that decides which task subsets complete the common project successfully, and which task subsets do not. Our results depend on viewing this boolean function as a simple coalitional game. We now define a simple coalitional game.[3)]

Definition 3.5 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 $v : 2^T \longrightarrow \mathbb{R}$, 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 ($v : 2^T \longrightarrow \{0, 1\}$). We say a subset $C \subseteq T$ *wins* if $v(C) = 1$, and say it *loses* if $v(C) = 0$. We denote the set of all subsets of tasks that win the simple game as $T_{\mathrm{win}} = \{T' \subseteq T | v(T') = 1\}$. 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 $C' \subseteq C \subseteq T$ 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 $C \subseteq T$ wins, every superset of C also wins.

We now define the weighted voting game, which is a famous game-theoretic model of cooperation in political bodies. In this game, 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 3.6 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 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

[3)] Rather than defining the function over agents, as typically done in definitions for coalitional games, we call the players in this game tasks, to keep our terminology consistent with the rest of the paper.

is power indices. A possible definition of the power of an agent is its *a priori* probability of having a significant impact on the outcome of the game. Different definitions of "significant impact" have resulted in the definition of different power indices, one of which is the Banzhaf index [3].

Definition 3.7 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 $\beta(v) = (\beta_1(v), \dots, \beta_n(v))$ where

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

The Banzhaf power index reflects the assumption that the agents are independent in their choices. Another prominent power index, the Shapley-Shubik power index, does not reflect such an assumption. This property of the Banzhaf index is similar to some of the assumptions of effort games.

3.2 The Effort Game Model

We now define effort game models and related problems.

Definition 3.8 An *effort game domain* is a domain that consists of the following: A set $I = \{a_1, \dots, a_n\}$ of n agents; a set of n tasks, $T = \{t_1, \dots, t_n\}$; a simple coalitional game G with task set T, such that $|T| = n$, and with the value function $v : 2^T \longrightarrow \{0, 1\}$; a set of success probability pairs $(\alpha_1, \beta_1), (\alpha_2, \beta_2), \dots, (\alpha_n, \beta_n)$ so that $\alpha_i, \beta_i \in \mathbb{R}$, and that $0 \leq \alpha_i < \beta_i \leq 1$; a set of effort exertion costs $c_1, \dots, c_n \in \mathbb{R}$, 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 coalitional 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 to be completed. However, exerting effort costs that agent a certain utility c_i. If the agent shirks (does not exert effort), then 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 $C \subseteq I$, 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 $T' \subseteq T$ 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 that 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:

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

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

$$\mathrm{Pr_C(Win)} = \sum_{T_{\mathrm{w}} \in T_{\mathrm{win}}} \mathrm{Pr_C}(T_{\mathrm{w}}).$$

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 exert effort, and needs to reward the agents so they will exert effort despite their cost of doing so. She thus designs a *reward scheme*, but attempts to minimize her costs.

Let $C \subseteq I$ 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 the 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 it 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 $r = (r_1, \dots, r_n)$.

Given the reward vector $r = (r_1, \dots, r_n)$, and given that the agents that exert effort are $C \subseteq I$, a_i's expected reward is $e_i(C) = \sum_{T_{\mathrm{w}} \in T_{\mathrm{win}}} \Pr_C \mathrm{C}(T_{\mathrm{w}}) \cdot r_i$. 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 she does not incur the cost c_i, but her 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 3.9 An *Effort Game* is the normal form game $G_{\mathrm{e}}(r)$ defined on the above domain with a simple coalitional game G and a reward vector $r = (r_1, \dots, r_n)$, as follows.

In $G_{\mathrm{e}}(r)$ agent a_i has two strategies: $S_i = \{\mathrm{exert}, \mathrm{shirk}\}$. Denote by Σ the set of all strategy profiles $\Sigma = S_1 \times \cdots \times S_n$. Given a strategy profile $\sigma = (s_1, \dots s_n) \in \Sigma$, we denote the coalition of agents that exert effort in σ by $C_\sigma = \{a_i \in I | s_i = \mathrm{exert}\}$. 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 $r = (r_1, \dots, r_n)$: the payoff of each agent in strategy profile σ is her expected reward minus the cost of the effort exerted. Thus:

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

G_{e} depends on r, so we denote it $G_{\mathrm{e}}(r)$.

Given a simple coalitional game G, each reward vector r defines a different effort game $G_{\mathrm{e}}(r)$. Given an effort game, 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'.[4)]

Definition 3.10 An *Incentive-Inducing Scheme for a Coalition C' in the Effort Game G_{e}* is a reward vector $r = (r_1, \dots, r_n)$, such that in the effort game $G_{\mathrm{e}}(r)$, in *any* strategy profile $\sigma \in \Sigma$ allowable by the solution concept, the agents in C' exert effort, so $C' \subseteq C_\sigma$.

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

Given an effort game domain G_{e} and a coalition of agents $C' \subseteq I$ that the principal wants to exert effort, we now define two types of incentive-inducing schemes: a dominant strategy scheme, and an iterated elimination of dominated strategies scheme.

Definition 3.11 A *Dominant Strategy Incentive-Inducing Scheme for C'* is a reward vector $r = (r_1, \dots, r_n)$ such that for any $a_i \in C'$ exerting effort is a dominant strategy for a_i.

Definition 3.12 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 $G_{\mathrm{e}}(r)$, after any sequence of eliminating dominated strategies, for any $a_i \in C'$, the only remaining strategy for a_i is to exert effort.

3.2.1 Strategy Changes and Expected Rewards

We now consider the effects of strategy changes and the expected rewards. We only specify some notation we need later in the paper, and state some theorems we later use. The full proofs are found in Appendix 9, and are given for completeness.[5)]

---

4) Several papers regarding "combinatorial agency" [1] have focused on a Nash equilibrium domain. We survey some of this work in Section 2. In this paper, we focus on a dominant strategy implementation, and on an iterated elimination of dominated strategies implementation.

5) These theorems or ones very similar to the ones presented here are given in [1], and are only given here to help the reader fully understand our other results.

We first define the following relation regarding effort exertion in strategy profiles. Let D be an effort game domain, and $r = (r_1, \ldots, r_n)$ be a reward vector. Let $\sigma_1, \sigma_2 \in \Sigma$ be two strategy profiles in $G_{\mathrm{e}}(r)$, so $\sigma_1 = (s_{1,1}, s_{1,2}, \ldots, s_{1,n})$ and $\sigma_2 = (s_{2,1}, s_{2,2}, \ldots, s_{2,n})$. We say σ_1 is *more exerting* than σ_2, and denote $\sigma_1 >_{\mathrm{e}} \sigma_2$ 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, so for all a_i we have that $s_{2,i} = \mathrm{exert}$ implies $s_{1,i} = \mathrm{exert}$, and for some a_j we have $s_{2,j} = \mathrm{shirk}$ and $s_{1,j} = \mathrm{exert}$.

For some of our results, we use the fact that the expected reward of a given agent increases when more of the other agents exert effort (no matter whether that agent exerts effort himself or not). We provide several theorems regarding this. First, if a *single* agent that shirked decides to exert effort, the expected rewards of all agents increase.

Theorem 3.13 *Let D be an effort game domain, $r = (r_1, \ldots, r_n)$ be a reward vector, and a_i be a certain agent in that domain. Let $\sigma_1, \sigma_2 \in \Sigma$ be two strategy profiles in $G_{\mathrm{e}}(r)$ so that for all $a_j \neq a_i$ we have that $\sigma_{1,j} = \sigma_{2,j}$, and that $\sigma_{1,i} = \mathrm{exert}$ and $\sigma_{2,i} = \mathrm{shirk}$. Then for all j we have that $e_j(C_{\sigma_1}) > e_j(C_{\sigma_2})$.*

The expected reward of any agent increases when more of the other agents exert effort.

Theorem 3.14 *Let D be an effort game domain, $r = (r_1, \ldots, r_n)$ 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 $G_{\mathrm{e}}(r)$ so that $\sigma_1 >_{\mathrm{e}} \sigma_2$, and so that $\sigma_{1,i} = \sigma_{2,i}$. Then $e_i(C_{\sigma_1}) > e_i(C_{\sigma_2})$.*

The above Theorems 3.13 and 3.14 have considered the expected rewards given a certain strategy profile, but can also be stated with regard to the probability of having a winning task subset.

Corollary 3.15 *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 $\sigma_{1,i} = \mathrm{exert}$ and $\sigma_{2,i} = \mathrm{shirk}$, then for all j we have*

$$\sum_{T_{\mathrm{w}} \in T_{\mathrm{win}}} \Pr_{C_{\sigma_1}}(T_{\mathrm{w}}) > \sum_{T_{\mathrm{w}} \in T_{\mathrm{win}}} \Pr_{C_{\sigma_2}}(T_{\mathrm{w}}) \ .$$

If σ_1, σ_2 are strategy profiles such that $\sigma_1 >_{\mathrm{e}} \sigma_2$ and such that $\sigma_{1,i} = \sigma_{2,i}$, then

$$\sum_{T_{\mathrm{w}} \in T_{\mathrm{win}}} \Pr_{C_{\sigma_1}}(T_{\mathrm{w}}) > \sum_{T_{\mathrm{w}} \in T_{\mathrm{win}}} \Pr_{C_{\sigma_2}}(T_{\mathrm{w}}) \ .$$

Proof. We simply consider Theorems 3.13 and 3.14 for the case where $r_i = 1$. □

4 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. Different assumptions the principal has about the rational behavior of the agents are reflected in the solution concept it uses. We now formally frame the above problems. In the rest of this section, we consider an effort game domain D, with agents $I = \{a_1, \ldots, a_n\}$, where each a_i is responsible for task $t_i \in T$. The underlying coalitional game is G, with the value function $v : 2^T \longrightarrow \{0, 1\}$. The set of success probabilities is $(\alpha_1, \beta_1), \ldots, (\alpha_n, \beta_n)$, and the effort exertion costs are $c_1, \ldots, c_n$.

The following problems concern a reward vector $r = (r_1, \ldots, r_n)$, the effort game $G_{\mathrm{e}}(r)$, and a target agent a_i.

Definition 4.1 DSE (DOMINANT STRATEGY EXERT): Given $G_{\mathrm{e}}(r)$, is "exert" a dominant strategy for a_i?

Definition 4.2 IEE (ITERATED ELIMINATION EXERT): Given $G_{\mathrm{e}}(r)$, 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 elimination, then for any strategy profile $\sigma' = (\sigma'_1, \ldots, \sigma'_n) \in \Sigma'$ we have $\sigma'_i = \mathrm{exert}$.

The following problems concern the effort game domain D, and a coalition C.

Definition 4.3 D-INI (DOMINANT INDUCING INCENTIVES): Given D, compute a dominant strategy incentive-inducing scheme $r = (r_1, \ldots, r_n)$ for C (see Definition 3.11).

Definition 4.4 IE-INI (ITERATED ELIMINATION INDUCING INCENTIVES): Given D, compute an iterated elimination of dominated strategies incentive-inducing scheme $r = (r_1, \ldots, r_n)$ for C (see Definition 3.12).

We are typically interested in incentive-inducing schemes $r = (r_1, \ldots, r_n)$ that *minimize* the sum of payments, $r = \sum_{i=1}^{n} r_i$. In many cases, it is always possible to find an incentive-inducing scheme given that the sum of payments used, r, is high enough, and the focus is on minimizing this sum of payments. We denote the problems of finding incentive-inducing schemes that *minimize* the sum of payments r as MD-INI and MIE-INI (for a dominant strategy implementation and an iterated elimination implementation, respectively). The computational complexity of these problems in various domains is investigated in the following sections.

5 Inducing Incentives in Unanimous Weighted Voting Games

We first consider a restricted class of effort games, called unanimous weighted voting domains (or unanimous domains for short), and show how to find a dominant strategy incentive-inducing scheme, or an iterated elimination of dominated strategies incentive-inducing scheme in these domains. The domain highlights the relation between power in the underlying coalitional game, in this case the weighted voting game, and the incentives for the resulting effort game domain. Unanimous domains are a restricted case of weighted voting domains. In these domains, in order to form a successful coalition, *all* the agents must reach a consensus about a course of action. Unanimous domains can model joint decision making, where the agents must *unanimously* decide on a course of action. Such domains can also model situation that have nothing to do with voting, when a certain project requires a set of tasks, where *all* the tasks must be completed in order to succeed in the entire project.

Consider a joint project, where each agent a_i is in charge of a task, and where *all* the tasks must be completed in order for the project to succeed. A task is completed successfully with probability α, and agent a_i may increase the probability of succeeding in the task she is in charge of to β, at a certain cost c_i of exerting effort. Formally unanimous effort games are defined as follows. The underlying coalitional game G 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 has an effort exertion cost of c_i.

Weighted voting games where a decision is passed only if all agents *unanimously* vote for it offer another motivation for the above definitions. In general voting game domains, voters decide on a course of action using weighted voting. Each voter has a weight, and a decision passes if the total weight of the agents that vote for it exceeds a certain threshold. We consider *unanimous* domains, where although the voters may have different weights (voter i has weight w_i), the quota is so high that in fact the decision can only pass when *all* of the agents vote for it: we assume that the quota for passing the decision is $q = \sum_{i=1}^{n} w_i$. Thus, in this restricted setting, all the voters have equal power (so the Banzhaf power index is the same for all the agents, even though they have different weights). We use effort games to model lobbying efforts, where an interested party attempts to make sure the voters do indeed reach a unanimous decision about the desired course of action, and employs lobbying agents, who attempt to influence the voters. Suppose each voter has a probability of α to vote in favor of the decision. A lobbying 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 effort, c_i. In our domain, we will assume the effort exertion cost is proportional to the voter's weight, so $c_i = w_i$.[6)]

Consider a principal, that wants to maximize the probability of the project completing successfully (or, in the case of a weighted voting domain, making sure the decision is passed). The principal wants all the agents to exert effort, to maximize the probability of all the tasks completing successfully (or to make sure the voters would have a high probability of accepting the decision). Observe the above effort game domain D. Given a reward vector $r = (r_1, \ldots, r_n)$ we get the effort game $G_{\mathrm{e}}(r)$. 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 r_i that makes exerting effort a dominant strategy for a_i in the above domain.

6) Thus, in unanimous weighted voting domains, the weights are simply another name for the effort exertion costs.

Lemma 5.1 *If* $r_i \geq \dfrac{c_i}{\alpha^{n-1} \cdot (\beta - \alpha)}$*, then exerting effort is a dominant strategy for* a_i*, and this reward is the minimal reward that makes exerting effort a dominant strategy for* a_i*.*

Proof. As seen in Theorem 7.3, the condition for exerting effort being a dominating strategy for a_i is that for every $\sigma^a \in \Sigma_{-i}$,

$$r_i \geq \frac{c_i}{\sum_{T_{\mathrm{w}} \in T_{\mathrm{win}}} (\Pr_{C_{\sigma^a_e}}(T_{\mathrm{w}}) - \Pr_{C_{\sigma^a_s}}(T_{\mathrm{w}}))} \ .$$

However, in this domain, the only winning task subset is T, so we can restate this as

$$r_i \geq \frac{c_i}{(\Pr_{C_{\sigma^a_e}}(T) - \Pr_{C_{\sigma^a_s}}(T))} \ .$$

Since the only difference between σ^a_e and σ^a_s is that a_i exerts effort in the first and shirks in the second, we have

$$\Pr_{C_{\sigma^a_e}}(T) = P_{\sigma^a(\mathrm{T})} \cdot \beta \quad \text{and} \quad \Pr_{C_{\sigma^a_s}}(T) = P_{\sigma^a(\mathrm{T})} \cdot \alpha \ .$$

The notation $P_{\sigma^a(\mathrm{T})}$ denotes the probability of completing T given a *partial* strategy profile $\sigma^a \in \Sigma_{-i}$ (similar to the notation used in Theorem 3.13). Thus, an equivalent condition for having exerting effort be the dominant strategy for a_i is that for every $\sigma^a \in \Sigma_{-i}$ we have

$$r_i \geq \frac{c_i}{\Pr_{C_{\sigma^a}}(T) \cdot (\beta - \alpha)} \ .$$

Each $\sigma^a \in \Sigma_{-i}$ thus requires a different minimal reward r_i, and the smaller $\Pr_{C_{\sigma^a}}(T)$, the higher r_i must be. In our domain, $\Pr_{C_{\sigma^a}}(T)$ is smallest when all the agents shirk in σ^a, so $C_{\sigma^a} = \emptyset$ and $\Pr_{C_{\sigma^a}}(T) = \alpha^{n-1}$. □

The above lemma allows us to solve MD-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 5.2 MD-INI *is in* P *for the effort game in the above unanimous weighted voting domain. The reward vector* $r^* = (\dfrac{c_1}{\alpha^{n-1} \cdot (\beta - \alpha)}, \dots, \dfrac{c_n}{\alpha^{n-1} \cdot (\beta - \alpha)})$ *is a dominant strategy incentive-inducing scheme.*

Proof. An MD-INI solution is a reward vector that makes exerting effort the dominant strategy for each of the agents. Due to Lemma 5.1, r^* is indeed such a vector. We also note that if $r_i < \dfrac{c_i}{\alpha^{n-1} \cdot (\beta - \alpha)}$, shirking is the better strategy for a_i under the strategy profile where all the other agents shirk, so the above r^* is the dominant strategy incentive-inducing scheme which is *minimal* in terms of $\sum_{i=1}^n r_i$. This direct formula can be calculated in polynomial time. □

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 5.1, if $r_i \geq \dfrac{c_i}{\alpha^{n-1} \cdot (\beta - \alpha)}$, 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 in the game. Consider an agent a_j, who knows a_i would exert effort. We denote the set of strategy profiles which miss both the strategies for a_i and a_j as $\Sigma_{-\{i,j\}} = \Sigma_1 \times \dots \times S_{i-1} \times S_{i+1} \times \dots \times S_{j-1} \times S_{j+1} \times S_n$. Similarly to the notation in Theorem 3.13, we denote the probability of completing $T_{\mathrm{x}} \subseteq T$ given a *partial* strategy profile $\sigma^b \in \Sigma_{-\{i,j\}} =$ (missing the strategies of both a_i and a_j) as

$$\Pr_{C_{\sigma^b}}(T_{\mathrm{x}}) = \prod_{t_k \in (T_{\mathrm{x}} \setminus \{t_i, t_j\})} p_k(C) \cdot \prod_{t_k \in (T \setminus T_{\mathrm{x}} \setminus \{t_i, t_j\})} (1 - p_k(C)) \ .$$

We note that this probability ignores the question of whether t_i or t_j were completed (even though $t_i, t_j \in T_{\mathrm{x}}$) and only takes into consideration the tasks in $T_{\mathrm{x}} \setminus \{t_i, t_j\}$. The following lemma shows that the fact that one agent is sure to exert effort makes it cheaper to make sure another agent exerts effort.

Lemma 5.3 *Let a_i, a_j be two agents, and r_i, r_j be their rewards so that $r_i \geq \frac{c_i}{\alpha^{n-1} \cdot (\beta - \alpha)}$ and that $r_j \geq \frac{c_j}{\alpha^{n-2} \cdot \beta \cdot (\beta - \alpha)}$. Then under iterated elimination of dominated strategies, the only remaining strategy for both a_i and a_j is to exert effort.*

P r o o f. We follow steps similar to Lemma 5.1. Since a_i exerts effort, the condition for exerting effort being a dominating strategy for a_j (after eliminating shirking as a strategy for a_i) is that for every $\sigma^b \in \Sigma_{-\{i,j\}}$ we have

$$r_j \geq \frac{c_j}{\Pr_{C_{\sigma^b}(T)} \cdot \beta \cdot (\beta - \alpha)} .$$

Similarly to Theorem 7.3, each $\sigma^b \in \Sigma_{-\{i,j\}}$ requires a different minimal reward r_j, and the smaller $\Pr_{C_{\sigma^b}}(T)$, the higher r_j must be. Again, $\Pr_{C_{\sigma^b}}(T)$ is smallest when all the agents shirk in σ^b, so $C_{\sigma^b} = \emptyset$ and $\Pr_{C_{\sigma^b}}(T) = \alpha^{n-2}$. Thus, if a_i is known to exert effort, a reward $r_j \geq \frac{c_j}{\alpha^{n-2} \cdot \beta \cdot (\beta - \alpha)}$ guarantees that exerting effort is a dominant strategy for a_j. □

We now consider an iterated elimination of dominated strategies incentive-inducing scheme (IE-INI). We first choose an ordering of the agents. We then go through the agents in this 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 (so $\pi : \{1, \ldots, n\} \longrightarrow \{1, \ldots, n\}$ and π is reversible). We denote by Π the set of all such permutations. $\pi(i)$ 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 5.4 IE-INI *is in* P *for the effort game in the above-mentioned specific weighted voting domain. Any reordering of the agents $\pi \in \Pi$, a reward vector r_π where the following holds for any agent a_i is an iterated elimination incentive-inducing scheme:*

$$r_{\pi(i)} \geq \frac{c_{\pi(i)}}{\alpha^{n-i} \cdot \beta^{i-1} \cdot (\beta - \alpha)} .$$

P r o o f. The above definition requirement for the first agent in the reordering, a_i such that $i = \pi(1)$, is exactly as required in Lemma 5.1, and for the second agent exactly as in Lemma 5.3. We can continue the process again. Consider the case where the following two equations hold:

$$r_i \geq \frac{c_i}{\alpha^{n-1} \cdot (\beta - \alpha)} \quad \text{and} \quad r_j \geq \frac{c_j}{\alpha^{n-2} \cdot \beta \cdot (\beta - \alpha)} .$$

If these two hold, we can calculate the minimal reward r_k that makes exerting effort a dominant strategy for yet another agent, a_k. Similarly to the proof of Lemma 5.3, we can see that

$$r_k \geq \frac{c_k}{\alpha^{n-3} \cdot \beta^2 \cdot (\beta - \alpha)} .$$

By induction and using the same proof technique as in Lemma 5.3, we can see that the following reward makes "exert" a dominant strategy for $a_{\pi(i)}$ after i rounds of eliminating dominated strategies:

$$r_{\pi(i)} \geq \frac{c_{\pi(i)}}{\alpha^{n-i} \cdot \beta^{i-1} \cdot (\beta - \alpha)} .$$

This direct formula can be calculated in polynomial time. □

We note that each reordering π of the agents results in a different reward vector. The principal is typically interested in minimizing $\sum_{i=1}^{n} r_i$, or in other words, the principal typically needs to solve MIE-INI (rather than IE-INI). We show that the best ordering is according to the agents' exertion costs $c_i = w_i$.

Theorem 5.5 MIE-INI *is in* P *for the effort game in the above unanimous weighted voting domain. 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 largest.*

Proof. Let $a = (a_{i_1}, a_{i_2}, \dots, a_{i_{k-1}}, a_{i_k}, a_{i_{k+1}}, a_{i_{k+2}}, \dots, a_{i_n})$ and $b = (a_{i_1}, a_{i_2}, \dots, a_{i_{k-1}}, a_{i_{k+1}}, a_{i_k}, a_{i_{k+2}}, \dots, a_{i_n})$ be two reorderings of the agents that are identical except for switching places between the adjacent a_{i_k} and $a_{i_{k+1}}$. Let $r^a = (r^a_1, \dots, r^a_n), r^b = (r^b_1, \dots, r^b_n)$, be the resulting reward vectors for these orderings, as in Theorem 5.4. It suffices to show that if $w_{i_k} > w_{i_{k+1}}$, then $\sum_{i=1}^n r_i^b < \sum_{i=1}^n r_i^a$, since if this holds, we have a "bubble sort" sequence of switches of agent pairs that monotonically drops the total rewards paid, and ends in the ordering of the agents according to their weight. Consider an agent a_i which is in the k'th location in the sequence. That is, the rewards for all the agents in locations $1, 2, \dots, k-1$ are such that after $k-1$ rounds of elimination of dominated strategies, they are all sure to exert effort.

Due to Theorem 5.4, the required reward for it is

$$r_i \geq \frac{c_i}{\alpha^{n-k} \cdot \beta^{k-1} \cdot (\beta - \alpha)} \ .$$

Moving it to the $k+1$ location would make this required reward be

$$r_i \geq \frac{c_i}{\alpha^{n-k-1} \cdot \beta^{k} \cdot (\beta - \alpha)} \ .$$

The ratio between the two required rewards is $q = \dfrac{\alpha}{\beta} < 1$.

Denote by A the required reward for a_{i_k} when he is in the k'th location, so Aq is the required reward when he is in the $k+1$'th location. Denote by B the required reward for $a_{i_{k+1}}$ when he is in the k'th location, so Bq is the required reward when he is in the $k+1$'th location. Thus switching the agent's location in the sequence changes the total rewards from $A + Bq$ to $Aq + B$. We note that $A > B$, due to the equation of Theorem 5.4. Since $q < 1$ and since $A > B$, the change in reward $A + Bq - (Aq + B)$ is positive, so we can improve the principal's situation by switching these agents. Thus, $w_{i_k} > w_{i_{k+1}}$, and then $\sum_{i=1}^n r_i^b < \sum_{i=1}^n r_i^a$. □

We have thus shown that in this domain we can compute MIE-INI by sorting the agents according to their weights (and thus their costs of exerting efforts), and using the equation from Theorem 5.4 to construct the reward vector.[7)]

6 Price of Myopia

We have examined two incentive-inducing schemes for the rewards in effort games: the Dominant Strategy Incentive-Inducing Scheme (DS), and the Iterated Elimination of Dominated Strategies Incentive-Inducing Scheme (IE). Each of these solution concepts represents a different model of rationality that the agents employ. A dominant strategy implementation uses payments that make sure the agents exert effort no matter what they believe the other agents might do. In such an implementation, an agent may assume anything about the behavior of the other agents, and would still find it profitable to exert effort.[8)] An iterated elimination implementation models agents that make more complex assumptions about the behavior of their counterparts. We show that when we assume agents follow the more sophisticated deliberation process of iterated elimination of dominated strategies, it is cheaper for the principal to motivate them, and the total rewards they obtain are smaller.

Let us denote by R_{DS} the minimal sum of the rewards ($\sum_{j=1}^n r_j$) such that for all agents a_i, r_i is a DS, and denote by R_{IE} the minimal sum of the rewards such that for all agents a_i, r_i is an IE. Obviously, $R_{\text{DS}} \geq R_{\text{IE}}$ (since in the IE we are taking into account only a subset of strategies). The natural question here is how large

[7)] Note that in Section 7.2 we discuss effort games over *general* weighted voting games, and show that in that domain it is NP-hard to find an IE-INI, since it is NP-hard to compute the Banzhaf power index in weighted voting games. However, the current section discusses *unanimous* weighted voting games, where the threshold is so high that the only winning coalition is that of *all* the agents. In this domain, all agents have the same Banzhaf power index, which can be calculated using a direct formula, and indeed, for this domain we have shown that MIE-INI is in P.

[8)] In this sense, a dominant strategy implementation is "safe" for myopic agents. This is the reason we choose the name "the price of myopia". Note however that we consider the ratio between a dominant strategy implementation and an *iterated elimination* implementation. Other possibilities (other than a dominant strategy solution) are the best or worst Nash equilibrium, which we do not examine in this paper.

the ratio $R_{\mathrm{DS}}/R_{\mathrm{IE}}$ can be. We call the latter ratio the Price of Myopia (PoM).[9] We study this question in the context of effort games such as in the previous section, where in the underlying coalitional game with task set T, a coalition of tasks $C \subseteq T$ wins if it contains all the tasks, and loses otherwise. All the agents have the same cost for exerting effort c, and the success probability pairs are identical for all tasks (α, β). We prove the following results:

Theorem 6.1 *For the above setting with n agents,* $\dfrac{R_{\mathrm{DS}}}{R_{\mathrm{IE}}} < n$.

Proof. Let a_j be a player. According to Lemma 5.1, exerting effort is a dominant strategy for a_j if and only if the following holds:

$$r_j \ge \frac{c}{\alpha^{n-1} \cdot (\beta - \alpha)} \ .$$

Since this is true for all the players a_j, then

$$(1) \qquad R_{\mathrm{DS}} = \textstyle\sum_{j=1}^{n} r_j = \dfrac{cn}{\alpha^{n-1} \cdot (\beta - \alpha)} \ .$$

By Theorem 5.4 we can obtain an iterated elimination incentive-inducing scheme by taking any reordering of the agents $\pi \in \Pi$, and a reward vector r_π where for any agent a_j we have

$$r_{\pi(j)} \ge \frac{c}{\alpha^{n-j} \cdot \beta^{j-1} \cdot (\beta - \alpha)} \ .$$

Since all the costs c are the same, for all the reorderings π the vectors r_π are permutations of each other, and in particular the sum of the rewards for Iterated Elimination Incentive-Inducing Scheme does not depend on π. So setting π to be the identity permutation, we have

$$(2) \qquad R_{\mathrm{IE}} = \textstyle\sum_{j=1}^{n} r_j = \dfrac{c}{\beta - \alpha} \textstyle\sum_{j=1}^{n} \dfrac{1}{\beta^{j-1}\alpha^{n-j}} \ .$$

Combining (1) and (2) together, we have

$$(3) \qquad \frac{R_{\mathrm{DS}}}{R_{\mathrm{IE}}} = \frac{\dfrac{cn}{(\beta-\alpha)\alpha^{n-1}}}{\dfrac{c}{\beta-\alpha}\sum_{j=1}^{n}\dfrac{1}{\beta^{j-1}\alpha^{n-j}}} = \frac{\dfrac{n}{\alpha^{n-1}}}{\sum_{j=1}^{n}\dfrac{1}{\beta^{j-1}\alpha^{n-j}}} = \frac{n}{\sum_{j=1}^{n}\dfrac{\alpha^{n-1}}{\beta^{j-1}\alpha^{n-j}}} = \frac{n}{\sum_{j=1}^{n}\dfrac{\alpha^{j-1}}{\beta^{j-1}}} \ .$$

Observe that $\dfrac{\alpha^{j-1}}{\beta^{j-1}} = 1$ for $j = 1$ and $\dfrac{\alpha^{j-1}}{\beta^{j-1}} > 0$ for $j \ge 2$. Thus we get $\sum_{j=1}^{n} \dfrac{\alpha^{j-1}}{\beta^{j-1}} > 1$. Hence we obtain

$$\frac{n}{\sum_{j=1}^{n}\dfrac{\alpha^{j-1}}{\beta^{j-1}}} < n.$$

□

Theorem 6.2 *Consider a sequence of effort games D_i for $i = 1, 2, \ldots$. In each effort game D_i all agents have identical success probability pairs (α_i, β_i) and the cost of exerting effort c. Suppose that $\lim_{i\to\infty} \alpha_i = 0$. Also suppose that there exists some $\beta > 0$, such that for all i, $\beta_i \ge \beta$, and denote by $R_{\mathrm{DS}}^{(i)}$ the sum of the rewards for* DS *for game D_i, and by $R_{\mathrm{IE}}^{(i)}$ the sum of the rewards for* IE *for the game D_i. Then:*

$$\lim_{i\to\infty} \frac{R_{\mathrm{DS}}^{(i)}}{R_{\mathrm{IE}}^{(i)}} = n \ .$$

[9] The reason for the name is that the dominant strategy model assumes agents only take into account "short term" or myopic considerations, and do not take into account "longer term" considerations, as done in the iterated elimination model. When the principal assumes agents only have simple and short term considerations he is forced to overpay them.

Proof. We suppose that in sequence of effort games D_i,

(4) $\lim_{i\to\infty} \alpha_i = 0$

and there exists $\beta > 0$ such that

(5) for all i, $\beta_i \geq \beta$.

Then by equation (3) for all i we have the following:

$$\frac{R_{\text{DS}}^{(i)}}{R_{\text{IE}}^{(i)}} = \frac{n}{\sum_{j=1}^{n} \frac{\alpha_i^{j-1}}{\beta_i^{j-1}}}.$$

From (4) and (5) we conclude that for all $j \geq 2$ we have

$$\lim_{i\to\infty} \frac{\alpha_i^{j-1}}{\beta_i^{j-1}} = 0 .$$

Thus we get that

$$\lim_{i\to\infty} \frac{n}{\sum_{j=1}^{n} \frac{\alpha_i^{j-1}}{\beta_i^{j-1}}} = \frac{n}{1} = n.$$

□

Theorem 6.3 *Consider again the sequence of effort games D_i for $i = 1, 2, \ldots$. Suppose that the following holds:*

$$\lim_{i\to\infty} \frac{\alpha_i}{\beta_i} = 1 .$$

Then the following also holds:

$$\lim_{i\to\infty} \frac{R_{\text{DS}}^{(i)}}{R_{\text{IE}}^{(i)}} = 1 .$$

Proof. Suppose that in sequence of games D_i, $\lim_{i\to\infty} \alpha_i/\beta_i = 1$. Then by (3):

$$\lim_{i\to\infty} \frac{R_{\text{DS}}^{(i)}}{R_{\text{IE}}^{(i)}} = \lim_{i\to\infty} \frac{n}{\sum_{j=1}^{n} \frac{\alpha_i^{j-1}}{\beta_i^{j-1}}} = \frac{n}{1 + \ldots + 1} = 1 .$$

□

7 Effort Games Over Series-Parallel Graphs

The previous sections have dealt with a restricted form of effort games, weighted voting effort games, where the underlying game was a weighted voting game. We now generalize our discussion to a wider class of games, defined by Series-Parallel Graphs (SPGs). We first define SPGs.

In this context the term graph refers to a multigraph. A Two-Terminal Graph (TTG) is a graph with two distinguished vertices, s and t called *source* and *sink*, respectively. The *parallel composition* $P = P(X, Y)$ of two TTGs X and Y is a TTG created from the disjoint union of graphs X and Y by merging the sources of X and Y to create the source of P and merging the sinks of X and Y to create the sink of P. The *series composition* $S = S(X, Y)$ of two TTGs X and Y is a TTG created from the disjoint union of graphs X and Y by merging the sink of X with the source of Y. The source of X becomes source of P and the sink of Y becomes the sink of P. A two-terminal series-parallel graph (TTSPG) is a graph that may be constructed by a sequence of series and parallel compositions starting from a set of copies of a single-edge graph K_2 with assigned terminals.

Definition 7.1 A graph is called *series-parallel* (SPG), if it is a TTSPG when some two of its vertices are regarded as source and sink.

We now study the connectivity effort game, which generalizes unanimous weighted voting effort games.[10)] We are given an SPG $G = (V, E)$ which represents a communication network. For each edge $e_i \in E$ there is an agent a_i responsible for maintenance tasks for e_i. a_i can exert an effort at the certain cost c_i to maintain the link e_i, and then it will have the probability β of functioning. Otherwise, when the agent does not exert effort to maintain the link, it will have the probability $\alpha < \beta$ of functioning. The winning coalitions in our game are those who contain a path from the source node to the sink node. Note that a special case of this setting is the weighted voting effort game, where the quota for accepting the decision is $q = \sum_{i=1}^{n} w_i$: it is a special case of SPG where all the compositions are series compositions. However, the SPG setting is much more general, as it allows parallel connections.[11)]

7.1 Price of Myopia in SPG Domains: Simulation Results

We now present the results of several simulations we have carried out in order to investigate the price of myopia in SPG effort games. We first describe our simulation settings.

All our experiments were carried out on SPGs with 7 edges, whose composition is described by the following binary tree. The leaves of the tree represent edges $1, 2, \dots, 7$. Every inner node of the tree represents a TTSPG which is a series or parallel composition of TTSPGs represented by its children.

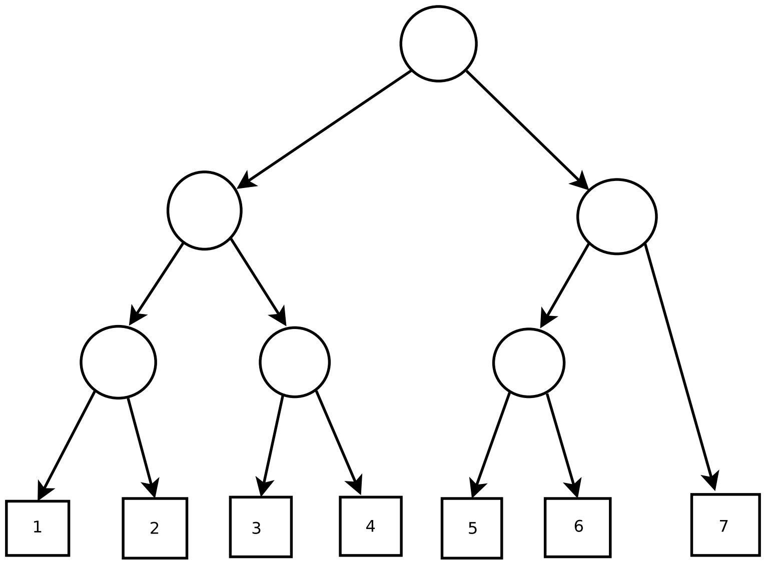

Fig. 1: Binary tree representing the SPGs used in the simulations

Among these SPGs we have sampled uniformly at random SPGs by giving the probability of $\frac{1}{2}$ for series composition and the same probability for parallel composition at each inner node of the above tree. We have computed the PoM for $\alpha = 0.1, 0.2, \dots, 0.8$ and $\beta = \alpha + 0.1, \alpha + 0.2, \dots, 0.9$ where the costs c_i have been sampled uniformly at random in $[0.0, 100.0)$. For each pair (α, β) 500 experiments have been made in order to find the average PoM, and the standard deviation of the PoM. The results of the experiments are summarized in the tables below:

[10)] Note that connectivity games only generalize *unanimous* weighted voting domains, and not *general* weighted voting domains.

[11)] In fact, SPGs can allow us to conjunctions and disjunctions of tasks to be accomplished that are "winning", and is, in this sense, quite a general language for expressing many simple coalitional games.

The average PoM:

α β	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
0.1	4.22	9.43	16.26	25.27	38.04	58.04	94.45	177.85
0.2		2.27	4.15	6.70	10.12	14.98	22.64	37.64
0.3			1.76	2.82	4.25	6.21	9.08	14.12
0.4				1.54	2.26	3.22	4.55	6.67
0.5					1.41	1.95	2.67	3.71
0.6						1.33	1.76	2.34
0.7							1.28	1.64
0.8								1.24

The standard deviation of the PoM:

α β	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
0.1	3.05	10.23	21.85	39.99	69.40	119.72	221.13	472.22
0.2		1.00	3.01	6.30	11.43	19.68	34.17	65.05
0.3			0.57	1.60	3.27	5.92	10.39	19.52
0.4				0.39	1.07	2.14	3.91	7.28
0.5					0.30	0.80	1.60	3.02
0.6						0.25	0.65	1.32
0.7							0.22	0.56
0.8								0.20

We give here an informal discussion of the simulation results. As one can see from the first table, the higher the distance between α and β, the higher the PoM is, so when there are large differences in the probabilities, it really matters which model of rational behavior we choose. This also makes the structure of the graph more important. These results somehow fit our expectations, since the smaller the difference between α and β, the smaller is the difference between exerting effort and shirking, and the less significant is the knowledge whether some players exert effort or not.

We now prove that in parallel connection the reward in Iterated Elimination Inducing Scheme does not drop, or in other words, $R_{\mathrm{DS}}/R_{\mathrm{IE}} = 1$.

Theorem 7.2 *Let G be* SPG *which is obtained by parallel composition of a set of copies of single-edge graphs* K_2*. As before each agent is responsible for a single edge, and a winning coalition is one which contains a path* (*an edge*) *connecting the source and target vertices. Then, in the notation of Theorem* 6.1, *we have* $R_{\mathrm{DS}} = R_{\mathrm{IE}}$.

P r o o f. By Theorem 7.3 the condition for exerting effort being a dominant strategy for a_i is that

(6) $$\text{for every } \sigma^a \in \Sigma_{-i} : r_i \geq \frac{c_i}{\sum_{T_{\mathrm{w}} \in T_{\mathrm{win}}} (Pr_{C_{\sigma_e^a}}(T_{\mathrm{w}}) - Pr_{C_{\sigma_s^a}}(T_{\mathrm{w}}))} .$$

Since the only set of edges that is not in T_{win} is the empty set, we have

$$\frac{c_i}{\sum_{T_{\mathrm{w}} \in T_{\mathrm{win}}} (Pr_{C_{\sigma_e^a}}(T_{\mathrm{w}}) - Pr_{C_{\sigma_s^a}}(T_{\mathrm{w}}))} = \frac{c_i}{1 - Pr_{C_{\sigma_e^a}}(\emptyset) - (1 - Pr_{C_{\sigma_s^a}}(\emptyset))} .$$

Denote by x_i the probability that the edge i will exist under σ^a, i.e.,

$$x_i = \begin{cases} \alpha_i & \text{if agent } i \text{ shirks,} \\ \beta_i & \text{if agent } i \text{ exerts effort.} \end{cases}$$

Then we have

$$r_i \geq \frac{c_i}{1 - Pr_{C_{\sigma_e^a}}(\emptyset) - (1 - Pr_{C_{\sigma_s^a}}(\emptyset))}$$

(7)
$$= \frac{c_i}{1 - \prod_{j=1,j\neq i}^{n}(1-x_j)\cdot(1-\beta_i) - (1 - \prod_{j=1,j\neq i}^{n}(1-x_j)\cdot(1-\alpha_i))}$$

$$= \frac{c_i}{\prod_{j=1,j\neq i}^{n}(1-x_j)(\beta_i-\alpha_i)} \ .$$

The above inequality must hold for each profile σ^a, i.e., for every choice of x_i, the smaller $\prod(1-x_j)$ is, the bigger r_i should be. $\prod(1-x_j)$ is the smallest when for all j, $x_j = \beta_j$, and so condition (6) is equivalent to the condition

$$r_i \geq \frac{c_i}{\prod_{j=1,j\neq i}^{n}(1-\beta_j)(\beta_i-\alpha_i)} \ .$$

Now suppose that a_i is known to exert effort. The condition for exerting effort being a dominating strategy for a_j is that for every $\sigma^b \in \Sigma_{-\{i,j\}}$ we have

$$r_j \geq \frac{c_j}{\prod_{k=1,k\neq i,j}^{n}(1-x_k)(1-\beta_i)(\beta_j-\alpha_j)} \ .$$

This is equivalent to

$$r_j \geq \frac{c_j}{\prod_{k=1,k\neq j}^{n}(1-\beta_k)(\beta_j-\alpha_j)} \ .$$

Similarly, it is proven by induction that for each reordering of the players π, the following reward is an Iterated Elimination Incentive-Inducing Scheme:

$$r_{\pi(i)} = \frac{c_{\pi(i)}}{\prod_{j=1,j\neq \pi(i)}^{n}(1-\beta_j)(\beta_{\pi(i)}-\alpha_{\pi(i)})} \ .$$

Note that in Dominant Strategies Incentive-Inducing Scheme the reward is also (as the reward of the first player):

$$\frac{c_{\pi(i)}}{\prod_{j=1,j\neq \pi(i)}^{n}(1-\beta_j)(\beta_{\pi(i)}-\alpha_{\pi(i)})} \ .$$

Thus we get $R_{\mathrm{DS}} = R_{\mathrm{IE}}$. □

7.2 Rewards and the Banzhaf Power Index

We now generalize the discussion to general effort games, where the underlying coalitional game may be any simple coalitional game. We show the relation between the complexity of the problems defined in Section 4, and power indices in the underlying coalitional game.

Theorem 7.3 *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 $\alpha_j = \beta_j = \frac{1}{2}$, and let $r = (r_1, \ldots, r_n)$ be a reward vector. Exerting effort is a dominant strategy for a_i in $G_{\mathrm{e}}(r)$ if and only if $r_i > \frac{c_i}{\beta_i(v)}$ (where $\beta_i(v)$ is the Banzhaf power index of t_i in the underlying coalitional game G, with the value function v).*

Proof. Agent a_i only has two strategies in $G_{\mathrm{e}}(r)$: $S_i = \{\mathrm{exert}, \mathrm{shirk}\}$. As in Definition 3.9, a_i's payoff in $G_{\mathrm{e}}(r)$ when exerting effort is $e_i(C_\sigma) - c_i$, and $e_i(C_\sigma)$ when not exerting effort. Given an incomplete strategy profile σ_{-i}^a (with a strategy missing for a_i), we denote its completion with a_i exerting effort as $\sigma_e^a = (\sigma_{-i}, s_i = \mathrm{exert})$, and its completion when a_i shirks as $\sigma_s^a = (\sigma_{-i}, s_i = \mathrm{shirk})$. σ_e^a and σ_s^a are two complete strategy profiles, which are identical in all strategies, except that of a_i.

For exerting effort to be a dominant strategy, a_i's payoff when exerting effort must be greater than its payoff when shirking, no matter what the other agents do. Thus, for exerting effort to be a dominant strategy the following must hold: for *all* incomplete strategy profiles σ^a_{-i} we have that $e_i(C_{\sigma^a_e}) - c_i > e_i(C_{\sigma^a_s})$.

We show that the above condition holds if and only if $r_i > \dfrac{c_i}{\beta_i(v)}$. The above condition can be restated as

$$\textstyle\sum_{T_{\mathrm{w}} \in T_{\mathrm{win}}} r_i \Pr_{C_{\sigma^a_e}}(T_{\mathrm{w}}) - c_i > \sum_{T_{\mathrm{w}} \in T_{\mathrm{win}}} r_i \Pr_{C_{\sigma^a_s}}(T_{\mathrm{w}})$$

or more briefly as

$$r_i > \frac{c_i}{\sum_{T_{\mathrm{w}} \in T_{\mathrm{win}}} (\Pr_{C_{\sigma^a_e}}(T_{\mathrm{w}}) - \Pr_{C_{\sigma^a_s}}(T_{\mathrm{w}}))} \ .$$

It remains to show that

$$\textstyle\sum_{T_{\mathrm{w}} \in T_{\mathrm{win}}} (\Pr_{C_{\sigma^a_e}}(T_{\mathrm{w}}) - \Pr_{C_{\sigma^a_s}}(T_{\mathrm{w}})) = \beta_i(v) \ .$$

Let $T_{\mathrm{w}} \in T_{\mathrm{win}}$ be a task subset so that $t_i \in T_{\mathrm{w}}$. Since $\alpha_i = 0$ and $\beta_i = 1$ and since for any $j \neq i$ we have $\alpha_j = \beta_j = \dfrac{1}{2}$, for any σ^a_{-i} we have

$$\Pr_{C_{\sigma^a_e}}(T_{\mathrm{w}}) = (\tfrac{1}{2})^{n-1} \quad \text{and} \quad \Pr_{C_{\sigma^a_s}}(T_{\mathrm{w}}) = 0 \ .$$

On the other hand, let $T_{\mathrm{w}} \in T_{\mathrm{win}}$ be a task subset so that $t_i \notin T_{\mathrm{w}}$. Due to the same reason, for any σ^a_{-i} we have

$$\Pr_{C_{\sigma^a_e}}(T_{\mathrm{w}}) = 0 \quad \text{and} \quad \Pr_{C_{\sigma^a_s}}(T_{\mathrm{w}}) = (\tfrac{1}{2})^{n-1} \ .$$

We thus get:

$$\begin{aligned} &\textstyle\sum_{T_{\mathrm{w}} \in T_{\mathrm{win}}} (\Pr_{C_{\sigma^a_e}}(T_{\mathrm{w}}) - \Pr_{C_{\sigma^a_s}}(T_{\mathrm{w}})) \\ &= \textstyle\sum_{T_{\mathrm{w}} \in T_{\mathrm{win}} | t_i \in T_{\mathrm{w}}} (\Pr_{C_{\sigma^a_e}}(T_{\mathrm{w}}) - \Pr_{C_{\sigma^a_s}}(T_{\mathrm{w}})) + \sum_{T_{\mathrm{w}} \in T_{\mathrm{win}} | t_i \notin T_{\mathrm{w}}} (\Pr_{C_{\sigma^a_e}}(T_{\mathrm{w}}) - \Pr_{C_{\sigma^a_s}}(T_{\mathrm{w}})) \\ &= \textstyle\sum_{T_{\mathrm{w}} \in T_{\mathrm{win}} | t_i \in T_{\mathrm{w}}} ((\frac{1}{2})^{n-1} - 0) + \sum_{T_{\mathrm{w}} \in T_{\mathrm{win}} | t_i \notin T_{\mathrm{w}}} (0 - (\frac{1}{2})^{n-1}) \\ &= (\tfrac{1}{2})^{n-1} \cdot [\textstyle\sum_{T_{\mathrm{w}} \in T_{\mathrm{win}} | t_i \in T_{\mathrm{w}}} 1 - \sum_{T_{\mathrm{w}} \notin T_{\mathrm{win}} | t_i \in T_{\mathrm{w}}} 1] \\ &= \beta_i(v) \ . \end{aligned}$$

Note that the final equality requires that v is increasing, so every winning coalition C that $a_i \notin C$ also wins when a_i is added to that coalition, so $C \cup \{a_i\}$ also wins. □

Consider an effort game domain D, the reward vector r, and the resulting effort game $G_{\mathrm{e}}(r)$. We now show that DSE, testing whether exerting effort is 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 known to be #P-hard in various domains (see Section 2), this shows that in general DSE is also #P-hard.

Theorem 7.4 DSE *is as hard computationally as calculating the Banzhaf power index of its underlying coalitional game G.*

Proof. Consider a simple coalitional game G with a characteristic function v. We reduce the problem of calculating $\beta_i(v)$, the Banzhaf index of agent t_i in G, to a polynomial number of DSE problems. There are 2^{n-1} possible values $\beta_i(v)$ can take: $\dfrac{1}{2^{n-1}}, \dfrac{2}{2^{n-1}}, \dfrac{3}{2^{n-1}}, \dots, 1$. We perform a binary search on the correct value of β_i, by using DSE queries. We construct an effort game domain, that contains an agent a_j for each task t_j in G. The success probabilities for a_i are $\alpha_i = 0$ and $\beta_i = 1$. For all other agents $a_j \neq a_i$ the success probabilities are $\alpha_j = \beta_j = \frac{1}{2}$. The effort exertion costs are $c_i = 1$ for all the agents. The first DSE query is regarding the reward vector of $r = (r_1 = 0, r_2 = 0, \dots, r_{i-1} = 0, r_i = 2 \cdot c_i, r_{i+1} = 0, \dots, r_n = 0)$. Due to Theorem 7.3, if the DSE – answer is "yes", the following must hold: $r_i = 2 \cdot c_i > \dfrac{c_i}{\beta_i(v)}$, or equivalently $\beta_i(v) > \dfrac{1}{2}$. Similarly, to test if $\beta_i > \frac{1}{k}$, we simply test DSE with the reward vector $r_1 = 0, r_2 = 0, \ldots, r_{i-1} = 0, r_i = k \cdot c_i$, $r_{i+1} = 0, \ldots, r_n = 0$. Thus, we are able to find β_i in $O(\log_2(2^{n-1})) = O(n)$, so a polynomial procedure for DSE can be used to construct a polynomial procedure for calculating the Banzhaf power index in the underlying game. □

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 whether exerting effort is a dominant strategy.

8 Conclusions and Future Work

We defined the effort game model of cooperation. We started by studying effort games where the underlying coalitional game is a restricted class of weighted voting games. We have shown that for this domain, we can answer several questions regarding the incentives in polynomial time. Namely, we have discussed how to compute optimal reward schemes in such effort-weighted voting games.

We defined the Price of Myopia (PoM), that quantifies by how much the principal must overpay agents when assuming they only have "myopic" considerations, and provided results regarding PoM in weighted voting effort games. We have generalized weighted voting effort games to effort games over Series-Parallel Graphs (SPGs), and gave simulation results regarding PoM in that domain as well. Finally, we have discussed how a general effort game relies on an underlying coalitional game, and showed that the complexity of testing whether exerting effort is a dominant strategy is at least as hard as calculating the Banzhaf power index in the underlying game. Defining the relation between computing incentives in an effort game and calculating the Banzhaf index in the underlying game allows us to easily use complexity hardness results regarding power indices to show complexity results in effort games.

It remains an open problem to examine the complexity of inducing incentives for other classes of underlying games. It also remains an open problem to consider the relation between the computational complexity of inducing effort in effort games, and the computational complexity of other problems in the underlying coalitional game. For example, we believe that inducing incentives in effort games is at least as hard computationally as various counting problems. Given our hardness results for general effort games, it will be interesting to see if incentive-inducing schemes can be approximated in polynomial time. It would also be interesting to study the price of myopia in various classes of effort games.

9 Appendix – Strategy Changes and Expected Rewards: Proofs

For the sake of completeness, and in order to ease the reading of this paper, we now provide proofs for the theorems in Section 3.2.1. Similar results may be found in some of the papers mentioned in Section 2.

We begin with the proof of Theorem 3.13.

Theorem 9.1 *Let D be an effort game domain, $r = (r_1, \ldots, r_n)$ be a reward vector, and a_i be a certain agent in that domain. Let $\sigma_1, \sigma_2 \in \Sigma$ be two strategy profiles in $G_{\mathrm{e}}(r)$ so that for all $a_j \neq a_i$ we have that $\sigma_{1,j} = \sigma_{2,j}$, and that $\sigma_{1,i} = \mathrm{exert}$ and $\sigma_{2,i} = \mathrm{shirk}$. Then for all j we have that $e_j(C_{\sigma_1}) > e_j(C_{\sigma_2})$.*

P r o o f. The only difference between σ_1 and σ_2 is the strategy used by a_i, who shirks in σ_2 but exerts effort in σ_1. Agent a_i is in charge of task t_i. C_σ is the set of agents that exert effort in strategy profile σ, so $C_{\sigma_1} = C_{\sigma_2} \cup \{a_i\}$.

Consider a task subset T_{x}. If $t_i \notin T_{\mathrm{x}}$, denote $T_{\mathrm{x}}^d = T_{\mathrm{x}} \cup \{t_i\}$, and if $t_i \in T_{\mathrm{x}}$ denote $T_{\mathrm{x}}^d = T_{\mathrm{x}} \setminus \{t_i\}$. The notation in Section 3.2 denoted the probability that exactly the tasks in T_{x} are the ones achieved as $\Pr_C(T_{\mathrm{x}}) = \prod_{t_i \in T_{\mathrm{x}}} p_i(C) \cdot \prod_{t_i \notin T_{\mathrm{x}}} (1 - p_i(C))$. We note that if $t_i \in T_{\mathrm{x}}$, then

$$\begin{aligned}\Pr_{C_{\sigma_1}}(T_{\mathrm{x}}) &= \textstyle\prod_{t_j \in (T_{\mathrm{x}} \setminus \{t_i\})} p_j(C_{\sigma_1}) \cdot \prod_{t_j \in (T \setminus T_{\mathrm{x}} \setminus \{t_i\})} (1 - p_i(C_{\sigma_1})) \cdot \beta_i \\ &> \textstyle\prod_{t_j \in (T_{\mathrm{x}} \setminus t_i)} p_j(C_{\sigma_1}) \cdot \prod_{t_j \in (T \setminus T_{\mathrm{x}} \setminus \{t_i\})} (1 - p_i(C_{\sigma_1})) \cdot \alpha_i \\ &= \Pr_{C_{\sigma_2}}(T_{\mathrm{x}}).\end{aligned}$$

Thus, if $t_i \in T_{\mathrm{x}}$, then $\Pr_{C_{\sigma_1}}(T_{\mathrm{x}}) > \Pr_{C_{\sigma_2}}(T_{\mathrm{x}})$.

We denote the probability of completing T_x given a *partial* strategy profile σ_{-i} (with a missing strategy for a_i) as $\Pr_{C_{\sigma_{-i}}}(T_x) = \prod_{t_j \in (T_x \setminus \{t_i\})} p_j(C) \cdot \prod_{t_j \in (T \setminus T_x \setminus \{t_i\})} (1 - p_j(C))$. We note that this probability ignores the question of whether t_i was completed (even if $t_i \in T_x$) and only takes into consideration the tasks in $T_x \setminus \{t_i\}$. Given this notation, we can see that for any task subset T_x we have that

$$\begin{aligned}\Pr_{C_{\sigma_1}}(T_x) + \Pr_{C_{\sigma_1}}(T_x^d) &= \Pr_{C_{\sigma_{-i}}}(T_x) \cdot \beta_i + \Pr_{C_{\sigma_{-i}}}(T_x) \cdot (1 - \beta_i) \\ &= \Pr_{C_{\sigma_{-i}}}(T_x) \cdot \alpha_i + \Pr_{C_{\sigma_{-i}}}(T_x) \cdot (1 - \alpha_i) \\ &= \Pr_{C_{\sigma_2}}(T_x) + \Pr_{C_{\sigma_2}}(T_x^d).\end{aligned}$$

In Section 3.2 we defined the expected reward as $e_i(C_\sigma) = \sum_{T_w \in T_{win}} \Pr_C(T_w) \cdot r_i$. We have assumed the underlying coalitional game G is increasing (Section 3), so if a task subset T_a wins, so does any superset of it, so if $T_a \subset T_b$ then $T_a \in T_{win} \Rightarrow T_b \in T_{win}$. Consider a *winning* task subset $T_w \in T_{win}$. Either $t_i \in T_w$ or $t_i \notin T_w$. If $t_i \notin T_w$, then T_w^d is also winning and $T_w^d \in T_{win}$ since $T_w \in T_{win}$ and G is increasing (and $T_w \subset T_w^d$). Thus, for any winning task subset $T_w \in T_{win}$ either $t_i \in T_w$ or T_w^d is also winning, so $T_w^d \in T_{win}$. Since $e_k(C_\sigma) = r_k \cdot \sum_{T_w \in T_{win}} \Pr_{C_\sigma}(T_w)$, we can express it as:
$e_k(C_\sigma) = r_k \cdot ((\frac{1}{2} \sum_{T_w \in \{T' \in T_{win} | T' \in T_{win} \wedge T'^d \in T_{win}\}} (\Pr_{C_\sigma}(T_w) + \Pr_{C_\sigma}(T_w^d))) + \sum_{T_w \in \{T' \in T_{win} | t_i \in T'\}} Pr_{C_\sigma}(T_w))$.
Since when moving from C_{σ_2} to C_{σ_1} the first sum remains the same and the second increases, we have that $e_k(C_{\sigma_1}) > e_k(C_{\sigma_2})$. We chose a_k to be any agent, so this holds for all agents. □

We now provide the proof for Theorem 3.14.

Theorem 9.2 *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 $e_i(C_{\sigma_1}) > e_i(C_{\sigma_2})$.*

Proof. Since $\sigma_1 >_e \sigma_2$, the only difference between σ_1 and σ_2 are several agents that exert effort in σ_1 but shirk in σ_2. It is thus possible to create a series of strategy profiles $\sigma_2 <_e \sigma'_1 <_e \sigma'_2 <_e \dots <_e \sigma'_k <_e \sigma_1$, such that σ'_j and σ'_{j+1} are identical, except for one agent that shirks in σ'_j but exerts effort in σ'_{j+1}. We can then apply Theorem 3.13, and get that $e_i(C_{\sigma'_{j+1}}) > e_i(C_{\sigma'_j})$, and that $e_i(C_{\sigma_1}) > e_i(C_{\sigma_2})$. □

References

[1] M. Babaioff, M. Feldman, and N. Nisan, Combinatorial agency. In: ACM EC'06(2006).
[2] Y. Bachrach and J. S. Rosenschein, Computing the Banzhaf power index in network flow games. In: AAMAS'07, pp. 323–329, (Honolulu, 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: EC-2005, pp. 88–97 (2005).
[5] X. Deng and C. H. Papadimitriou, On the complexity of cooperative solution concepts. Math. Oper. Res. **19**, 257–266 (1994).
[6] E. Elkind, L. A. Goldberg, P. W. Goldberg, and M. Wooldridge, A tractable and expressive class of marginal contribution nets and its applications. In: The International Conference on Autonomous Agents and Multi-Agent Systems (L. Padgham, D. C. Parkes, J. Muller, and S. Parsons, eds.), pp. 1007–1014 (IFAAMAS, 2008).
[7] I. Gilboa, E. Kalai, and E. Zemel, The complexity of eliminating dominated strategies. Math. Oper. Res. **18**, 553 – 565 (1993).
[8] B. Holmstrom, Moral hazard in teams. Bell Journal of Economics **13**, 324–340 (1982).
[9] D. E. Knuth, C. H. Papadimitriou, and J. N. Tsitsiklis, A note on strategy elimination in bimatrix games. Operations Research Letters **7**, 103–107 (1988).
[10] Y. Matsui and T. Matsui, NP-completeness for calculating power indices of weighted majority games. Theoretical Computer Science **263**, 305–310 (2001).
[11] D. Monderer and M. Tennenholtz, k-Implementation. In EC '03: Proceedings of the 4th ACM conference on Electronic commerce, pp. 19–28 (ACM, 2003).
[12] M. J. Osborne and A. Rubinstein, A Course in Game Theory (MIT Press, 1999).
[13] L. S. Shapley, A value for n-person games. In: Contributions to the Theory of Games, pp. 31–40 (1953).
[14] L. S. Shapley and M. Shubik, A method for evaluating the distribution of power in a committee system. American Political Science Review **48**, 787–792 (1954).
[15] E. Winter, Incentives and discrimination. American Economic Review **94**, 764–773 (2004).

