Mechanisms for Hostile Agents with Capacity Constraints

Prashanth L A†, H L Prasad♯, Nirmit Desai$, and Shalabh Bhatnagar♯

†SequeL Team, INRIA Lille - Nord Europe, FRANCE, prashanth.la@inria.fr
♯Indian Institute of Science, INDIA, {hlprasu,shalabh}@csa.iisc.ernet.in
$IBM Research, INDIA, nirmit.desai@in.ibm.com

ABSTRACT

Several key economic scenarios involve agents having limited capacities whose types change with time, e.g., service workers attending to service requests, power plants supplying to power grids, and machines connected to computing grids. Dynamic mechanisms have been proposed to address the issue of dynamic types. Also, a few mechanisms have been proposed to account for limited capacities in static settings. However, no prior work considers *hostile* agents having a preference for harming other competing agents by making capacity over-reports. This paper proposes two novel mechanisms that possess desired properties even when the agents are hostile. First, we extend a static mechanism with capacity constraints with (1) a novel utility function that captures the preference to harm others and (2) a marginal compensation penalty scheme that minimizes the cost of capacity misreports. Next, we extend such a mechanism to the case where both the unit cost and the capacity elements of agent types are dynamic. We show that both of our mechanisms are ex-post incentive compatible, ex-post individually rational, and socially efficient.

Categories and Subject Descriptors

J.4 [**Social and Behavioral Sciences**]: Economics; I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence- Multiagent systems

General Terms

Economics, Design, Algorithms

Keywords

mechanism design, multi-agent systems, VCG, social welfare

1. INTRODUCTION

Mechanism design has been a key technique for decision making in various economic scenarios. In most real economic scenarios, agents' have limited capacities, for instance, the supply capacity of a power plant on a given day. Also, in many of the economic scenarios of interest, both the cost and the capacity of the agents' change with time, e.g., the cost and capacity of producing a good. Most importantly, economic scenarios can be fiercely competitive where one agent's loss is another's gain. We call such agents *hostile* because they may be willing to over-report capacities and dent the allocations to their competitors, even though the over-report attracts a fixed penalty in future.

In this paper, we consider hostile agents in both the static setting as well as a dynamic setting in which the private types evolve over time according to a Markov process. The hostility is captured via a novel component in the utility function that increases linearly with the allocation even at the cost of allocation to others. We first propose an enhanced static mechanism that adapts the transfer scheme of [6] to deal with over-reported capacity. The novelty here is a variable penalty scheme that penalizes an agent to the extent of damage his misreport caused in the allocation caused to the other agents. Next, we extend this idea for the domain that lies at the intersection of the two scenarios – agents having limited capacities and unit costs, both of which change with time.

We consider a procurement setting in which a single buyer attempts to obtain D units of a good from a set of n sellers. Each seller i is capacity-constrained, in the sense that they can produce up to c_i units at a constant unit price u_i. The capacity and unit price are private information. While a simple VCG-like transfer scheme ensures truthful unit price reports from the individual agents, the same is not true w.r.t. capacity type element (See Example 1).

Several mechanisms have been proposed to deal with the case of limited capacity of agents in the static case, but we focus on the work of [6]. In their work, the limited capacity of agents is accommodated by ensuring that an agent is never allocated more than its reported capacity and by imposing a fixed penalty of a positive δ to ensure truthtelling w.r.t. the capacity type element. Here, if an agent misreports its capacity, produces less units than allocated, and is penalized as above, the agent would derive utility of $-\delta$. In a static setting, such a fixed penalty scheme is shown to be strategyproof, i.e., truthful type reports is a dominant strategy for each of the agents, see [6].

In spite of the fact that a fixed δ-penalty achieves strategyproofness as shown by [6], in practical competitive scenarios, agents could be willing to incur a loss of δ, if it is relatively small compared to the major losses inflicted to other agents due to the over-report. We incorporate the preference of agent to harm other agents via capacity misreport through an additive term in the utility function that captures the agent's allocation under the reported capacity. Note that the reported capacity of an agent may be overstated, resulting in an increased allocation at the cost of other agents. In this modified utility setting, as we demonstrate via examples later, the mechanism of [6] does not ensure truthful capacity reports. In other words, a fixed δ-penalty does not deter capacity misreports by an agent because the degree of losses inflicted on other agents via a capacity misreport is a variable, which depends on the allocation and hence the individual agents' reported types.

Appears in: *Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2013), Ito, Jonker, Gini, and Shehory (eds.), May, 6–10, 2013, Saint Paul, Minnesota, USA.*

To address this issue, we present a static mechanism $\mathcal{MC}$ that enhances the mechanism from [6], to incorporate a novel variable penalty scheme. The penalty imposed on agents misreporting their capacities is the same as the loss of allocation to other agents due to misreported capacities. We formally show that $\mathcal{MC}$ is strategyproof. In particular, we show that $\mathcal{MC}$ ensures truthful capacity reports from the individual agents in a setting where the agents have a preference to overstate capacity and hence, negatively impact the allocation of other agents.

Dynamic mechanism design is a key area, especially when the center makes allocations to participating agents repeatedly. For example, service workers employed by a service provider would typically attend to thousands of service requests during their employment period. Each service worker has a limited amount of time (which keeps changing every day) and her estimate of the amount of time required to resolve service requests also changes with skill gain or decay. For an efficient allocation, i.e., for minimizing the total time required to resolve a set of service requests, it is not enough for the provider to consider the current types of service workers but also the future horizon. Significant results have been already established in this area [2, 3, 5]. We focus on the work of Bergemann and Valimaki (B&V) [2], that proposes a dynamic pivot mechanism that is truthful, efficient, and individually rational. The optimal allocation is computed as the expectation on the discounted sum of valuations of each agent with his payoff computed based on his marginal contribution. However, B&V's mechanism considers a setting where a common resource is allocated to an agent at every time step and the agent's capacities are not considered. Hence the question of capacity misreports and the consequent need to penalize misreporting agents does not arise in their work.

Unfortunately, capacity constraints are nontrivial to accommodate in the design of truthful and socially efficient dynamic mechanisms. The main challenge is that in a given time step, the payoff to an agent cannot be determined until the time that the actual units produced by an agent is known. If the allocation π to agent i to produce π_i units is made at time t_1 based on the expected discounted sum of current and future valuations of all agents, and the actual number of units $\bar{c}_i$ produced by agent i is known at a later time $t_2 > t_1$, then the payoff to agent i for the allocation π cannot be made prior to t_2 because it depends on $\bar{c}_i$. However, because the valuations are discounted in the future, the marginal contribution computed at t_1 may not yield allocative efficiency at t_2. This is because, intuitively, a payment made at t_1 is worth more than the same payment at t_2. The question then is what should the payoff to agent i be at time t_2 for allocation π such that the resulting mechanism is truthful in terms of unit costs as well as capacities and is socially efficient? Note that this question does not arise for B&V's work because the payoffs can be made at t_1, i.e., the time of allocations.

To answer the above question, we develop a delayed transfer scheme that consists of a marginal contribution component and a penalty component. The marginal contribution component is similar to the transfer scheme in the dynamic pivot mechanism [2]. However, we establish via a counterexample that providing the marginal contribution alone is not sufficient to ensure incentive compatibility of the mechanism. For this purpose, a penalty component is necessary in the transfer to the agents. With this motivation, we propose $\mathcal{DMC}$, the first dynamic mechanism to tackle capacity constraints. While $\mathcal{DMC}$ is an extension of $\mathcal{MC}$ for the dynamic case, a key novelty of $\mathcal{DMC}$ is the delayed transfer scheme where payoffs are adjusted based on the period elapsed since allocation. We establish that $\mathcal{DMC}$ is allocatively efficient, within period ex-post incentive compatible, and individually rational.

1.1 Literature review

For a survey of the core results to date in the area of dynamic mechanism design, the reader is referred to [4]. The authors in [5] present a dynamic mechanism without capacity constraints wherein agents become unavailable for a period after the allocation is made and the payoffs can only happen when the agents eventually become available. The payoffs are adjusted upwards by compounding the discount factor for all the time points during which the agent is unavailable. In [1], the authors obtain a Bayes-Nash incentive compatible, efficient and budget-balanced mechanism for a persistent-population, dynamic-type environment with private values and independent type transitions. While the works outlined above consider interesting and realistic extensions to dynamic mechanism design, they do not consider capacity constraints and hence are complementary to our work.

The rest of the paper is organized as follows: In Section 2, we describe in detail the static setting and present a mechanism ($\mathcal{MC}$) that incorporates a variable penalty scheme. In Section 3, we present the dynamic mechanism framework with capacity constraints and develop an incentive compatible mechanism ($\mathcal{DMC}$) in this framework. Finally, in Section 4 we provide the concluding remarks.

2. STATIC MECHANISM WITH CONSTRAINTS ($\mathcal{MC}$)

2.1 The Setting

Consider a setting involving N agents (numbered $1, 2, \ldots, N$) that are capable of performing certain manufacturing tasks. Note that the results of this paper would apply to any domain where the agents are allocated some work for the external customers. A central controller receives a task of manufacturing D items of a certain commodity. Each agent i is characterized by his type information - a tuple (u_i, c_i), where u_i is the unit price and c_i is the maximum capacity that agent i can provide. We assume here that the collective capacity of the agents exceeds the demand D and further, the same holds true even in the absence of an agent i, where $i = 1, \ldots, N$. The type vector θ is given by $\theta = (\theta_1, \ldots, \theta_N)$, where $\theta_i = (u_i, c_i)$. We let $\theta_i \in \Theta_i$, where Θ_i denotes the type space of agent i. The joint type Θ is then given by $\Theta = \Theta_1 \times \ldots \times \Theta_N$. The problem at the central controller is to obtain an efficient allocation such that the demand is met and the total cost is minimized as well. We denote an allocation by $y = (y_1, \ldots, y_N)$, where y_i is the number of items that have to be produced by agent i. The central controller thus solves the following optimization problem:

$$\begin{aligned}
&\text{Find } \pi(\theta) = \underset{y \in \mathcal{Y}}{\operatorname{argmin}} \sum_{j=1}^{N} u_j y_j \\
&\text{s.t. } 0 \leq y_j \leq c_j, j = 1, 2, \ldots, N, \\
&\text{and } \sum_{j=1}^{N} y_j = D.
\end{aligned} \tag{1}$$

In the above, $\mathcal{Y}$ is the set of all possible allocations. The policy $\pi(\theta)$ obtained as a solution to (1) is called socially efficient and we use π_i to denote the allocation to agent i under the efficient policy π. For the purpose of defining payments to agent i in the marginal sense, we also require the solution to (1) with agent i removed, i.e., a solution to the following optimization problem:

$$\begin{aligned}
&\text{Find } \pi_{-i}(\theta) = \underset{y \in \mathcal{Y}}{\operatorname{argmin}} \sum_{j \neq i} u_j y_j \\
&\text{s.t. } 0 \leq y_j \leq c_j, j \neq i, \\
&\text{and } \sum_{j \neq i} y_j = D.
\end{aligned} \tag{2}$$

The allocation to agent j under the policy obtained as a solution to (2) is denoted $\pi_{-i,j}$. The problem now is to define a mechanism that results in truthful type reports by the individual agents. This will happen if the agent's utility function is maximized under true reports, which constitutes *strategyproofness*. Before formalizing our mechanism, we define the utility function U_i of an agent i in our setting.

DEFINITION 1. *The utility function U_i of agent i is* quasi-linear *and is given by*

$$U_i(\pi, t_i, \hat{\theta}) = t_i - u_i \bar{c}_i(\hat{\theta}) + \pi_i((u_i, \hat{c}_i), \hat{\theta}_{-i}), \tag{3}$$

In the above, $\bar{c}_i(\hat{\theta})$ denotes the achieved capacity of agent i, given that its allocation was $\pi_i(\hat{\theta})$. Note that if the agent has overstated its capacity, then the achieved capacity can in fact be lesser than the allocation. The second term in (3) *signifies the cost to agent i under the allocation policy π and t_i is the payment that agent i receives from the central controller. The rationale behind the final term $\pi_i(\cdot)$ above is that the utility an agent derives increases linearly with its allocation, even if it is at the cost of other agents. This is evident by noting that $\pi_i(\theta) = D - \sum_{j \neq i} \pi_j(\theta)$. Thus, the final term captures the preference of an agent to ruin others by overstating capacity. This is unlike the utility function formulation in [6], where the utility function contained the first two terms of* (3).

REMARK 1. *One could also use a reward function $f(\cdot)$ that converts from allocation units to money or reward points in the utility function* (3) *as follows:*

$$U_i(\pi, t_i, \hat{\theta}) = t_i - u_i \bar{c}_i(\hat{\theta}) + f(\pi_i((u_i, \hat{c}_i), \hat{\theta}_{-i})), \tag{4}$$

The exact details of the reward function $f(\cdot)$ is immaterial for the discussion here, as long as it is ensured to be monotonic in its parameter. Further, we do not incorporate the reward function in the material next and the analysis can be seen to be trivially extended to include $f(\cdot)$.

Formally, a mechanism is a tuple (π, t), where π is the policy component that maps the types to an allocation vector and t denotes the transfer scheme, with $t_i : \Theta \to \mathcal{R}$ denoting the monetary payment received by agent i. The policy employed in our mechanism is socially efficient (as defined by (1)). Further, the transfer scheme t ensures strategyproofness and individual rationality - properties desirable of any mechanism. We later prove these properties formally.

DEFINITION 2. *A mechanism is* strategyproof *if truth-telling constitutes a dominant strategy for the agents, i.e., for all agents $i = 1, 2, \ldots, N$,*

$$U_i(\pi, t_i(\theta_i, \hat{\theta}_{-i}), (\theta_i, \hat{\theta}_{-i})) \geq U_i(\pi, t_i(\hat{\theta}), \hat{\theta}), \quad \forall \hat{\theta} \in \Theta$$

where $\hat{\theta}$ denotes the reported types of all agents and $\hat{\theta}_{-i}$ the reported types without agent i.

DEFINITION 3. *A* strategyproof *mechanism is* individually rational *if the agents have an incentive to participate than leave it i.e., for all agents $i = 1, 2, \ldots, n$, for all types $\theta \in \Theta$*

$$U_i(\pi, t_i(\theta), \theta) \geq 0.$$

We assume that the utility an agent derives by leaving the mechanism is 0.

An important aspect of the mechanism $\mathcal{MC}$ worth noting here is that on completion of a production task by agent i, the allocation is re-computed with achieved capacity of agent i being used instead of the reported capacity. The policy that results from this allocation is used to decide the transfer for agent i. Table 1 summarizes the notation used for the policies obtained by solving (1) under different type vector arguments. Note that the allocation to individual agents is obtained via subscripting. For instance, $\pi_j(\bar{\theta}_i, \hat{\theta}_{-i})$ denotes the allocation to agent j when (1) is solved with the type argument $(\bar{\theta}_i, \hat{\theta}_{-i})$.

2.2 Motivation

Before formalizing the various aspects of $\mathcal{MC}$, we present a motivating example that considers a limited capacity setting and shows that the popular VCG mechanism's transfer scheme as well as the transfer scheme featuring a fixed δ-penalty proposed by [6] fail to ensure truthful capacity reports. However, note that eliciting truthful unit price reports is not a problem and the VCG mechanism's transfer scheme ensures this part. It is the capacity constraints that make the problem non-trivial. In all the example scenarios through the paper, the demand D is assumed to be 150.

EXAMPLE 1. *Consider a scenario where there are three agents with their types given by $(u_1, c_1) = (1, 100)$, $(u_2, c_2) = (2, 50)$ and $(u_3, c_3) = (3, 130)$. Suppose agent 1 misreports his capacity to be 125 while reporting his unit price truly. The other agents report their true types. Now, the efficient allocation with reported types is $(125, 25, 0)$. However, after the agents complete their task, the achieved capacities would be $(100, 25, 0)$. A VCG-like payment scheme in this setting would be according to:*

$$t_i = \sum_{j \neq i} \widehat{u}_j \pi_{-i,j}(\hat{\theta}_{-i}) - \sum_{j \neq i} \widehat{u}_j \pi_j(\hat{\theta}). \tag{5}$$

As per the above payment rule, agent 1's payoff would be $t_1 = (2 \times 50 + 3 \times 100) - (2 \times 25) = 350$. On the other hand, if agent 1 reported his capacity truthfully, he would have received as transfer $t_1 = (2 \times 50 + 3 \times 100) - (2 \times 50) = 300$. Thus, it is evident that agent 1 has an incentive to misreport his capacity and receive a higher transfer at the cost of other agents.

To handle capacity constraints, the mechanism of [6] incorporated a fixed δ-penalty based transfer scheme, where the payment t_i to an agent i is according to:

$$t_i = \sum_{j \neq i} \widehat{u}_j \pi_{-i,j}(\hat{\theta}_{-i}) - \sum_{j \neq i} \widehat{u}_j \pi_j(\bar{\theta}_i, \hat{\theta}_{-i}) - \delta \beta_i. \tag{6}$$

In the above, β_i is a binary variable which is equal to 1 if $\bar{c}_i < \pi_i(\hat{\theta})$, i.e., when the agent has over-stated his capacity, and 0 otherwise. The first two terms in (6) *refer to the marginal contribution of agent i. This is calculated using the achieved capacity of agent i, denoted by $\bar{c}_i$.*

Thus, as per (6)*, agent 1's payoff would be $t_1 = (2 \times 50 + 3 \times 100) - (2 \times 50) - \delta = 300 - \delta$. On the other hand, if agent 1 reported his capacity truthfully, he would have received as transfer $t_1 = (2 \times 50 + 3 \times 100) - (2 \times 50) = 300$.*

While the mechanism proposed in [6] achieved strategyproofness, the utility function of agent i in their setting was $U_i(\pi, t_i, \theta) = t_i - u_i \bar{c}_i(\theta)$. However, in our setting with the utility function as defined in (3), strategyproofness isn't necessarily ensured using the transfer of [6]. This is because, an agent i has an incentive to misreport his capacity and improve his allocation at the cost of others (see the last term in (3)). To make this precise, the utility U_1 of agent 1 in our setting, i.e., with U_i given by (3), turns out to be $(300 - \delta - 1 \times 100 + 125) = 325 - \delta$, whereas with true capacity report it is $(300 - 1 \times 100 + 100) = 300$. Thus, it is clear that truthtelling

Table 1: Summary of the various allocation policies used in $\mathcal{MC}$

Notation	Description	Input type
$\pi(\hat{\theta})$	Efficient allocation with reported types	$\hat{\theta} = (\hat{\theta}_1, \hat{\theta}_2, \ldots, \hat{\theta}_N)$, where $\hat{\theta}_i = (\hat{u}_i, \hat{c}_i)$
$\pi(\bar{\theta}_i, \hat{\theta}_{-i})$	Efficient allocation with achieved type of agent i and reported types of all other agents	$(\bar{\theta}_i, \hat{\theta}_{-i})$, where $\bar{\theta}_i = (\hat{u}_i, \bar{c}_i)$

w.r.t. the capacity element does not guarantee a higher utility for all values of δ.

The choice of utility function in our setting is realistic because in practical competitive scenarios, agents could choose to inflict major losses to other agents. The mechanism of [6] penalized a misreporting agent by a fixed δ, which in our setting may not be enough to ensure truthful capacity reports. This is because, an agent may be willing to incur a loss of δ, if it is relatively small compared to the major losses inflicted to other agents due to the misreport. This is evident in the above example where agent 1 may choose to misreport his capacity and significantly impact the allocation of agent 2 (reduced by 25 units) while incurring a penalty of a potentially relatively small δ. On the other hand, a large value of δ also is not advisable. Consider again the above example and assume that $\delta = 1000$. It can be the case that agent 1 was probably capable of producing 125 units but fell short due to some unforeseen issues that he faced during production. A large δ may not be appropriate in this scenario. In general, precisely because the degree of losses to others that can be caused by capacity misreports in a given market are unknown, it is difficult to determine a fixed δ that effectively deters capacity misreports. Instead, it is necessary to devise a transfer scheme that penalizes the agent with the extent of damage (in the marginal sense) his misreport caused. $\mathcal{MC}$ achieves this by having a novel variable penalty scheme, that we present below.

REMARK 2. *As we observed in the above example, an agent has an incentive to overstate its capacity, while it has no incentive to under-state its capacity. Suppose an agent under-reports its capacity and this influences the optimal allocation* (8). *This would imply that the optimal allocation* (8) *would be performed with tighter constraints and hence the marginal contribution of the agent is reduced, implying a reduced utility to the agent. In a similar fashion, it can be argued that an agent has no incentive to over-report its unit price.*

2.3 The Mechanism

Given the above problem description, we now define our centralized mechanism. Our mechanism builds on the one proposed by [6], while incorporating a novel variable penalty scheme. The penalty scheme proposed here enables our mechanism to ensure zero loss to other agents when one or more agents misreport their capacities. Our mechanism is direct where each agent reports his type information and the central controller then calculates an efficient allocation. The transfer, however, happens with a delay and is made only after the agent completes the allotted production task.

Transfer Scheme

The transfer t_i that an agent i receives is a sum of two components and is made when agent i's production task is completed. The first component, denoted x_i, is the marginal contribution of agent i to the system, while the second component is a penalty, denoted p_i, imposed on agents who misreport their capacities to influence the efficient allocation. The transfer components are given by:

$$\begin{aligned} t_i &= x_i + p_i, \\ \text{where} \\ x_i &= \sum_{j \neq i} \widehat{u}_j \pi_{-i,j}(\hat{\theta}_{-i}) - \sum_{j \neq i} \widehat{u}_j \pi_j(\bar{\theta}_i, \hat{\theta}_{-i}) \\ p_i &= \sum_{j \neq i} \pi_j(\hat{\theta}) - \sum_{j \neq i} \pi_j(\bar{\theta}_i, \hat{\theta}_{-i}). \end{aligned} \quad (7)$$

The first component x_i of the payment t_i to the agent i is his marginal contribution towards reducing the total production cost, i.e., the difference between total costs of allocation that other agents obtain without and with agent i, respectively. The penalty component p_i is also marginal in the sense that it is the difference between the total costs of allocation that other agents obtain with agent i's reported capacity and achieved capacity, respectively. Hence, the loss caused by agent i's misreport is compensated via p_i.

REMARK 3. *Considering that* $\sum_{j=1}^{N} \pi_j = D$*, the penalty component can be seen as equivalent to* $p_i = \pi_i(\bar{\theta}_i, \hat{\theta}_{-i}) - \pi_i(\hat{\theta})$.

REMARK 4. *The reward function* $f(\cdot)$ *mentioned previously, can be incorporated into the penalty component as follows:*

$$p_i = \sum_{j \neq i} f(\pi_j(\hat{\theta})) - \sum_{j \neq i} f(\pi_j(\bar{\theta}_i, \hat{\theta}_{-i})).$$

2.4 Discussion

I) Now we demonstrate the usefulness of our penalty scheme using example 1, which involved a single agent misreporting his capacity. In example 1, $\pi_{-1}(\hat{\theta}_{-1}) = (50, 100)$ and $\pi(\bar{\theta}_1, \hat{\theta}_{-1}) = (100, 50, 0)$. $\mathcal{MC}$ would result in a payment of $x_1 = (2 \times 50 + 3 \times 100) - (2 \times 50) = 300$. to agent 1 and the penalty component would be $p_1 = 25 - 50 = -25$. Note that while the marginal contribution component with $\mathcal{MC}$ is the same as for [6], the penalty is variable and calculated based on the damage caused by agent 1's misreport, unlike [6] where it was a fixed quantity δ. As explained before, the utility U_1 derived by agent 1 under true capacity report is 300, while under capacity misreport, $U_1 = 325 - \delta$ in the case of [6]. On the other hand, U_1 in the corresponding case for $\mathcal{MC}$ turns out to be $(300 - 25) - 1 \times 100 + 125 = 300$, which is equal to the utility derived under true report.

II) We now provide an example to illustrate that truth-telling in unit price is a weakly dominant strategy in $\mathcal{MC}$. Suppose there are three agents with $(u_1, c_1) = (3, 125)$, $(u_2, c_2) = (2, 50)$ and $(u_3, c_3) = (4, 100)$. Assume that the agents report their true capacities. With a unit price report of $(\widehat{u}_1, \widehat{u}_2, \widehat{u}_3) = (1, 2, 4)$, i.e., with agent 1 misreporting his unit price, the optimal allocation is $\pi(\hat{\theta}) = (125, 25, 0)$. In this case, agent 1's payment is $(2 \times 50 + 4 \times 100) - 2 \times 25 = 450$, whereas with the true type reports, it is $(2 \times 50 + 4 \times 100) - 2 \times 50 = 400$. Hence, the utility derived by agent 1 in the former case (with unit price misreport) is

$450 - 3 \times 125 + 100 = 175$, whereas with true type report, agent 1's utility is $400 - 3 \times 100 + 100 = 200$. This illustrates that $\mathcal{MC}$ forces truth telling by agent 1 in the unit price component and hence is dominant.

III) We now provide an example where both unit price and capacity elements are misreported. Suppose there are three agents with $(u_1, c_1) = (3, 75)$, $(u_2, c_2) = (2, 50)$ and $(u_3, c_3) = (4, 100)$. Agent 1 misreports its type as $(1, 125)$, whereas the other agents report their true types. The optimal allocation $\pi(\hat{\theta}) = (125, 25, 0)$ and $\pi(\bar{\theta}_1, \theta_{-1}) = (75, 50, 25)$. The transfer components x_1 and p_1 to agent 1 can be calculated as follows:

$$x_1 = (2 \times 50 + 4 \times 100) - (2 \times 50 + 4 \times 25) = 300,$$
$$p_1 = 75 - 125 = -50.$$

Thus, the utility of agent 1 is $(300 - 50) - 3 \times 75 + 75 = 100$. On the other hand, the optimal allocation under true type report by all agents (including agent 1) is $\pi(\theta) = (75, 50, 25)$. Since the penalty to agent 1 is 0, the transfer to agent 1 can be calculated from the marginal contribution component x_1 as $(2 \times 50 + 4 \times 100) - (2 \times 50 + 4 \times 25) = 300$. Thus, the utility derived by agent 1 is $300 - 3 \times 75 + 75 = 150$, which is strictly greater than the utility of 100 derived under the case when agent 1 misreported its type.

The main result concerning the properties of $\mathcal{MC}$ is given as follows:

THEOREM 1. *The mechanism $\mathcal{MC}$ is strategyproof and individually rational.*

PROOF. Strategyproofness of $\mathcal{MC}$ will be established by the following steps: (i) First showing that reporting the true unit price is utility maximizing for agent i, regardless of the reported capacity of agent i and of what other agents report, i.e., $U_i(\hat{\theta}) \leq U_i((u_i, \hat{c}_i), \hat{\theta}_{-i})$. (ii) Next, showing that reporting the true capacity is utility maximizing for agent i, given true unit price report and regardless of what other agents report, i.e., $U_i((u_i, \hat{c}_i), \hat{\theta}_{-i}) \leq U_i(\theta_i, \hat{\theta}_{-i})$. The reader is referred to [7] for a detailed proof of the above theorem. □

3. DYNAMIC MECHANISM WITH CONSTRAINTS ($\mathcal{DMC}$)

Unlike $\mathcal{MC}$, here we consider a dynamic setting that involves an infinite time horizon over which the individual agent types evolve. The $\mathcal{DMC}$ that we develop in the following involves 1. a socially efficient allocation which differs from B&V [2] as the capacity constraints are included, and 2. a novel delayed transfer scheme that involves a penalty component apart from the payment.While the payment to an agent is his marginal contribution as in B&V [2], the penalty component that ensures true capacity reports is new. We illustrate the need for the delayed penalty based scheme via an example and also prove that $\mathcal{DMC}$ is incentive compatible in Theorem 2.

3.1 The Setting

Consider a setting where there are N agents (numbered $1, 2, \dots, N$), capable of performing certain manufacturing tasks. A central controller receives new tasks from external sources. We use an infinite horizon discrete time-line $1, 2, \dots, n, \dots$ to indicate allocation of new tasks and also completion of tasks by individual agents. We allow for completion of a particular task by different agents to be at different instants. Upon arrival of a new task at current instant requiring manufacturing of $D \in \Theta_c$ number of units of commodity, the central controller performs allocation to all the agents based on the current capacity c_i and unit cost u_i of manufacturing for each agent i. Let $\theta_i = (u_i, c_i) \in \Theta_i$ be the cost-capacity type vector for agent i and $\theta = (D, \theta_1, \theta_2, \dots, \theta_N) \in \Theta \triangleq \Theta_c \times \Theta_1 \times \Theta_2 \times \dots \times \Theta_N$ be the joint type vector. As before, we use sub-script $-i$ to denote all agents other than i. For instance, $\theta_{-i} = (D, \theta_1, \theta_2, \dots, \theta_{i-1}, \theta_{i+1}, \dots, \theta_N) \in \Theta_{-i} \triangleq \Theta_c \times \Theta_1 \times \Theta_2 \times \dots \times \Theta_{i-1} \times \Theta_{i+1} \times \dots \times \Theta_N$.

We assume that θ evolves as a Markov process on the discrete-time horizon. Also, we assume that the Markov process thus formed is ergodic for all allocation scenarios from the central controller. Thus, we restrict ourselves to stationary Markovian allocation policies where allocation is given by $y : \Theta \to \mathcal{R}^N$, i.e., at a given instant if the Markov process is at θ, the allocation is given by $y(\theta)$. Let $y_i : \Theta \to \mathcal{R}$ denote the allocation to agent i. Let $\mathcal{Y}$ be the set of all admissible allocation scenarios. When necessary for clarification, we use super-script n to denote a quantity at time instant n. For example, $\theta^n \in \Theta$ represents the types at time instant n, D^n represents the number of units to be manufactured at time instant n, etc. Note that unlike the setting in the static mechanism $\mathcal{MC}$, this setting for dynamic mechanism allows for all parameters including the manufacturing capacity c_i of each agent i, to evolve or change with time instants.

Allocative Efficiency

Given the Markov process of θ, the central controller needs to allocate to all agents in a way that the total cost of manufacturing is minimized with no agent asked to manufacture beyond its capacity. Also, the central controller needs to ensure that the requirement of manufacturing of a particular D units for the task at each instant is met. Thus, the central controller performs optimal allocation according to the following optimization problem:

$$\left.\begin{array}{l} \text{Find } \pi \triangleq \underset{y \in \mathcal{Y}}{\text{argmin}} \sum_{\theta \in \Theta} \sum_{i=1}^{N} V_i(\theta, y) \\ \text{s.t.} \\ 0 \leq y_i(\theta) \leq c_i, i = 1, 2, \dots, N, \forall \theta \\ \sum_{i=1}^{N} y_i(\theta) = D, \forall \theta, \end{array}\right\} \quad (8)$$

$$\text{where } V_i(\theta, y) = \text{E}\left[\sum_{k=0}^{\infty} \gamma^k v_i(\theta^k, y_i) | \theta^0 = \theta, y\right]$$

represents the discounted cost infinite horizon value added to the system by agent i, where $v_i(\theta^k, y_i) = u_i^k y_i(\theta^k)$ represents the cost incurred at instant k by that agent and $0 < \gamma < 1$ is the discount factor. The expectation is over various sample paths $(\theta, \theta^1, \theta^2, \dots)$ starting with θ. The expression for $V_i(\theta, y)$ could be rearranged using dynamic programming as follows:

$$V_i(\theta, y) = v_i(\theta, y_i) + \gamma \text{E}\left[V_i(\theta', y) | \theta\right],$$

where the expectation is now over the next state $\theta' \in \Theta$. We assume that for all possible $\theta \in \Theta$, $\sum_{i=1}^{N} c_i \geq D$, so that there are feasible solutions available for the above optimization problem. In the optimization problem (8), we assume that only one optimal allocation exists which is defined to be $\pi \in \mathcal{Y}$. In cases where there are multiple optimal allocations possible, π is assumed to represent any one of them. We also formulate an auxiliary optimization problem which is for computing optimal allocation without an agent i:

$$\left.\begin{array}{l}\text{Find } \pi_{-i} \stackrel{\triangle}{=} \underset{y\in\mathcal{Y}_{-i}}{\text{argmin}} \sum_{\theta\in\Theta} \sum_{\substack{j=1\\j\neq i}}^{N} V_j(\theta,y)\\ \text{s.t.}\\ 0\le y_j(\theta)\le c_j, j=1,2,\ldots,N, j\neq i,\forall\theta\\ \sum_{\substack{j=1\\j\neq i}}^{N} y_j(\theta)=D,\forall\theta.\end{array}\right\} \quad (9)$$

In the above, $\mathcal{Y}_{-i}$ denotes the set of admissible allocation policies of the form $y : \Theta_{-i} \rightarrow \mathcal{R}^{N-1}$. The optimal allocation thus obtained is denoted by $\pi_{-i} \in \mathcal{Y}_{-i}$. Also, $\pi_{-i,j} : \Theta_j \rightarrow \mathcal{R}$ represents efficient allocation to agent j without participation by agent i.

Given θ, the problem thus is simply to solve the optimization problem (8). However, in practical circumstances there are two potential issues: **(i)** θ_i is the private information of agent i. Thus, the controller needs to derive this information either directly or indirectly. In either case, the problem is to obtain the correct θ-value as otherwise the allocation via π may not make any useful sense; and **(ii)** The capacity, though assumed to be c_i by an agent i, would be subject to realization. Thus, the actual capacity attained by the end of manufacturing may be $\bar{c}_i$ which could be different from c_i. So, the problem here is that the allocation is done at the beginning of a task while the attained capacity ($\bar{c}_i$) is known only at the end of the task.

In order to address these issues, we will formulate in the next subsection, a dynamic mechanism (π, t) where π is an optimal allocation policy discussed previously and $t = \left(t^1, t^2, \ldots\right)$ is the payment policy for the entire time horizon. At each time instant, agents are asked to report their types, represented by $\hat{\theta} = \left(D, \hat{\theta}_1, \hat{\theta}_2, \ldots, \hat{\theta}_N\right)$, based on which a suitable payment t is given by the controller. Let $\bar{\theta}_i = (\hat{u}_i, \bar{c}_i)$ represent the type with achieved capacity of agent i. Then, the payment for each allocation corresponding to instant n is of the functional form, $t_i^n : \Theta_i \rightarrow \mathcal{R}$, i.e., $t_i^n(\bar{\theta}_i, \hat{\theta}_i, \hat{\theta}_{-i})$.

REMARK 5. *As in the case of the static setting, the payment to agent i is made using the report type vector ($\hat{\theta}^k$) as well as its achieved capacity ($\bar{c}_i$). The need for using the achieved capacity of an agent (known only at a later instant, see Figure 1) is illustrated via Example 2.*

We assume that $\sigma_i : \Theta_i \rightarrow \Theta_i$, a stationary strategy, is used by agent i to report its type, i.e., $\hat{\theta}_i = \sigma_i(\theta_i), \forall i = 1, \ldots, N$. Henceforth, we use $\hat{\theta}_i$ and $\sigma_i(\theta_i)$ interchangeably. Let $\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_N)$. Also, we use $V_i(\theta, \pi, \sigma)$ interchangeably with $V_i(\hat{\theta}, \pi)$.

We define the utility U_i realized by participation by agent i as follows.

DEFINITION 4. *The utility U_i realized by agent i is given by*

$$U_i(\theta, \pi, \sigma_i) = E\left[\sum_{k=0}^{\infty} \gamma^k \tilde{U}_i(\theta_i^k, \pi_i, \hat{\theta}^k) | \theta^0 = \theta, \pi, \sigma_i\right], \quad (10)$$

where

$$\tilde{U}_i(\theta_i^k, \pi_i, \hat{\theta}^k) = t_i^k(\bar{\theta}_i^k, \hat{\theta}^k) - v_i(\theta_i^k, \pi_i) + \pi_i((u_i^k, \hat{c}_i^k), \hat{\theta}_{-i}^k) \quad (11)$$

is the quasi-linear single-stage utility of agent i at instant k. While the first component in $\tilde{U}_i(\cdot)$ above is the transfer of agent i, the second component is the cost and the third component is the allocation received under agent i's (possibly misreported) type. Note that the utility function formulation above is different from the one used in [2]. The difference is that, unlike [2], we have an extra additive term (the last term in (11)) that denotes the allocation to agent i at instant k, under true unit price report and a possibly overstated capacity report. As noted in the static setting, this factor is incorporated to capture the preference of an agent to harm others by obtaining a higher allocation. Further, as in the static setting, the reward function $f(\cdot)$ can be incorporated to convert from allocation units to money in the last term of (11).

Summing the individual components over the infinite horizon, one obtains the following form for the utility of agent i:

$$U_i(\theta, \pi, \sigma_i) = T_i(\hat{\theta}, \pi, \sigma_i) - V_i(\theta, \pi, \sigma_i) + \mathcal{A}_i(\tilde{\theta}, \pi, \sigma_i),$$

where

$$T_i(\hat{\theta}, \pi, \sigma_i) = E\left[\sum_{k=0}^{\infty} \gamma^{k+\delta_i(k)} t_i(\bar{\theta}_i^k, \hat{\theta}^k) | \theta^0 = \hat{\theta}, \pi, \sigma\right], \quad (12)$$

$$\mathcal{A}_i(\tilde{\theta}, \pi, \sigma_i) = E\left[\sum_{k=0}^{\infty} \gamma^k \pi_i(\tilde{\theta}^k) | \theta^0 = \tilde{\theta}, \pi, \sigma\right], \quad (13)$$

where $\tilde{\theta} = ((u_i, \hat{c}_i), \hat{\theta}_{-i})$ and $\tilde{\theta}^k = ((u_i^k, \hat{c}_i^k), \hat{\theta}_{-i}^k)$. In (12)*, $\delta_i(k)$ is the time taken by agent i to complete the task allocated at instant k. As discussed later, $t_i(\bar{\theta}_i^k, \hat{\theta}^k)$ is a payment made to agent i after completion of the task by him. So, in order to account for the delay, $\gamma^{\delta_i(k)}$ discount factor is used in the above definition of the total transfer to agent i.*

We assume standard definitions of within-period ex-post Nash equilibrium, incentive compatibility and individual rationality (see, for instance, [3]).

3.2 Motivation

Capacity constraints are not considered previously in the context of dynamic mechanism design. For instance, B&V [2] propose a marginal contribution based transfer scheme for allocating a common resource. The transfer scheme proposed in B&V's work [2] cannot be applied in our setting where agents have limited capacities. This is because the transfer scheme of B&V [2] would not ensure true capacity reports from the agents. On the other hand, in $\mathcal{DMC}$ the same is ensured via a delayed penalty based transfer scheme.

EXAMPLE 2. *We illustrate by this example that the transfer scheme from the dynamic pivot mechanism [2] would not ensure incentive compatibility, when the setting involves capacity constraints. Consider a scenario where $D^n = 150, n \geq 0$. Let there be three agents with their types given by $(u_1^n, c_1^n) = (1, 100)$, $(u_2^n, c_2^n) = (2, 50)$ and $(u_3^n, c_3^n) = (3, 100)$, for all $n \geq 0$. Fix a time instant n and suppose that the reported types at n are given by $(\hat{u}_1^n, \hat{c}_1^n) = (1, 125)$, $(\hat{u}_2^n, \hat{c}_2^n) = (2, 50)$ and $(\hat{u}_3^n, \hat{c}_3^n) = (3, 100)$. Also, assume that the agents report truthfully for all time instants $m > n$. Hence, we have a scenario where agent types are static and agent 1 misreports his capacity at time instant n.*

From the fact that the future joint type of agents remain the same in the above setting and also that the agents report truthfully in the future, we have that $\pi_i(\theta^k) \equiv \pi_i$, a constant allocation for all time instants. Hence, the efficient allocation to agents remains as $(100, 50, 0)$ for all time instants $m > n$ in the future. Further, the value function of agent i turns out as follows:

$$V_i(\theta^m, \pi) = \sum_{k=m}^{\infty} \gamma^{k-m} u_i^k \pi_i(\theta^k) = u_i \pi_i \sum_{k=m}^{\infty} \gamma^{k-m} = \frac{u_i \pi_i}{1-\gamma}.$$

Assuming $\gamma = \frac{3}{4}$, we obtain

$$V(\theta^m, \pi) = \sum_{i=1}^{3} V_i(\theta^m, \pi) = \frac{1 \times 100 + 2 \times 50 + 3 \times 0}{\frac{1}{4}} = 800.$$

Also, it is easy to see that $V_1(\theta^m, \pi) = 400 = V_2(\theta^m, \pi)$ and $V_3(\theta^m, \pi) = 0, \forall m > n$.

We denote the payment made to agent i at instant n using the transfer scheme from B&V's mechanism [2] as $\tilde{x}_i^n(\hat{\theta})$, i.e.,

$$\tilde{x}_i^n(\hat{\theta}) = V_{-i}(\hat{\theta}, \pi_{-i}) - \Bigg(v_{-i}(\hat{\theta}_{-i}, \pi(\hat{\theta})) + \gamma \boldsymbol{E}_{\theta'} \left[V_{-i}(\theta', \pi_{-i}) | \hat{\theta}, \pi(\hat{\theta}) \right] \Bigg).$$

For the given example, since types are constant for all time instants, this payment simplifies to

$$\tilde{x}_i^n(\hat{\theta}) = V_{-i}(\hat{\theta}, \pi_{-i}) - \left[v_{-i}(\hat{\theta}_{-i}, \pi(\hat{\theta})) + \gamma V_{-i}(\hat{\theta}, \pi_{-i}) \right].$$

We observe that for instant n, the efficient allocation is $\pi(\hat{\theta}^n) = (125, 25, 0)$ and $\pi_{-1}(\hat{\theta}^n_{-1}) = (50, 100)$. Hence, $V_{-1}(\hat{\theta}, \pi_{-1}) = \dfrac{(2 \times 50 + 3 \times 100)}{\frac{1}{4}} = 1600$ and

$$\tilde{x}_1^n(\hat{\theta}) = 1600 - (2 \times 25 + \frac{3}{4} \times 1600) = 350,$$

while with true report, the agent would have got,

$$x_1^n(\theta) = 1600 - (2 \times 50 + \frac{3}{4} \times 1600) = 300.$$

Hence, the mechanism of B&V [2] did not penalize agent 1 for misreporting his capacity. In the $\mathcal{DMC}$ scheme that we present subsequently, we design a delayed transfer scheme with a penalty that compensates the loss caused by agent 1's capacity misreport.

Note that any mechanism which gives a penalty < 0 to an agent upon misreport of capacity by him will not necessarily be ex-post incentive compatible. For instance, we could simply extend the static mechanism of [6], i.e., give a constant penalty $p < 0$ upon misreport regardless of the value of the misreported capacity. However, as observed in the static case, the utility derived by an agent i in our setting, i.e., according to (10), may be higher with a misreported capacity as compared to that with true capacity report. This is clear in the example setting above where the utility derived under true report by agent is $300 - 1 \times 100 + 100 = 300$, whereas with a capacity misreport (agent 1 overstating his capacity to be 125), the utility is $300 - \delta - 1 \times 100 + 125 = 325 - \delta$.

As we observed in the above example, an agent has an incentive to overstate his capacity. Our mechanism $\mathcal{DMC}$ results in truthful reporting by the agents in the system by means of a delayed transfer scheme with a variable penalty component. An important aspect of the mechanism is that on completion of a production task by agent i, the allocation is re-computed with achieved capacity of agent i being used instead of the reported capacity. The policy that results from this allocation is used in deciding the transfer for agent i. We formalize these aspects of $\mathcal{DMC}$ below.

3.3 The Mechanism

We employ a direct dynamic mechanism in which the central controller solicits type information $\hat{\theta}$ from each agent i at every time instant n. Based on the reported types, the central controller then makes an efficient allocation. The payment, however, is not

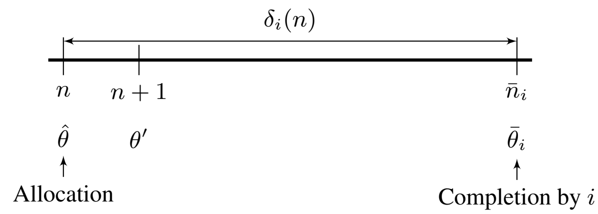

Figure 1: A portion of the time-line illustrating the process

immediately disbursed. Instead, the agent continues to produce according to the allocation and on completion of the production task, the central controller observes the achieved capacity level of the agent and makes a payment which is the marginal contribution.

We use the notation as in Table 1 for denoting allocations with reported and achieved types and combinations thereof. The overall allocation process in the dynamic mechanism can be illustrated by the Figure 1. At instant n, an allocation is performed based on reported types $\hat{\theta}$ of all the agents. The task gets completed by agent i in a time span of $\delta_i(n)$. The achieved capacity of agent i is captured in $\bar{\theta}_i = (\hat{u}_i, \bar{c}_i)$ at completion instant $\bar{n}_i$.

The transfer $t_i(\bar{\theta}_i, \hat{\theta})$ to agent i for the task allocated at instant n is performed upon completion of the task by that agent. In order to account for this delay in payment, a compounding factor of $\frac{1}{\gamma^{\delta_i(n)}}$ is used in computing the transfer t_i^n. $\mathcal{DMC}$ can be described step-by-step as follows:

1. Solicit the types, i.e., the unit price and capacity of each of the agents. The reported type vector is $\hat{\theta}$ as noted before.
2. Perform an allocation $\pi(\hat{\theta})$, by solving the system (8) using the reported unit prices and capacities of the agents.
3. The individual agents perform production tasks as per the above allocation.
4. Each agent i completes his part of the allocated task at instant $\bar{n}_i$. Wait till all agents finish their portions of the task.
5. Compute the transfer to each agent according to (14). The above procedure is repeated in the infinite horizon.

Transfer scheme

The expression for t_i^n is as given below:

$$t_i(\bar{\theta}_i, \hat{\theta}) = \frac{1}{\gamma^{\delta_i(n)}} \left[x_i(\bar{\theta}_i, \hat{\theta}) + p_i(\bar{\theta}_i, \hat{\theta}) \right], \tag{14}$$

which is composed of two additive components: (i) $x_i(\bar{\theta}_i, \hat{\theta})$, the marginal gain brought into the process by agent i's participation at instant n, and (ii) $p_i(\bar{\theta}_i, \hat{\theta})$, the penalty imposed on agent i to cover the damage caused to the process by misreport of capacity by him. Let

$$W_{-i}(\theta) = v_{-i}(\theta_{-i}, \pi(\theta)) + \gamma\, \mathbf{E}_{\theta'} \left[V_{-i}(\theta', \pi_{-i}) | \theta, \pi(\theta) \right],$$

where

$$v_{-i}(\theta_{-i}, \pi_{-i}) = \sum_{\substack{j=1 \\ j \neq i}}^{N} v_j(\theta, \pi_{-i,j}), V_{-i}(\theta, \pi_{-i}) = \sum_{\substack{j=1 \\ j \neq i}}^{N} V_j(\theta, \pi_{-i}).$$

In the above, the expectation is over $F(\theta' | \theta, \pi(\theta))$, the conditional distribution of the next state θ' of the Markov process, given θ and the allocation policy π. The two components of transfer $t_i(\bar{\theta}_i, \hat{\theta})$ can be now written as

$$x_i(\bar{\theta}_i, \hat{\theta}) = V_{-i}(\hat{\theta}, \pi_{-i}) - W_{-i}(\bar{\theta}_i, \hat{\theta}_{-i}), \text{ and} \tag{15}$$

$$p_i(\bar{\theta}_i, \hat{\theta}) = \sum_{j \neq i} \pi_j(\hat{\theta}) - \sum_{j \neq i} \pi_j(\bar{\theta}_i, \hat{\theta}_{-i}). \quad (16)$$

The quantity $V_{-i}(\hat{\theta}, \pi_{-i})$ in (15) represents the total cost incurred by all agents other than i if agent i had not participated at all while the second term in the equation represents the total cost incurred by them if agent i did participate. Thus, the difference gives the marginal gain brought in by agent i into the setting at instant n. On the other hand, because of the misreporting of capacity, the possible loss caused by agent i is characterized in $p_i(\bar{\theta}_i, \hat{\theta})$ in equation (16). If not for capacity related constraints, $p_i(\bar{\theta}_i, \hat{\theta})$ would not be needed and the payment can simply be the marginal contribution $x_i(\bar{\theta}_i, \hat{\theta})$ which may have been given at time instant n itself without waiting for task to be completed. But as shown previously in an example, the marginal contribution alone given as payment would not be enough to ensure that revealing true capacity is incentive compatible. As will be shown later, the additive marginal compensation $p_i(\bar{\theta}_i, \hat{\theta})$ is sufficient to ensure this fact. In equation (16), the first term represents the total cost incurred by agents other than i when agent i misreports its capacity and the second term represents the total cost incurred by them when the achieved capacity by agent i is used instead of its reported capacity. Thus one can see that the difference gives the net loss created by the misreport of capacity by agent i. As noted before in the static setting, the penalty $p_i(\bar{\theta}_i, \hat{\theta})$ can be seen as equivalent to $\pi_i(\bar{\theta}_i, \hat{\theta}_{-i}) - \pi_i(\hat{\theta})$. Further, the reward function can be included in the penalty component in exactly the same manner as in the static setting (see Remark 4). Several remarks are in order.

REMARK 6. *Comparing with the transfer scheme in [2], i.e., $\tilde{x}_i(\hat{\theta})$ and the payment in our scheme, i.e., $x_i(\bar{\theta}_i, \hat{\theta})$, we note that both are similar in form except that the former is computed without the achieved capacity and is paid at the allocation instant. However, as discussed previously in example 2, agent i will have incentive to misreport capacity in the former case.*

REMARK 7. *In the example mentioned above, $\bar{c}_1^n = 100$ and hence, $\bar{\pi}(\bar{\theta}_i^n, \hat{\theta}_{-i}^n) = (100, 50, 0)$. Using the transfer scheme of $\mathcal{DMC}$ (14), we obtain:*

$$\begin{aligned} x_i^n(\bar{\theta}_i^n, \hat{\theta}^n) &= 1600 - (100 + \tfrac{3}{4} \times 1600) = 300, \\ p_i^n(\bar{\theta}_i^n, \hat{\theta}^n) &= 25 - 50 = -25 < 0. \end{aligned} \quad (17)$$

The utility derived by agent 1 with an overstated capacity of 125, can be calculated as $300 - 25 - 1 \times 100 + 125 = 250$. On the other hand, as noted before, the utility with true capacity report turns out to be 300. Hence, in comparison to the dynamic pivot mechanism's [2] transfer scheme, $\mathcal{DMC}$ penalizes the agent i to the extent of the damage his misreport caused.

The main result concerning the properties of $\mathcal{DMC}$ is given as follows:

THEOREM 2. *$\mathcal{DMC}$ is ex-post incentive compatible and ex-post individually rational.*

PROOF. Suppose there is a simplified mechanism where we know the achieved capacity of agent i at the instant of allocation. Hence, the payment can be made in each period without any delay. Let $\tilde{t}_i$ denote the payment in this simplified mechanism, i.e.,

$$\tilde{t}_i(\bar{\theta}_i, \hat{\theta}) = \left[x_i(\bar{\theta}_i, \hat{\theta}) + p_i(\bar{\theta}_i, \hat{\theta}) \right],$$

which does not have any scaling factor with γ. Let $\tilde{T}_i(\hat{\theta})$ denote the total payment in the simplified mechanism. So,

$$\tilde{T}_i(\hat{\theta}) = E\left[\sum_{k=0}^{\infty} \gamma^k \tilde{t}_i(\hat{\theta}^k, \bar{\theta}_i) | \theta^0 = \hat{\theta}, \pi, \sigma \right].$$

Note that $\tilde{T}_i(\hat{\theta}) = T_i(\hat{\theta})$ because

$$\begin{aligned} &E\left[\sum_{k=0}^{\infty} \gamma^{k+\delta_i(k)} t_i(\hat{\theta}^k, \bar{\theta}_i) | \theta^0 = \hat{\theta}, \pi, \sigma \right] \\ =&E\left[\sum_{k=0}^{\infty} \gamma^k \tilde{t}_i(\hat{\theta}^k, \bar{\theta}_i) | \theta^0 = \hat{\theta}, \pi, \sigma \right]. \end{aligned}$$

So, we use $T_i(\hat{\theta})$ itself to represent the payment rather that $\tilde{T}_i(\hat{\theta})$ in the simplified mechanism. The proof follows by first establishing that this simplified mechanism is incentive compatible and then, by proving that the expected discounted sum of transfers in our mechanism is equivalent to this simplified mechanism where transfers are made without any delay. The reader is referred to [7] for a detailed proof of the above theorem. ☐

4. CONCLUSIONS

We presented two novel mechanisms with progressively realistic assumptions about agent types aimed at economic scenarios where agents have limited capacities and are hostile. For the simplest case where agent types consist of a unit cost of production and a capacity that does not change with time, we proposed a novel utility function and mechanism $\mathcal{MC}$ that extends the work of [6] with a novel variable penalty based transfer scheme to address the hostile agents over-reporting capacities to harm competing agents. We established the strategyproofness of this mechanism. Next, we proposed $\mathcal{DMC}$ that accommodates agents having dynamic types in $\mathcal{MC}$. A penalty scheme is needed for achieving truthful reporting of capacities. However, penalties cannot be determined until the actual number of units produced by agents are known. We showed the non-triviality of extending the current dynamic frameworks in determining the payoffs in the case of limited capacities. In $\mathcal{DMC}$, this was achieved by adjusting the payoffs upwards based on a compounded future discount factor. We showed that $\mathcal{DMC}$ possesses the desired properties of ex-post incentive compatibility, individual rationality, and allocative efficiency.

5. ACKNOWLEDGMENTS

The first author would like to thank the European Community's Seventh Framework Programme (FP7/2007 – 2013) under grant agreement n^o 270327 for funding the research leading to these results. The work of S. Bhatnagar was supported in part by a Department of Science and Technology project.

6. REFERENCES

[1] S. Athey and I. Segal. An efficient dynamic mechanism. Technical report, Stanford University, 2007.

[2] D. Bergemann and J. Valimaki. The dynamic pivot mechanism. *Econometrica*, 78(2):771–789, 2010.

[3] R. Cavallo. Efficiency and redistribution in dynamic mechanism design. In *ACM EC*, pages 220–229. ACM, 2008.

[4] R. Cavallo. Mechanism design for dynamic settings. *SIGecom Exch.*, 8:7:1–7:5, December 2009.

[5] R. Cavallo, D. Parkes, and S. Singh. Efficient mechanisms with dynamic populations and dynamic types. *Unpublished manuscript, Harvard University*, 2009.

[6] R. K. Dash, P. Vytelingum, A. Rogers, E. David, and N. R. Jennings. Market-based task allocation mechanisms for limited-capacity suppliers. *IEEE TSMC*, 37(3):391–405, 2007.

[7] L. A. Prashanth, H. L. Prasad, N. Desai, and S. Bhatnagar. Mechanisms for Hostile Agents with Capacity Constraints. Technical report, Stochastic Systems Lab, IISc, 2013.