Creating incentives to prevent execution failures: an extension of VCG mechanisms

Yingqian Zhang and Mathijs de Weerdt

Delft University of Technology, The Netherlands
Yingqian.Zhang@tudelft.nl, M.M.deWeerdt@tudelft.nl

Abstract. When information or control in a multiagent system is private to the agents, they may misreport this information or refuse to execute an agreed outcome, in order to change the resulting end state of such a system to their benefit. In some domains this may result in an execution failure. We show that in such settings VCG mechanisms lose truthfulness, and that the utility of truthful agents can become negative when using VCG payments (i.e., VCG is not strongly individually rational). To deal with this problem, we introduce an extended payment structure which takes into account the actual execution of the promised outcome. We show that this extended mechanism can guarantee a nonnegative utility and is (i) incentive compatible in a Nash equilibrium, and (ii) incentive compatible in dominant strategies if and only if all agents can be verified during execution.

1 Introduction

A multiagent system often involves a set of self-interested agents, which may manipulate the system by mis-reporting their private information. Research into mechanism design is about creating incentives for such self-interested agents to report the correct information. Such a mechanism usually consists of (i) a social choice function, which selects a socially optimal outcome given the declaration of agents; and (ii) a set of payments, which decide for each agent how much it pays (or receives) to (or from) the mechanism. Unfortunately, the outcome selected by a mechanism may fail to execute for two main reasons: agents may have mis-reported some capabilities, but they are *not able to deliver on* these promises during execution, or they may simply *refuse to execute* (part of) their tasks. Rational agents act in such a way if they can profit from it.

Most existing work in mechanism design assumes that, although misreports may lead to sub-optimal outcomes, the outcome selected by the mechanism is always attainable, and that the agents indeed are willing to do their part in the execution. However, there are many settings in which an outcome cannot be enforced. For example in multiagent planning domains, scheduling domains with precedences, and all kinds of domains where agents need to declare capacities or capabilities, the information that agents give directly influences the set of feasible outcomes. Moreover, in almost any domain the refusal of an agent to execute its part leads to a (partial) failure. In such settings we want to create incentives for the agents in order to make sure that any selected outcome is attained.

Dagstuhl Seminar Proceedings 08461
Planning in Multiagent Systems
http://drops.dagstuhl.de/opus/volltexte/2009/1870

For example, consider a multiagent planning problem (MAP) [5, 12], which is concerned with planning by and for a group of (self-interested) agents. Such a MAP contains a private, individual planning problem for each agent. A typical individual planning problem of agent i includes a set of operations (with some costs attached, and a pre- and post-condition) that i can perform, a set of goals (with reward values), and the current state of this agent. The solution of a MAP is a plan: a partially ordered sequence of actions that, when executed successfully, results in a set of achieved goals for some of the agents. The utility of a plan is defined as the difference between the total reward of the achieved goals and the total cost of the actions used. The mechanism design problem of MAP is, given all agents' declared private planning problems, to determine the plan that has the highest utility as well as the payments of all agents.

Example 1. As a simple example of a multiagent planning problem, let there be two agents. Agent 1 has a goal which is to complete task t_1. Completing t_1 requires operations (also called actions) a_1 and a_2 for which we also have a precedence relation $a_1 \prec a_2$ (i.e., a_1 has to be executed before a_2). Agent 1 itself is able to perform only action a_1, with cost $c_1(a_1) = 8$. The reward of achieving t_1 is 10. Agent 2 does not have any goal, but can execute action a_2 with cost $c_2(a_2) = 1$. The optimal plan ω for this planning problem is to execute a_1 and then a_2 such that the goal t_1 can be obtained. The utility of this plan is $10 - 8 - 1 = 1$.

However, if either agent 1 or agent 2 refuses to execute its action, the plan ω will fail to be attained. This situation is highly undesirable.

A basic mechanism design model does not consider the actual execution on the outcome. So, the payment phase is always before the execution. Therefore, one may wonder whether a simple solution to such unsuccessful executions in the example could be handing over payments *after* the selected plan has been executed. Unfortunately, it is not just a matter of having the payment phase *before* or *after* the execution. In this example, using the VCG mechanism[1], the payments of agent 1 and agent 2 are 1 (paying to the mechanism) and -2 (receiving from the mechanism), respectively. Hence, the plan may fail in each case: (i) in the case of payment up-front, agent 2 will leave before executing, directly after it *receives* money from the mechanism; (ii) in the case of payment afterwards, agent 1 may leave before the payment phase to save money by *not paying*.

In this paper we deal with this problem by asking agents for a *deposit* before execution. In addition, we also generalize an idea put forward by Nisan and Ronen [9]. Their idea was to take advantage of the actual execution to gain more information about the private information of the agents. They distinguish between two phases in a mechanism: a *declaration phase* where agents declare their private information, and an *execution phase* where agents actually execute the agreed outcome. Their mechanism verifies the agents' declarations during the execution phase, and awards payments agents after execution. Their approach relies on the following two assumptions: (1) all declared information of the agents in the selected outcome can be verified during execution, and (2) the payments of the agents can be guaranteed. They show that the proposed mech-

[1] We will describe the VCG mechanism in Section 3.

anism, called *mechanism with verification*, is truthful for a task scheduling problem, as any agent's real execution time can be measured in the exectued schedule.

Unfortunately, without the first assumption, this mechanism can not guarantee any properties on truthfulness or individual rationality. Moreover, these two assumptions are not always realistic. First of all, in most multiagent applications, due to their highly distributed nature, it is very difficult or even impossible to verify the declared information of the agents during execution. Secondly, as we already mentioned after Example 1, a lying agent may just walk away without bothering with the payment. It is not always possible that the mechanism can enforce the payments of the agents after execution.

Hence, in this paper, we relax these two assumptions. We characterize the problem domains where the approach in [9] is feasible, and we present a mechanism where truthtelling leads to a Nash equilibrium without any condition on verification.

Our main contributions thus are that we show that for settings where the execution of agents may fail,

- the VCG mechanism is not truthful nor strongly individually rational (in Section 3),
- an extension of the VCG mechanism with a more complex payment structure is both truthful and strongly individually rational if and only if the execution can be verified (Section 4), and
- such a mechanism is both strongly individually rational and incentive compatible in Nash (Section 4).

In this paper we restrict ourselves to direct mechanisms with payments, as these usually give a good indication of the (im)possibilities of other variants (because of the revelation principle and the Gibbard-Satterthwaite theorem [6]). To arrive at these results, we first introduce some notation and we define different misreporting types (over- and under-reporting) formally (in Section 2).

2 Notation and Definitions

Let the *type*, i.e., the private information, of each agent $i \in N$ be denoted by θ_i. We let Θ_i be the allowable subset of types for each $i \in N$, and let Θ denote $\Theta_1 \times \cdots \times \Theta_n$. A *type profile* θ is a vector $(\theta_1, \ldots, \theta_n) \in \Theta$ associating each agent with a type. A *direct-revelation mechanism* M defines a function $g : \Theta \to \Omega$, which outputs outcomes given types Θ.

Since in some settings not all outcomes in Ω are *feasible given a true type profile* θ, i.e. executable given θ, we assume that in each domain there is a function $F : \Theta \to 2^{\Omega}$ which defines the set of all feasible outcomes given the (true) preference profile θ. Note however, that when another profile $\hat{\theta}$ is given, with $\hat{\theta} \neq \theta$, then typically $F(\hat{\theta})$ does not represent feasible outcomes, but only the hypothetically feasible outcomes in case $\theta = \hat{\theta}$. We use this fact to define over-reporting and under-reporting.

If for agent i the set of (hypothetically) feasible outcomes given $\hat{\theta}_i$ is not contained in the true set of feasible outcomes (given θ_i), either because the agent is lying, or it just is not going to execute its part, we say that agent i over-reports.

Definition 1. *Given a preference profile θ, an agent* over-reports *its declared type $\hat{\theta}_i$, iff $F(\hat{\theta}) \supset F(\theta)$.*

For example, if agent 1 in Example 1 declares it can also execute action a_2 (which it cannot), we say that agent 1 is over-reporting. In addition, we also say that agent 2 is over-reporting if agent 2 declares its action a_2 but refuses to execute it later during execution.

Conversely, if agent 1 declares it cannot do any of the actions, the set of feasible outcomes given such a declaration $\hat{\theta}_i$ is strictly contained in the true set of feasible outcomes (given θ_i). In such a case we say that agent i under-reports.

Definition 2. *Given a preference profile θ, an agent* under-reports *its declared type $\hat{\theta}_i$, iff $F(\hat{\theta}) \subset F(\theta)$.*

Besides over- and under-reporting we of course also consider *traditional lying* scenarios where declaring a type $\hat{\theta} \neq \theta$ does not influence the set of feasible outcomes, i.e., where $F(\hat{\theta}) = F(\theta)$. Notice that under-reporting can be considered as a special case in traditional lying, by declaring a cost of ∞ for the parts that an agent does not want to execute.

The preferences of an agent i with type θ_i are defined by the valuation function $v_i(\omega, \theta_i)$ which assigns a value to each outcome $\omega \in \Omega$. However, in case an outcome is infeasible, the actual valuation is only defined on the state of the world *after the execution has failed*. One can think of this as there being some function $e : \Omega \times \Theta \to \Omega$ that given the true types of all agents and an outcome selected by a mechanism, returns the outcome that is really achieved. Obviously, if no agent is lying, then for any outcome ω selected by a mechanism, it holds that $e(\omega, \theta) = \omega$. In this paper, we thus use $v_i(e(\omega, \theta), \theta_i)$ to denote the actual valuation of agent i on a selected outcome ω.

Our goal in this paper is to design direct mechanisms where it is each agent's dominant strategy to declare its true type. Following existing literature, we call such mechanisms *incentive compatible in dominant strategies*, or simply *truthful*. We approach our goal by first discussing truthfulness separately for three types of misreporting: traditional lying, over-, and under-reporting, and then extending the discussions to any mixed type of misreporting. We use θ_{-i} to denote the type profile $(\theta_1, \ldots, \theta_{i-1}, \theta_{i+1}, \ldots, \theta_N)$.

Definition 3. *A mechanism is* truthful with respect to over-reporting *iff for any agent i with type θ_i that is over-reporting with $\hat{\theta}_i$, for any true or over-reporting type profile $\hat{\theta}_{-i}$, the outcome $e\left(g(\hat{\theta}_i, \hat{\theta}_{-i}), \theta\right)$ is strictly more preferred by i than $g(\theta_i, \hat{\theta}_{-i})$.*

Definition 4. *A mechanism is* truthful with respect to under-reporting *iff for any agent i with type θ_i that is under-reporting with $\hat{\theta}_i$, for any true or under-reporting type profile $\hat{\theta}_{-i}$, the outcome $e\left(g\left(\hat{\theta}_i, \hat{\theta}_{-i}, \theta\right)\right)$ is strictly more preferred by i than $g(\theta_i, \hat{\theta}_{-i})$.*

Truthfulness for traditional lying is defined in the usual way [8]. A mechanism is truthful if it is truthful with respect to *any* kind of misreporting.

Definition 5. *A mechanism is* truthful *iff for any agent i with type $\hat{\theta}_i \neq \theta_i$, given any type profile $\hat{\theta}_{-i}$, the outcome $e\left(g(\hat{\theta}_i, \hat{\theta}_{-i}), \theta\right)$ is strictly more preferred by i than $g(\theta_i, \hat{\theta}_{-i})$.*

Given that the only truthful mechanisms for unrestricted domains without payments are *dictatorships* (by Gibbard and Satterthwaite [6]), we study mechanisms with payments. We introduce a payment function $p_i : \Theta \to \mathbb{R}$ for each agent i to specify the amount that i must pay. As usual, the goal of mechanism design is then to find a mechanism $(g, p_1, \dots p_n)$ such that $g(\theta)$ returns an optimal outcome that maximizes the *social welfare*. The social welfare is defined as the total valuation of the agents on the outcome $\omega = g(\theta)$, i.e., $v(\omega, \theta) = \sum_{i \in N} v_i(\omega, \theta_i)$. The *utility* of an agent i on the outcome ω is defined by: $u_i(\omega, \theta_i) = v_i(e(\omega, \theta), \theta_i) - p_i(\theta)$. A rational agent tries to maximize this utility.

3 VCG revisited

We are interested in developing mechanisms which are able to prevent any type of misreporting. For this, we start by studying one of the most successful truthful mechanisms, and in some settings even the only one, called VCG [3, 7, 13].

Definition 6. *A* Vickrey-Clarke-Groves (VCG) *mechanism $(g, p_1, \dots, p_n)$ is composed of two elements: a social choice function g, and an n-tuple of payments $p_1, \dots, p_n$, where*

- *$g(\theta) \in \arg\max_{\omega \in F(\theta)} v(\omega, \theta)$, where $v(\omega, \theta) = \sum_i v_i(\omega, \theta_i)$, i.e., g maximizes social welfare;*
- *for any functions $h_1, \dots, h_n : \Theta^{n-1} \to \mathbb{R}$, for all types $\theta = (\theta_1, \dots, \theta_n)$, the payments are defined by $p_i(\theta) = h_i(\theta_{-i}) - \sum_{j \neq i} v_j(g(\theta), \theta)$.*

A basic VCG mechanism works as follows.

1. *The mechanism asks the agents to declare their types θ_i.*
2. *The mechanism finds an outcome using function g, and computes for each agent i its payment.*
3. *The mechanism informs the agents of the outcome $g(\theta)$, and asks each agent to pay the amount $p_i(\theta)$.*

A VCG mechanism is truthful with respect to traditional lying [9]. Moreover, when a VCG mechanism is used with an optimal algorithm, under-reporting can be easily prevented, because under-reporting has the same effect as declaring a cost of ∞ for the parts that an agent cannot or does not want to execute.

Proposition 1. *A VCG mechanism is truthful with respect to any combination of under-reporting and traditional lying.*

A basic VCG mechanism creates incentives for agents to declare their types truthfully, assuming that the selected outcome can really be obtained. However, when one or more agents are over-reporting, these same incentives turn out to be insufficient to guarantee truthfulness.

Lemma 1. *A VCG mechanism is not truthful with respect to over-reporting, and the gain by over-reporting can be arbitrarily large.*

Proof. A simple example suffices. Suppose that in a multiagent planning problem, agent 1 has a goal t_1 to achieve, which requires actions a_1 and a_2 with precedence relation $a_1 \prec a_2$ (i.e., a_1 has to be executed before a_2). In addition, once action a_1 has been done, agent 2's goal t_2 can be attained. Agent 1 itself is able to perform only action a_1, with cost 0. The reward for achieving goal t_1 is $\beta_1 > 0$ and of achieving goal t_2 is $\beta_2 > 0$.

Suppose that both agents truthfully declare their types. The outcome ω specifies executing action a_1 and thus t_2 can be achieved. Then $e(\omega, \theta) = \omega$, and the payment of agent 2 is: $p_2(\theta) = h_2(\theta_{-2})$, and thus its utility is:

$$u_2(\omega, \theta_2) = v_2(\omega, \theta_2) - p_2(\theta) = \beta_2 - h_2(\theta_{-2})$$

Suppose now that agent 2 over-reports that it is able to execute action a_2, with cost 0. With this declaration $\hat{\theta}$, the outcome $\hat{\omega}$ specifies executing a_1 and then a_2, and thus achieving both t_2 and t_1. The payment of agent 2 is then given by $p_2(\hat{\theta}) = h_2(\theta_{-2}) - \beta_1$. However, since action a_2 cannot be executed, we know that $e(\hat{\omega}, \theta) \neq \hat{\omega}$. In this example, the outcome that can really be achieved is ω, i.e., $e(\hat{\omega}, \theta) = \omega$. Agent 2's utility in this case thus is

$$u_2(e(\hat{\omega}, \hat{\theta}), \theta_2) = v_2(\omega, \theta_2) - p_2(\hat{\theta}) = \beta_2 - h_2(\theta_{-2}) + \beta_1.$$

So, we have $u_2(e(\hat{\omega}, \hat{\theta}), \theta_2) - u_2(\omega, \theta_2) = \beta_1 > 0$. Agent 2 can increase its utility by over-reporting. This example also shows that the gain by over-reporting can be arbitrarily large, depending on the reward of the goal (t_1) by the other agent.

Another property we would like a mechanism to have besides truthfulness, is that agents should not get a negative utility by participating. A mechanism is usually called *individually rational* (or IR) if agents never receive negative utility in equilibrium [8, 11]. The equilibrium strategies that we aim for in this paper are truthtelling. Hence, it is straightforward to see that if all agents are following this strategy, the VCG-mechanism is individually rational.

In this paper, however, we would like agents to also have a nonnegative utility guaranteed in case other agents are not truthful, specifically since part of the execution may then fail. We call mechanisms for which this holds *strongly individually rational*. In earlier work this same concept has been referred to as a *participation constraint* [9].

Definition 7. *A mechanism is* strongly individually rational *(sIR) if for every agent i, for every type profile $\hat{\theta} = (\theta, \hat{\theta}_{-i})$ where agent i is following the equilibrium strategy (i.e., i is truthful), the utility of agent i is non-negative, i.e., $u_i(e(g(\hat{\theta}), \theta_i)) \geq 0$.*

We now show that when an agent is over-reporting, the mechanism is not strongly individually rational anymore, because other agents may end up with a negative utility. We show this by the following multiagent planning problem called MAP(β) which can be defined for any value $\beta > 0$.

Example 2. This example is a generalization of Example 1. We consider a setting with two agents, where agent 1 has a goal t_1 which requires actions a_1 and a_2, and where these actions have a precedence relation $a_1 \prec a_2$. However, in this problem the reward for achieving goal t_1 is $\beta + 2$. Agent 1 itself is able to perform only action a_1, with cost β. Agent 2 claims that it can execute action a_2, with cost 1.

The type declaration of agent 1 is: θ_1, consisting of t_1 with reward $\beta + 2$ and action a_1 with cost β. The declaration of agent 2 is: θ_2, consisting of action a_2 with cost 1. Based on these declarations, any VCG mechanism computes an optimal outcome ω where goal t_1 can be achieved by letting agent 1 execute action a_1, and then asking agent 2 to do action a_2. The social welfare of this outcome is then 1. In addition, a VCG mechanism computes the payments of both agents as follows. The payment of agent 1 is $h_1(\theta_2) - (-1)$, i.e., agent 1 needs to pay 1 plus some function of θ_2, and the payment of agent 2 is $h_2(\theta_1) - (\beta + 2 - \beta)$, i.e, agent 2 receives 2 minus some function of θ_1.

Lemma 2. *A VCG mechanism is not strongly individually rational with respect to over-reporting.*

Proof. Suppose that in the example above, agent 2 actually over-reported its type—it is not able or not willing to do action a_2. In this case we end up in an outcome ω' where only action a_1 has been executed. The valuation $v_1(\omega', \theta_1)$ of this outcome for agent 1 is then based only on the cost of this action, i.e., $-\beta$. Since a VCG mechanism is used, the utility of agent 1 is then $-\beta - h_1(\theta_2) - 1$. Since h_1 depends only on θ_2 and not on β, no matter which function h_1 a VCG mechanism uses, we can construct an example by taking $\beta > -h_1(\theta_2) - 1$ where the utility for agent 1 is negative. Thus, the VCG mechanism is not individually rational with respect to over-reporting.

The following theorem immediately follows from the two lemmas above (Lemma 1 and Lemma 2).

Theorem 1. *A VCG mechanism is not strongly individually rational nor truthful with respect to over-reporting.*

Consequently, over-reporting can destroy truthfulness. This seems to be because the resulting outcome is infeasible. It is therefore interesting to know, for the purpose of detecting, whether there exist situations where an agent over-reports to improve its utility, but the mechanism chooses a *feasible* outcome. This turns out to be impossible in every situation where there is an incentive for the agent to do so.

Proposition 2. *Given a VCG mechanism and a type profile $\theta \in \Theta$, if an agent i has an incentive to over-report when other agents are truthful, then the outcome $g(\hat{\theta})$ (with $\hat{\theta} = (\hat{\theta}_i, \theta_{-i})$) is infeasible.*

Proof. We show this result by contraposition.

Let $\omega = \arg\max_{\omega \in F(\theta)} v(\omega, \theta)$ and let $\omega' = g(\hat{\theta})$. Assume ω' is a *feasible* outcome, i.e., ω' is also in $F(\theta)$. Hence, we have $e(\omega', \theta) = \omega'$. It also holds that $v_i(\omega', \theta_i) = v_i(\omega', \hat{\theta}_i)$, since agent i is over-reporting. The difference δ of the utility of agent i of the two outcomes ω and ω' is then given by

$$\begin{aligned}\delta &= u_i(\omega,\theta) - u_i(\omega',\hat{\theta}) \\ &= v_i(\omega,\theta_i) - p_i(\theta) - (v_i(\omega',\theta_i) - p_i(\hat{\theta})) \\ &= v(\omega,\theta) - h_i(\theta_{-i}) - \\ &\quad \Big(v(\omega',\hat{\theta}) - h_i(\theta_{-i}) + v_i(\omega',\theta_i) - v_i(\omega',\hat{\theta}_i)\Big) \\ &= v(\omega,\theta) - v(\omega',\hat{\theta}).\end{aligned}$$

Since ω is the best feasible outcome, $\delta \geq 0$. So, agent i will not gain more utility by over-reporting.

The consequence of this result is that if an agent over-reports, this can in principle always be detected, since the selected outcome is infeasible. This leads us to the main idea presented in the next section. If an over-reporting agent is detected, we can give this agent a penalty, and thus hopefully restore truthfulness.

4 Conditional payments

From the previous section we have seen that no VCG mechanism is truthful or individually rational with respect to over-reporting. One reason that an agent i may over-report is that i could receive some large payment by doing so. It seems that a straightforward solution to this problem is to hand over the payments to agents *after execution* like what Nisan and Ronen did in [9]. Unfortunately, this does not always solve the problem of the loss of individual rationality. Imagine in a MAP instance, the execution of the computed plan fails due to agent i's over-reporting. Even if i does not receive any payment from the mechanism after execution, some other agents may still lose money since they execute very costly actions but their goals are not achieved. If the mechanism is not able to compensate these agents, due to its limited budget, the agents will receive a negative utility.

Therefore, in this section we propose to extend the payment structure in the following two ways:

1. we ask for deposits from agents in advance, and then place a (high) penalty on agents that turn out to be lying by not returning their deposits, and
2. we compensate agents that have become a victim of such lying agents.

We keep the VCG payment in the extended payment structure in order to deal with situations where there are no over-reporting agents. In the following we present this extension by introducing two additional payment functions (dep and comp), but they can also be seen as part of just one (much more complicated) payment function.

Definition 8. *A conditional-VCG mechanism, defined by* $(g,(p_1,\ldots,p_n),(\mathsf{dep}_1,\ldots,\mathsf{dep}_n),(\mathsf{comp}_1,\ldots,\mathsf{comp}_n))$*, works as follows:*

1. *The mechanism asks the agents to declare their types* θ_i*.*
2. *The mechanism then asks each agent* i *to pay an amount (*$\mathsf{dep}_i : \Theta \rightarrow \mathbb{R}^+$*) as a deposit.*

3. *The mechanism finds an outcome using the function g taking into account only the agents who have paid the deposit.*
4. *Each agent i pays the amount according to the VCG payment p_i computed by the mechanism.*
5. *The mechanism informs the agents of the outcome $g(\theta)$, and each agent i executes its part.*
6. *If any part of the execution fails due to the agent i's declaration, agent i will not get its deposit back. All other agents are returned their deposits $\mathsf{dep}_j(\theta)$, as well as an amount as a compensation depending on the resulting outcome after failure ($\mathsf{comp}_j : \Omega \times \Theta \to \mathbb{R}^+$).*

If the execution is done successfully, we know there has not been any problem due to over-reporting, so the normal VCG payments apply. In this case, the compensations to all agents should be 0. The mechanism proposed here is similar to the mechanism for task scheduling with verification [9], except for a number of differences. For example, we do not assume that payments can always be enforced, but we have the deposit instead. The main difference, however, is that this mechanism is general for all kinds of problems. All results in this section hold for this general setting.

A conditional-VCG mechanism is called truthful if it is a dominant strategy for agents to declare their true types, no matter what other agents declare, and regardless of the state after the execution.

Definition 9. *A conditional-VCG mechanism $(g, p, \mathsf{dep}, \mathsf{comp})$ is* truthful *if for any agent i with type $\theta_i \in \Theta_i$, any true or mis-reporting type profile $\theta_{-i} \in \Theta_{-i}$, for all types $\hat{\theta}_i \neq \theta_i$, it holds that*

(1) if lying of i is detected, then

$$v_i(e(g(\theta), \theta), \theta_i) - p_i(\theta) + \mathsf{comp}_i(g(\theta), \theta)$$
$$\geq v_i(e(g(\hat{\theta}), \theta), \theta_i) - p_i(\hat{\theta}) - \mathsf{dep}_i(\hat{\theta})$$

(2) and otherwise

$$v_i(e(g(\theta), \theta), \theta_i) - p_i(\theta) + \mathsf{comp}_i(g(\theta), \theta)$$
$$\geq v_i(e(g(\hat{\theta}), \theta), \theta_i) - p_i(\hat{\theta}) + \mathsf{comp}_i(g(\hat{\theta}), \hat{\theta}).$$

In item (1) of the above definition, the left hand of the inequality specifies the utility of i when it is truthful, where the mechanism returns the deposit to i, together with some compensation in case other agents incur an infeasible outcome. The right hand of the inequality shows that when agent i's lying is detected, the deposit will not be returned to it. Moreover, it will not receive any compensation. In item (2), if agent i lies and its lying is not detected, the mechanism will return it its deposit, together with some compensation.

From Definition 9, it becomes clear that the incentives of agents relate to the payments, the deposits, the compensations, and moreover, whether or not the lying (over-, under-reporting, traditional forms of lying) agents can be caught. We now examine the influence of these parameters on truthfulness and individual rationality.

4.1 Truthfulness in dominant strategies

Since on successful execution the separate deposit stage does not enlarge the strategy space of the agents, it is straightforward to see that if the agents are truthful under the VCG mechanism, they will not be better off by lying under the conditional-VCG mechanism either.

Proposition 3. *A conditional-VCG mechanism is truthful in all settings where the VCG mechanism is truthful.*

Consequently, conditional-VCG is truthful with respect to traditional lying, under-reporting, and in addition, any combination. Separetely, we can show that it can prevent over-reporting as long as all over-reporting agents can be detected.

Proposition 4. *The conditional-VCG mechanism is truthful with respect to (only) over-reporting when all over-reporting agents can be detected.*

Proof. From Proposition 2, we know that agent i has an incentive to over-report only if the resulting outcome is infeasible. Suppose an infeasible outcome ω' is generated due to i's declaration $\hat{\theta}_i$. Agent i will then be caught by the mechanism during the execution stage, since all over-reporting agents can be detected. Given that a significantly large deposit was placed, agent i's utility when declaring $\hat{\theta}_i$ is not higher than that of being truthful (item (1) in Definition 9). Therefore, agent i is never worse off by truth-telling. This holds for any agent in the system.

In some problem domains this condition that all over-reporting agents can be detected can easily be satisfied. For instance, in a combinatorial exchange, all agents have to pass their promised resources simultaneously. Therefore, every over-reporting agent can be detected during the execution phase.

Proposition 4 also relies on the fact that the deposit dep_i that was placed is *big enough*. To ensure truthfulness, the deposit dep_i should be at least the difference between the utility of the *best* outcome that an agent i could have and the *worst* outcome that it may get. This way agent i can never be better off by losing the deposit, and thus it does not have an incentive to lie. We now show how to establish a lower bound on the deposit for a multiagent planning problem.

Example 3. Suppose that if every agent declares its true type, no goal can be attained. That is, the plan is empty. However, one agent i claims it can help to achieve the declared goals of all other agents G by using its actions O_i. The mechanism computes an (infeasible) outcome ω', where actions O_i are planned to achieve goals G. According to the declaration of agent i, the best outcome for i is ω' as agent i can receive a maximal utility of the total reward $r(G)$ of all goals G. In addition, the worst outcome for i is the empty plan with utility 0. Thus, the minimal deposit dep_i for agent i is $r(G)$.

So given such a large enough deposit, being able to detect all over-reporting agents is sufficient to prevent all agents from over-reporting. Unfortunately, we can show that if at the same time another agent i misreports, this cannot be detected. This result depends on the properties a truthful mechanism should have. In the remainder of this section we give an example problem where having all these properties lead to a contradiction. To arrive at this result, we first show that if a conditional-VCG mechanism is truthful, then the compensation to each agent must be independent of its declaration.

Proposition 5. *If a conditional-VCG mechanism is truthful, then the compensation* comp_i *to agent* i *does not depend on this agent's declaration* θ_i.

More specifically, given a type profile $\theta_{-i} \in \Theta_{-i}$, for all possible declarations $\hat{\theta}_i$ of agent i, if $g(\hat{\theta}_i, \theta_{-i}) = \omega$, then $\mathsf{comp}_i(e(\omega, \theta), \theta) = \mathsf{comp}_\omega$, where comp_ω is a fixed (positive) value.

Proof. Let θ_i and $\hat{\theta}_i$ be the true type and a lying type of agent i. Given θ_{-i}, denote $\theta = (\theta_i, \theta_{-i})$ and $\hat{\theta} = (\hat{\theta}_i, \theta_{-i})$.

We show this by contraposition. Suppose that the compensation depends on an agent's own declaration. Furthermore, let there be a setting where the two declarations of i result in a same outcome, i.e, $g(\theta) = g(\hat{\theta}) = \omega$. Since the payments p_i are computed based on VCG, it holds that given the same outcome ω, $p_i(\theta) = p_i(\hat{\theta}) = p_\omega$. Let ω' denote the outcome that is the result of a failed execution of outcome ω, i.e., $\omega' = e(\omega, \theta)$. Notice that the outcome ω' is same for the situation of declaring θ_i or $\hat{\theta}_i$.

Now assume that declaring θ_i and $\hat{\theta}_i$ have different compensations. Without loss of generality, let $\mathsf{comp}_i(\omega', \hat{\theta}) > \mathsf{comp}_i(\omega', \theta)$. Agent i is not caught by the mechanism as a lying agent, and thus its utilities when declaring θ_i and $\hat{\theta}_i$ are given by

$$u_i(\omega', \theta_i) = v_i(\omega', \theta_i) - p_\omega + \mathsf{comp}_i(\omega', \theta)$$

and

$$u_i(\omega', \hat{\theta}_i) = v_i(\omega', \theta_i) - p_\omega + \mathsf{comp}_i(\omega', \hat{\theta})$$

Therefore, it holds that

$$u_i(\omega', \hat{\theta}_i) - u_i(\omega', \theta_i) = \mathsf{comp}_i(\omega', \hat{\theta}) - \mathsf{comp}_i(\omega', \theta) > 0$$

So agent i with type θ_i can increase its utility by declaring $\hat{\theta}_i$, and the mechanism is thus not truthful.

Given this characterisation of the compensation function, the following proposition says that conditional-VCG cannot prevent under-reporting or traditional lying anymore when there are one or more over-reporting agents.

Proposition 6. *If there are one or more over-reporting agents, conditional-VCG cannot prevent other types of mis-reporting, even if all over-reporting agents can be detected.*

We use an example to present a such setting.

Proof. We consider two related multiagent planning problems with three agents. Agent 1 has two goals: (1) t_1 requires actions a_1 and a_2 and a_3, with a precedence relation $a_1 \prec a_2 \prec a_3$, (2) t'_1 requires actions a'_1 and a_3, with a precedence relation $a'_1 \prec a_3$. Agent 2 can execute a_2, and agent 3 can execute a_3 but only once. Therefore, given these two goals, there are two possible (non-empty) outcomes (plans):

– ω_1: achieve goal t'_1 by executing action a'_1 and then a_3; and
– ω_2: achieve goal t_1 by executing actions a_1, a_2, and then a_3.

Among these, the mechanism selects the outcome that has the highest social welfare, based on the declaration of agents.

Suppose that all over-reporting agents can be detected. In the first problem instance, the true and declared types of three agents are shown in Table 1. We see that agents 2 and 3 over-report non-existing actions a_2 and a_3. Thus both two outcomes ω_1 and ω_2 are infeasible. That is, if ω_1 is chosen, the execution will be stopped after agent 1 executes a'_1 due to the failure of agent 3 on executing a_3, and agent 3 is caught. If ω_2 is chosen, the execution will be stopped after agent 1 executes a_1 due to the failure of agent 2 on executing a_2. As a result, agent 2 is caught. Agent 1 will not be caught as a lying agent in both cases because it is able to execute the actions that it promised, i.e., no infeasible outcome generated by it. Consequently, it will receive some compensation. According to Proposition 8, we know that the compensation to agent i has to be a fixed value given θ_{-i} and a same outcome. Let the compensations for the outcomes $e(\omega_1, \theta)$ and $e(\omega_2, \theta)$ for agent 1 be comp_{ω_1} and comp_{ω_2}, respectively.

Given the declarations of agent 2 and 3 as shown in the table 1, we compare the utility of agent 1 when it is truthful (thus the outcome is ω_1) and when it is under-reporting (thus the outcome is ω_2):

$$\begin{aligned}
u_1(\omega_1, \theta_1) &= v_i(e(\omega_1, \theta), \theta_1) - p_i(\theta) + \mathsf{comp}_1(\omega_1, \theta) \\
&= -(\beta + 1) - (h_1(\theta_{-1}) - v(\omega_1, \theta) + v_1(\omega_1, \theta_1)) \\
&\quad + \mathsf{comp}_{\omega_1} \\
&= -(\beta + 1) - (h_1(\theta_{-1}) - 0) + \mathsf{comp}_{\omega_1} \\
&= -\beta - 1 - h_1(\theta_{-1}) + \mathsf{comp}_{\omega_1}
\end{aligned}$$

$$\begin{aligned}
u_1(\omega_2, \hat{\theta}_1) &= v_i(e(\omega_2, \theta), \theta_1) - p_i(\hat{\theta}) + \mathsf{comp}_1(\omega_2, \hat{\theta}) \\
&= -\beta - (h_1(\theta_{-1}) - v(\omega_2, \hat{\theta}) + v_1(\omega_2, \hat{\theta}_1)) \\
&\quad + \mathsf{comp}_{\omega_2} \\
&= -\beta - (h_1(\theta_{-1}) - 1) + \mathsf{comp}_{\omega_2} \\
&= -\beta - 1 - h_1(\theta_{-1}) + \mathsf{comp}_{\omega_2}
\end{aligned}$$

To ensure that agent 1 is truthful, we need to have $u_1(\omega_1, \theta_1) \geq u_1(\omega_2, \hat{\theta}_1)$. Thus it follows that $\mathsf{comp}_{\omega_1} \geq \mathsf{comp}_{\omega_2}$.

i	true type θ_i	declared type $\hat{\theta}_i$
1	a_1 with cost β, a'_1 with cost $\beta+1$, t_1 with reward $\beta+2$, t'_1 with reward $\beta+3$	a_1 with cost β, a'_1 with cost $\beta+1$, t_1 with reward $\beta+2$
2	$\emptyset$	a_2 with cost 1
3	$\emptyset$	a_3 with cost 0

Table 1. Case 1: When agent 1 under-reports by not declaring its goal t'_1, the resulting plan is ω_2, with a value of $\beta + 2 - 1 - \beta = 1$. If agent 1 declares truthfully, the resulting plan is ω_1 with a value of $\beta + 3 - \beta - 1 = 2$.

i	true type θ_i	declared type $\hat{\theta}_i$
1	a_1 with cost β, a'_1 with cost β, t_1 with reward $\beta + 2$, t'_1 with reward β	a_1 with cost β, a'_1 with cost β, t_1 with reward $\beta + 2$, t'_1 *with reward* $\beta + 2$
2	$\emptyset$	a_2 with cost 1
3	$\emptyset$	a_3 with cost 0

Table 2. Case 2: When agent 1 lies about the reward of its goal t'_1, the resulting plan is ω_1, with a value of $\beta + 2 - \beta = 2$. If agent 1 declares truthfully, the resulting plan is ω_2 with a value of $\beta + 2 - 1 - \beta = 1$.

Consider now another problem instance with (declared) types as given in Table 2. In this setting, we compare the utility of agent 1 when it is truthful (and the selected outcome is ω_2) to when it is under-reporting (and the outcome is ω_1):

$$\begin{aligned} u'_1(\omega_2, \theta_1) &= v_i(e(\omega_2, \theta), \theta_1) - p_i(\theta) + \mathsf{comp}_1(\omega_2, \theta) \\ &= -\beta - (h_1(\theta_{-1}) - 1) + \mathsf{comp}_{\omega_2} \\ &= -\beta - 1 - h_1(\theta_{-1}) + \mathsf{comp}_{\omega_2} \end{aligned}$$

$$\begin{aligned} u'_1(\omega_1, \hat{\theta}_1) &= v_i(e(\omega_1, \theta), \theta_1) - p_i(\hat{\theta}) + \mathsf{comp}_1(\omega_1, \hat{\theta}) \\ &= -\beta - (h_1(\theta_{-1}) - 0) + \mathsf{comp}_{\omega_1} \\ &= -\beta - h_1(\theta_{-1}) + \mathsf{comp}_{\omega_1} \end{aligned}$$

To ensure give agent 1 an incentive to be truthful in this setting, we need that $u'_1(\omega_2, \theta_1) \geq u'_1(\omega_1, \hat{\theta}_1)$. Thus it follows that $\mathsf{comp}_{\omega_2} \geq \mathsf{comp}_{\omega_1} + 1$. From Case 1, we have $\mathsf{comp}_{\omega_1} \geq \mathsf{comp}_{\omega_2}$. Clearly we cannot define comp such that conditional-VCG is truthful in both cases.

In other words, even if all over-reporting agents can be detected, conditional-VCG is not truthful. Only when *all* agents can be verified during the execution stage, conditional-VCG is truthful, because all lying agents will be caught by the mechanism. Thus a lying agent i will not get its deposit back from the mechanism. Given that a significantly large deposit was placed, agent i's utility when declaring $\hat{\theta}_i$ is not higher than that of being truthful. Therefore, agent i is never worse off by truth-telling. This holds for any agent in the system.

Lemma 3. *The conditional-VCG mechanism is truthful if and only if the execution of all agents can be verified.*

This result is the generalized version of Nisan and Ronen's result on task scheduling with verification [9]. In this domain agents' declarations are the execution times for certain jobs. Assuming that agents will not delay such executions on purpose, we can verify the types of all agents regarding the scheduled jobs during the execution of the schedule.

In the remainder of this paper we show that this mechanism is truthful in a Nash equilibrium without such a strong assumption.

4.2 Truthfulness in a Nash equilibrium

The condition in Lemma 3 that the execution of all agents can be verified is a strong assumption. Often in many multiagent applications, it is impossible to detect every single mis-reporting during execution, especially imagining the cases that in a single shot game, an agent may only participate once in the mechanism, and thus it is impossible to verify whether this agent under-reports its type.

Therefore, we wonder, without the assumption given in Lemma 3, whether it is possible for the conditional-VCG mechanism to obtain the truthfulness in a weaker notion, i.e., truthful in a Nash equilibrium. Notice that even this weaker notion does not hold for the original VCG mechanism.

Proposition 7. *A VCG mechanism is not truthful with respect to over-reporting even in a Nash equilibrium.*

The proof is simple, by observing the example in the proof of Lemma 1, where when agent 1 is truthful, agent 2 is still better off by over-reporting.

Fortunately, for the conditional-VCG mechanism, we can show that it is best for each agent to report its true type, provided that others are also doing so.

Lemma 4. *The conditional-VCG mechanism is truthful in a Nash equilibrium.*

Proof. Assume all other agents are reporting their true types. Suppose agent i has an incentive to *over-report*. This will thus cause an infeasible outcome (with Proposition 2). In the execution stage, agent i will be thus easily detected, and therefore lose its deposit. Assume this deposit is large enough, agent i will not receive higher utility by over-reporting.

Both *under-reporting* and *traditional lying* will also never lead to a higher utility with Proposition 1. When the mis-reporting of agent i combines over-reporting and other two lying types, an infeasible outcome will also be chosen (with Proposition 2). Thus agent i will be caught, and will not receive higher utility by doing so.

After these results on the truthfulness of conditional-VCG, we now study whether we can guarantee each truthful agent a nonnegative utility.

4.3 Strongly individual rationality

It is clear that when all agents are truth-telling, the mechanism is strongly individually rational, since the payments are computed by the VCG formula. However, when the execution is not completely successful, an agent's utility without compensation may become negative, even when itself has declared its true type. We have shown this in the proof of Lemma 2 in Section 3. Therefore, any agent who is not caught by the mechanism as a lying agent, should be given some compensation to guarantee its utility to be nonnegative. We first restate the definition of strong individual rationality (Definition 7) specifically for the conditional-VCG mechanism.

Definition 10. *A conditional VCG mechanism* $(g, p, \mathsf{dep}, \mathsf{comp})$ *is* strongly individually rational *(sIR) if for any agent i with true type $\theta_i \in \Theta_i$, for any $\theta_{-i} \in \Theta_{-i}$, and for any $\hat{\theta}_{-i} \neq \theta_{-i}$ it holds that*

$$v_i(e(g(\hat{\theta}), \theta), \theta_i) - p_i(\hat{\theta}) + \mathsf{comp}_i(g(\hat{\theta}), \hat{\theta}) \geq 0,$$

where $\hat{\theta} = (\theta_i, \hat{\theta}_{-i})$.

We now show the minimal compensation that the mechanism should give to the (truthful) agents in order to maintain strong individual rationality.

Proposition 8. *Given the declaration of the agents $\hat{\theta}$ and the outcome $\omega = g(\hat{\theta})$, the compensation* comp_i *that a strongly individually rational conditional-VCG mechanism needs to give to each agent i that is not lying is at least $p_i(\hat{\theta}) - v_i(e(\omega, \theta), \theta_i)$.*

Proof. Let ω' denote the outcome that is the result of a failed execution of outcome ω, i.e., $\omega' = e(\omega, \theta)$. An agent i that is not lying gets its deposit back from the mechanism, and its utility is then given by

$$u_i(\omega', \theta_i) = v_i(\omega', \theta_i) - p_i(\hat{\theta}) + \mathsf{comp}_i(\omega', \hat{\theta}).$$

It is not hard to see that in order for $u_i(\omega', \theta_i) \geq 0$, the minimal compensation the mechanism needs to pay to the agent depends on the payment that the agent received before execution, and its valuation on the partially executed outcome.

$$\mathsf{comp}_i(\omega', \hat{\theta}) \geq p_i(\hat{\theta}) - v_i(\omega', \theta_i).$$

We would like to guarantee a nonnegative utility for any truthful agent, even when other agents are not, and at the same time we would like to have a truthful mechanism.

Theorem 2. *The conditional-VCG mechanism with compensation function defined in Proposition 8 is both strongly individually rational and truthful if and only if the execution of all agents can be verified.*

Proof. From right to left we get truthfulness immediately from Lemma 3. For a truthful agent we can ensure that the compensation is always sufficient according to Proposition 8, so strongly individual rationality follows.

From left to right we prove by contraposition. Assume that there is a situation with an agent that is lying and the mechanism does not detect this.Then by Lemma 3 the mechanism cannot be truthful.

For the situations where it is impossible to verify all agents during execution, we present the following result.

Theorem 3. *The conditional-VCG mechanism with compensation function defined in Proposition 8 is truthful and strongly individual rational in a Nash equilibrium.*

Proof. We have shown in Lemma 4 that truth-telling is a Nash equilibrium strategy for agents. For a truthful agent, we can ensure that its utility is non-negative given the compensation defined in Proposition 8.

5 Related work

The failure of VCG to handle over-reporting has been mentioned before in the context of a task scheduling problem [9], a task allocation domain [4], and a multiagent planning problem [12]. Two different methods have been proposed to deal with this problem.

A deposit has been proposed in [12], and an extension of VCG called *mechanism by verification* has been proposed by Nisan and Ronen [9].

As we have already mentioned in Section 1, our work is similar to the latter approach [9], as we both address the need of handling the incentives not only in the declaration phase, but also in the actual execution phase. Nisan and Ronen [9] show that their mechanism for task scheduling is truthful, which they called *implementable with verification* if there exists a payment that ensures any agent is better off by reporting its type correctly. However, their mechanism solely depends on the assumption of *full verification*, that is, all declared information that is used in the outcome can be verified during execution, no matter whether the execution is successful or not. There is no discussion in their work on what properties a mechanism can have when such a strong assumption does not hold. It is also not clear whether it is possible to apply their mechanism to other problem domains other than task scheduling. In this paper we formalized the idea of a deposit from [12], generalized this to any problem domain, and extended it by having also a compensation. We proved that our mechanism is truthful and (strongly) individual rational in Nash without the requirement of being able to verify each agent. We just need to be able to detect the (over-reporting) agent that has caused the unsuccessfully termination. We believe that this mechanism can thus be applied to a broader set of problem domains.

Auletta et al. [1, 2] are also interested in *characterizing* social choice functions which are implementable with verification. They show that the famous weak monotonicity condition [8] is not always a sufficient condition of truthfulness for a mechanism with verification. The mechanism with verification studied in [2] is not able to detect every form of misreporting. For instance, in the task scheduling problem, an agent may report to be slower than it actually is. They do not consider such types of misreporting, which have lower valuations than truth-telling. In this paper we have seen that a combination of different types of mis-reporting can lead to problems, even when the mechanism is truthful with respect to any one of them separately. It would be interesting to see if such a combination could make any difference for their results.

6 Conclusion and Discussion

When the agents in a multiagent system are self-interested and rational they may try to manipulate the outcome of the system in their favor. Most work in mechanism design assumes that agents can only lie about the valuation they have for certain outcomes, but in real systems agents may also lie about their capabilities or refuse to execute their part to reach a certain outcome. We have proved that VCG is not truthful (not even in Nash) nor strongly individual rational in such domains where the execution of a selected outcome can fail.

We then presented an extension of VCG, conditional-VCG, that can deal with such settings. This (direct) mechanism has a conditional payment after an outcome has been achieved. If this outcome turns out to be different from the outcome selected by the mechanism, at least one of the agents failed to execute its part of the outcome. We showed that this mechanism has the following properties.

1. If and only if the execution of all agents can be verified, conditional-VCG is incentive compatible in dominant strategies and strongly individually rational.
2. When at most one failure on the execution can be detected, conditional-VCG is incentive compatible in Nash and strongly individually rational.

These results indicate a partitioning of mechanism design problems into three categories.

– When there is no possibility of a failed execution due to rational misbehavior of agents, traditional VCG mechanisms can be used.
– When execution can fail due to misreporting agents, or agents that refuse to execute part of an outcome, and when all such misbehaving agents can be detected during execution, conditional-VCG can be used. It guarantees a dominant strategy and a nonnegative utility for all other truthful agents. This applies for example to the task scheduling domain with verification.
– Finally, when execution may fail and at least one such misbehaving agent can be detected, conditional-VCG guarantees a nonnegative utility for all truthtelling agents, and is incentive compatible in a Nash equilibrium. This is for example the case in a multiagent planning domain, but also in other domains with dependencies of one agent upon another in execution, like scheduling with precedences.

This categorization gives an indication of what kind of results can be expected in a given domain.

To continue this line of work, we would like to study (i) the influence of using non-optimal algorithms on the properties of the proposed conditional-VCG mechanism [10], and (ii) to see whether the weak monotonicity condition [8] together with a condition like monitoring is sufficient for truthfulness.

Another interesting topic is to find bounds on the depoit. In Example 3 we gave a lower bound on the deposit to ensure truthfulness for MAP, but suppose that we also want the collected deposit to be sufficient to cover all compensations to the other (truthful) agents. Is it posible then to define a deposit function such that deposit-VCG is both (weakly) budget balance and individually rational?

Finally, we would like to study settings with infeasible outcomes in online mechanisms. We conjecture that a payment in each step of the execution is sufficient to allow for a mechanism that is incentive compatible in dominant strategies, instead of one that is incentive compatible in Nash as holds for the one-shot mechanisms in this paper.

References

1. V. Auletta, R. D. Prisco, P. Penna, and G. Persiano. The power of verification for one-parameter agents. In J. Díaz, J. Karhumäki, A. Lepistö, and D. Sannella, editors, *Proc. of 31st Int. Colloquium on Automata, Languages and Programming (ICALP 2004)*, volume 3142 of *LNCS*, pages 171–182. Springer, 2004.
2. V. Auletta, R. D. Prisco, P. Penna, P. Persiano, and C. Ventre. New constructions of mechanisms with verification. In *Proc. of 33rd Int. Colloquium on Automata, Languages and Programming (ICALP 2006)*, volume 4051 of *LNCS*, pages 507–608, 2006.
3. E. H. Clarke. Multipart pricing of public goods. *Public Choice*, 11(1), 1971.

4. M. M. de Weerdt and Y. Zhang. Preventing under-reporting in social task allocation. In H. L. Poutre and O. Shehory, editors, *Proceedings of the 10th workshop on Agent-Mediated Electronic Commerce (AMEC-X)*. IFAAMAS, 2008.
5. M. Ghallab, D. Nau, and P. Traverso. *Automated Planning, theory and practice*. Morgan Kaufmann Publishers, 2004.
6. A. F. Gibbard. Manipulation of voting schemes: A general result. *Econometrica*, 41(4):587–601, 1973.
7. T. Groves. Incentives in teams. *Econometrica*, 41(4):617–31, 1973.
8. N. Nisan. Introduction to mechanism design (for computer scientists). In *Algorithmic Game Theory*, pages 209–242. Cambridge University Press, 2007.
9. N. Nisan and A. Ronen. Algorithmic mechanism design. *Games and Economic Behavior*, 35(1–2):166–196, 2001.
10. N. Nisan and A. Ronen. Computationally feasible VCG mechanisms. *Journal of AI Research*, 29:19–47, 2007.
11. Y. Shoham and K. Leyton-Brown. *Multiagent Systems: Algorithmic, Game-Theoretic and Logical Foundations*. Cambridge University Press, 2008.
12. R. van der Krogt, M. de Weerdt, and Y. Zhang. Of mechanism design and multiagent planning. In M. Ghallab, C. D. Spyropoulos, N. Fakotakis, and N. Avouris, editors, *Proceedings of the 18th European Conference on Artificial Intelligence (ECAI-08)*, pages 423–427. IOS Press, 2008.
13. W. Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *The Journal of Finance*, 16(1):8–37, 1961.