Fault Tolerant Mechanism Design for General Task Allocation[*]

Dengji Zhao   Sarvapali D. Ramchurn   Nicholas R. Jennings

Electronics and Computer Science
University of Southampton
Southampton, SO17 1BJ, UK
{dz1a13, sdr, nrj}@ecs.soton.ac.uk

ABSTRACT

We study a general task allocation problem, involving multiple agents that collaboratively accomplish tasks and where agents may fail to successfully complete the tasks assigned to them (known as execution uncertainty). The goal is to choose an allocation that maximises social welfare while taking their execution uncertainty into account (i.e., fault tolerant). To achieve this, we show that the post-execution verification (PEV)-based mechanism presented by Porter et al. (2008) is applicable if and only if agents' valuations are risk-neutral (i.e., the solution is almost universal). We then consider a more advanced setting where an agent's execution uncertainty is not completely predictable by the agent alone but aggregated from all agents' private opinions (known as trust). We show that PEV-based mechanism with trust is still applicable if and only if the trust aggregation is multilinear. Given this characterisation, we further demonstrate how this mechanism can be successfully applied in a real-world setting. Finally, we draw the parallels between our results and the literature of efficient mechanism design with general interdependent valuations.

CCS Concepts

•Theory of computation → Algorithmic mechanism design; •Computing methodologies → Multi-agent systems;

Keywords

Execution uncertainty; interdependent valuations; type verification; trust aggregation

1. INTRODUCTION

We study a general task allocation problem, where multiple agents collaboratively accomplish a set of tasks. However, agents may fail to successfully complete the task(s) allocated to them (known as execution uncertainty). Such task allocation problems arise in many real-world applications such as transportation networks [17], data routing [16], cloud computing [1] and the sharing economy [2]. Execution uncertainty is typically unavoidable in these applications due to unforeseen events and limited resources, especially sharing economy applications such as *Uber* and *Freelancer.com*, where services are mostly provided by individuals with no qualifications or certifications.

In addition to the execution uncertainty underlying the task allocation problem, the completion of a task may also depend on the completion of other tasks, e.g., in *Uber* a rider cannot travel without a driver offering the ride. The completion of the tasks of an allocation gives a (private) value to each agent, and our goal is to choose an allocation of tasks that maximises the total value of all agents, while taking their execution uncertainty into account.

It has been shown that traditional mechanism design (based on Groves mechanisms [5]) is not applicable to settings that involve execution uncertainty [14, 4]. This is because execution uncertainty implies interdependencies between the agents' valuations (e.g., a rider's value for a ride will largely depend on whether the driver will successfully finish the drive). To combat this problem, Porter et al. [14] have proposed a solution based on post-execution verification (PEV), which is broadly aligned with type verification [11]. The key idea behind the PEV-based mechanism is that agents are paid according to their task executions (i.e., whether they complete their tasks or not), rather than what they have reported.

While Porter et al. [14] considered a single task requester setting where one requester has multiple tasks that can be completed by multiple workers, Stein et al. [18] and Conizter and Vidali [4] studied similar settings but considering workers' uncertain task execution time. Moreover, Ramchurn et al. [15] looked at a more complex setting where each agent is a task requester and is also capable of completing some tasks for the others. Apart from considering different settings, all the solutions in these studies are PEV-based. However, these results may not apply to other different problem settings where, for example, agents' valuations may have externalities, e.g., agent A prefers working with B instead of others [7], and an agent may even incur some costs without doing any task, e.g., when a government builds a costly public good [8].

Therefore, in this paper, we study a general task allocation setting where agents' valuations are not constrained (as in previous works). Under this general setting, we characterise the applicability of the PEV-based mechanism. Our

[*]The authors acknowledge funding from the EPSRC-funded International Centre for Infrastructure Futures (ICIF) (EP/K012347/1).

Appears in: *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2016), J. Thangarajah, K. Tuyls, C. Jonker, S. Marsella (eds.), May 9–13, 2016, Singapore.*

contributions advance the state of the art in the following ways:

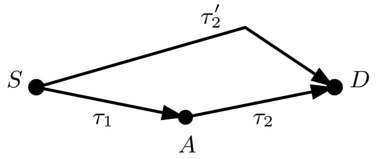
- We propose a generalisation of the work of Porter et al. [14] and Ramchurn et al. [15]: (1) Both [14] and [15] showed that PEV-based mechanisms are ex-post truthfully implementable in their limited settings, while this paper shows that PEV-based mechanisms are ex-post truthful for all risk-neutral settings (with or without externalities). (2) Ramchurn et al. [15] further showed that PEV-based mechanisms are also ex-post truthful in a trust-based environment with linear trust aggregation, while this paper shows that truthfulness still holds for a much wider class of trust aggregations, namely all multilinear aggregations.
- We also bound the applicability of the PEV-based mechanisms by showing that risk-neutral valuations and multilinear aggregations are also necessary conditions for achieving ex-post truthfulness. This bound has never been shown in any valuation setting before.
- Given the above characterisation, we demonstrate how this mechanism can be successfully applied in the real-world. We consider ridesharing as an example, where people may have very complex preferences for sharing with others (i.e., externalities exist naturally).
- Since our problem setting is essentially an interdependent valuation setting, we also draw the parallels between our results and the literature of efficient mechanism design with interdependent valuations, which further demonstrates the significance of the PEV-based mechanism.

The remainder of the paper is organized as follows. Section 2 presents the general task allocation model. Section 3 introduces the PEV-based mechanism and characterises its applicability. We then extend the characterisation to trust-based environments in Section 4 and show a promising application example in Section 5. Lastly, we compare our results to the literature of efficient design with general interdependent valuations.

2. THE MODEL

We study a task allocation problem where there are n agents denoted by $N = \{1, ..., n\}$ and a finite set of task allocations T[1] (e.g., rides to ride-sharers or packages to delivery trucks). Each allocation $\tau \in T$ is defined by $\tau = (\tau_i)_{i \in N}$, where τ_i is a set of tasks assigned to agent i. Let $\tau_i = \emptyset$ if there is no task assigned to i in τ. For each allocation τ, agent i may fail to successfully complete her tasks τ_i, which is modelled by $p_i^\tau \in [0, 1]$, the probability that i will successfully complete her tasks τ_i. Let $p_i = (p_i^\tau)_{\tau \in T}$ be i's **probability of success (PoS)** profile for all allocations T, and $p^\tau = (p_i^\tau)_{i \in N}$ be the PoS profile of all agents for allocation τ.

Note that the completion of one task in an allocation may depend on the completion of the other tasks. Take the delivery example in Figure 1 with two agents $1, 2$ delivering one package from S to D. There are two possible task allocations to finish the delivery: τ is collaboratively executed by agents 1 and 2, while τ' is done by agent 2 alone. It is clear that task τ_2 depends on τ_1. However, p_2^τ only indicates 2's PoS for τ_2, assuming that 1 will successfully complete τ_1. That is, p_i^τ does not include task dependencies and it only specifies i's probability to successfully complete τ_i, if τ_i is ready for i to execute.



$$Allocations: \quad \begin{array}{l} \tau = (\tau_1 = \{S \to A\}, \tau_2 = \{A \to D\}) \\ \tau' = (\tau_1' = \emptyset, \tau_2' = \{S \to D\}) \end{array}$$

Figure 1: Package delivery from S to D with two agents $1, 2$

agent	allocation	p_i	v_i
1	τ	$p_1^\tau = 0$	$v_1(\tau, p^\tau) = 0$
	τ'	$p_1^{\tau'} = 0$	$v_1(\tau', p^{\tau'}) = 0$
2	τ	$p_2^\tau = 1$	$v_2(\tau, p^\tau) = p_1^\tau \times p_2^\tau$
	τ'	$p_2^{\tau'} = 0.5$	$v_2(\tau', p^{\tau'}) = p_2^{\tau'}$

Table 1: A valuation setting for the exmaple in Figure 1

For each allocation $\tau \in T$, the completion of τ brings each agent i a value (either positive or negative), which combines costs and benefits. For example, offering a ride to someone in ridesharing may incur a detour cost to the driver, but it may also bring both the driver and the rider a valuable journey. Considering the execution uncertainty, agent i's valuation is modelled by a function $v_i : T \times [0, 1]^N \to \mathbb{R}$, which assigns a value to each allocation τ, for each PoS profile $p^\tau = (p_i^\tau)_{i \in N}$.

For each agent i, we assume that v_i and p_i are privately observed by i, known as i's **type** and denoted by $\theta_i = (v_i, p_i)$. Let $\theta = (\theta_i)_{i \in N}$ be the type profile of all agents, θ_{-i} be the type profile of all agents except i, and $\theta = (\theta_i, \theta_{-i})$. Let Θ_i be i's type space, $\Theta = (\Theta_i)_{i \in N}$ and $\Theta_{-i} = (\Theta_j)_{j \neq i \in N}$.

Given the above setting, our goal is to choose one task allocation from T that maximises all agents' valuations, i.e., a socially optimal allocation. This can be achieved (according to the revelation principle [10]) by designing a mechanism that directly asks all agents to report their types and then chooses an allocation maximising their valuations. However, agents may not report their types truthfully. Therefore, we need to incentivize them to reveal their true types, which is normally achieved by choosing a specific allocation of tasks and an associated monetary transfer to each agent. The direct revelation **allocation mechanism** is defined by a task **allocation choice** function $\pi : \Theta \to T$ and a **payment** function $x = (x_1, ..., x_n)$ where $x_i : \Theta \to \mathbb{R}$ is the payment function for agent i.

2.1 Solution Concepts

The goal of the allocation mechanism is to choose a task allocation that maximises the valuation of all agents, i.e., the social welfare. Since the agents' types are privately observed by the agents, the mechanism is only able to maximise social welfare if it can receive their true types. Therefore, the mechanism needs to incentivise all agents to report their

[1] T is the task allocation outcome space, which may contain all feasible task allocations (with or without redundancy) that agents can execute. The precise definition depends on the applications.

types truthfully. Moreover, agents should not lose when they participate in the task allocation mechanism, i.e., they are not forced to join the allocation. In the following, we formally define these concepts.

We say an allocation choice π is efficient if it always chooses an allocation that maximises the expected social welfare for all type report profiles.

Definition 1. *Allocation choice π is **efficient** if and only if for all $\theta \in \Theta$, for all $\tau' \in T$, let $\tau = \pi(\theta)$, we have:*

$$\sum_{i \in N} v_i(\tau, p^{\tau}) \geq \sum_{i \in N} v_i(\tau', p^{\tau'})$$

where $p^{\tau} = (p_i^{\tau})_{i \in N}$, and $p^{\tau'} = (p_i^{\tau'})_{i \in N}$.

Note that the expected social welfare calculated by π is based on the agents' reported types, which are not necessarily their true types. However, agents' actual/realized valuation for an allocation only depends on their true types.

Given the agents' true type profile θ, their reported type profile $\hat{\theta}$ and the allocation mechanism (π, x), agent i's expected **utility** is quasilinear and defined as:

$$u_i(\theta_i, \pi(\hat{\theta}), x_i(\hat{\theta}), p^{\pi(\hat{\theta})}) = v_i(\pi(\hat{\theta}), p^{\pi(\hat{\theta})}) - x_i(\hat{\theta}),$$

where $p^{\pi(\hat{\theta})} = (p_i^{\pi(\hat{\theta})})_{i \in N}$ is agents' true PoS profile for task $\pi(\hat{\theta})$ and $\hat{p}^{\pi(\hat{\theta})} = (\hat{p}_i^{\pi(\hat{\theta})})_{i \in N}$ is what they have reported.

Definition 2. *Mechanism (π, x) is **individually rational** if for all $i \in N$, for all $\theta \in \Theta$, for all $\hat{\theta}_{-i} \in \Theta_{-i}$, $u_i(\theta_i, \pi(\theta_i, \hat{\theta}_{-i}), x_i(\theta_i, \hat{\theta}_{-i}), p^{\pi(\theta_i, \hat{\theta}_{-i})}) \geq 0$.*

That is, an agent never receives a negative expected utility in an individually rational mechanism if she reports truthfully, no matter what others report.

Furthermore, we say the mechanism is **truthful** (aka *dominant-strategy incentive-compatible*) if it always maximises an agent's expected utility if she reports her type truthfully no matter what the others report, i.e., reporting type truthfully is a dominant strategy. It has been shown that truthful and efficient mechanism is impossible to achieve in a special settings of the model [14]. Instead we focus on a weaker solution concept (but still very valid) called *ex-post truthful*, which requires that reporting truthfully maximises an agent's expected utility, if everyone else also reports truthfully (i.e., reporting truthfully is an ex-post equilibrium).

Definition 3. *Mechanism (π, x) is **ex-post truthful** if and only if for all $i \in N$, for all $\theta \in \Theta$, for all $\hat{\theta}_i \in \Theta_i$, we have $u_i(\theta_i, \pi(\theta_i, \theta_{-i}), x_i(\theta_i, \theta_{-i}), p^{\pi(\theta_i, \theta_{-i})}) \geq u_i(\theta_i, \pi(\hat{\theta}_i, \theta_{-i}), x_i(\hat{\theta}_i, \theta_{-i}), p^{\pi(\hat{\theta}_i, \theta_{-i})})$.*

2.2 Failure of the Groves Mechanism

The Groves mechanism is a well-known class of mechanisms that are efficient and truthful in many domains [5]. However, they are not directly applicable in our domain due to the interdependent valuations created by the execution uncertainty. As we will see later, a simply variation of the Groves mechanism can solve the problem. In the following, we briefly introduce the Groves mechanism and show why it cannot be directly applied.

Given agents' type report profile θ, Groves mechanisms compute an efficient allocation $\pi^*(\theta)$ (π^* denotes the efficient allocation choice function) and charge each agent i

$$x_i^{Groves}(\theta) = h_i(\theta_{-i}) - V_{-i}(\theta, \pi^*) \tag{1}$$

where

- h_i is a function that only depends on θ_{-i},
- $V_{-i}(\theta, \pi^*) = \sum_{j \neq i} v_j(\pi^*(\theta), p^{\pi^*(\theta)})$ is the social welfare for all agents, excluding i, under the efficient allocation $\pi^*(\theta)$.

Since h_i is independent of i's report, we can set $h_i(\theta_{-i}) = 0$, and then each agent's utility is $v_i(\pi^*(\theta)) + V_{-i}(\theta, \pi^*)$, which is the social welfare of the efficient allocation. The following example shows that the Groves mechanism is not directly applicable in our task allocation setting.

Take the example from Figure 1 with the setting from Table 1. If both 1 and 2 report truthfully, the efficient allocation is τ' with social welfare 0.5 (which is also their utility if $h_i(\theta_{-i}) = 0$). Now if 1 misreported $\hat{p}_1^{\tau} > 0.5$, then the efficient allocation will be τ with social welfare $\hat{p}_1^{\tau} > 0.5$, i.e., 1 can misreport to receive a higher utility.

3. APPLICABILITY OF PEV-BASED MECHANISMS

As shown in the last section, the Groves mechanisms are not directly applicable due to the interdependency of agents' valuations created by their probability of success (PoS). The other reason is that the Groves payment is calculated from agents' reported PoS rather than their realized/true PoS.

The fact is that we can partially verify their reported PoS by delaying their payments until they have executed their tasks (post-execution verification). In line with this, Porter et al. [14] have proposed a variation of the Groves mechanism which pays an agent according to their actual task completion, rather than what they have reported. More specifically, we define two payments for each agent: a reward for successful completion and a penalty for non-completion. Let us call this mechanism *PEV-based mechanism.*

Porter et al. [14] have considered a simple setting where there is one requester who has one or multiple tasks to be allocated to multiple workers each of whom have a fixed cost to attempt each task. Later, Ramchurn et al. [15] extended Porter et al.'s model to a multiple-requester setting (a combinatorial task exchange) and especially considered trust information which will be further studied later in this paper. Our setting generalises both models and allows any types of valuations and allocations. In what follows, we formally define the PEV-based mechanism and analyse its applicability in our general domain.

Given the agents' true type profile θ and their reports $\hat{\theta}$, let p_{-i}^{τ} be the true PoS profile of all agents except i for task τ, $p^{\tau} = (p_i^{\tau}, p_{-i}^{\tau})$, and $\hat{p}_{-i}^{\tau}, \hat{p}^{\tau}$ be the corresponding reported, PEV-based payment x^{PEV} for each agent i is defined as:

$$x_i^{PEV}(\hat{\theta}) = \begin{cases} h_i(\hat{\theta}_{-i}) - V_{-i}^{1}(\hat{\theta}, \pi^*) & \text{if } i \text{ succeeded,} \\ h_i(\hat{\theta}_{-i}) - V_{-i}^{0}(\hat{\theta}, \pi^*) & \text{if } i \text{ failed.} \end{cases} \tag{2}$$

where

- $h_i(\hat{\theta}_{-i}) = \sum_{j \in N \setminus \{i\}} \hat{v}_j(\pi^*(\hat{\theta}_{-i}), (0, \hat{p}_{-i}^{\pi^*(\hat{\theta}_{-i})}))$ is the maximum expected social welfare that the other agents can achieve without i's participation,
- $V_{-i}^{1}(\hat{\theta}, \pi^*) = \sum_{j \in N \setminus \{i\}} \hat{v}_j(\pi^*(\hat{\theta}), (1, p_{-i}^{\pi^*(\hat{\theta})}))$ is the realized expected social welfare of all agents except i under the efficient allocation $\pi^*(\hat{\theta})$ when $p_i^{\pi^*(\hat{\theta})} = 1$, i.e., i

succeeded. $V^0_{-i}(\hat{\theta}, \pi^*) = \sum_{j \in N \setminus \{i\}} \hat{v}_j(\pi^*(\hat{\theta}), (0, p_{-i}^{\pi^*(\hat{\theta})}))$ is the corresponding social welfare when $p_i^{\pi^*(\hat{\theta})} = 0$.

Note that $h_i(\hat{\theta}_{-i})$ is calculated according to what agents have reported, while $V^1_{-i}(\hat{\theta}, \pi^*), V^0_{-i}(\hat{\theta}, \pi^*)$ are based on the realization of their task completion, which is actually their true PoS as we used in the calculation. x_i^{PEV} pays/rewards agent i the social welfare increased by i if she completed her tasks, otherwise penalizes her the social welfare loss due to her failure.

Porter et al. [14] have shown that the mechanism (π^*, x^{PEV}) is ex-post truthful and individually rational if the dependencies between tasks are non-cyclical. In Theorem 1, we show that (π^*, x^{PEV}) is ex-post truthful in general if agents' valuations satisfy a multilinearity condition (Definition 4), which generalizes the non-cyclical task dependencies condition applied in [14].

Definition 4. *Valuation v_i of i is* ***multilinear in PoS*** *if for all type profiles $\theta \in \Theta$, for all allocations $\tau \in T$, for all $j \in N$, $v_i(\tau, p^\tau) = p^\tau_j \times v_i(\tau, (1, p^\tau_{-j})) + (1 - p^\tau_j) \times v_i(\tau, (0, p^\tau_{-j}))$.*

Intuitively, v_i is multilinear in PoS if all its variables but p^τ_j are held constant, v_i is a linear function of p^τ_j, which also means that agent i is risk-neutral (with respect to j's execution uncertainty). However, multilinearity in PoS does not indicate that v_i has to be a linear form of $v_i(\tau, p^\tau) = b + a_1 p^\tau_1 + ... + a_n p^\tau_n$, where b, a_i are constant (see Table 1 for example).

3.1 Multilinearity in PoS is Sufficient for Truthfulness

Theorem 1. *Mechanism (π^*, x^{PEV}) is ex-post truthful if for all $i \in N$, v_i is multilinear in PoS.*

Proof. According to the characterization of truthful mechanisms given by Proposition 9.27 from [13], we need to prove that for all $i \in N$, for all $\theta \in \Theta$:

1. $x_i^{PEV}(\theta)$ does not depend on i's report, but only on the task allocation alternatives;

2. i's utility is maximized by reporting θ_i truthfully if the others report θ_{-i} truthfully.

From the definition of x_i^{PEV} in (2), we can see that given the allocation $\pi^*(\theta)$, agent i cannot change $V^1_{-i}(\theta, \pi^*)$ and $V^0_{-i}(\theta, \pi^*)$ without changing the allocation $\pi^*(\theta)$. Therefore, x_i^{PEV} does not depend on i's report, but only on the task allocation outcome $\pi^*(\theta)$.

In what follows, we show that for each agent i, if the others report types truthfully, then i's utility is maximized by reporting her type truthfully.

Given an agent i' of type θ_i and the others' true type profile θ_{-i}, assume that i reported $\hat{\theta}_i \neq \theta_i$. For the allocation $\tau = \pi^*(\hat{\theta}_i, \theta_{-i})$, according to x_i^{PEV}, when i finally completes her tasks, i's utility is $u^1_i = v_i(\tau, (1, p^\tau_{-i})) - h_i(\theta_{-i}) + V^1_{-i}((\hat{\theta}_i, \theta_{-i}), \pi^*)$ and her utility if she fails is $u^0_i = v_i(\tau, (0, p^\tau_{-i})) - h_i(\theta_{-i}) + V^0_{-i}((\hat{\theta}_i, \theta_{-i}), \pi^*)$. Note that i's expected valuation depends on her true valuation v_i and all agents' true PoS. Therefore, i's expected utility is:

$$\begin{aligned}
& p^\tau_i \times u^1_i + (1 - p^\tau_i) \times u^0_i = \\
& \quad p^\tau_i \times v_i(\tau, (1, p^\tau_{-i})) && (3)\\
& \quad + (1 - p^\tau_i) \times v_i(\tau, (0, p^\tau_{-i})) && (4)\\
& \quad + p^\tau_i \sum_{j \in N \setminus \{i\}} v_j(\tau, (1, p^\tau_{-i})) && (5)\\
& \quad + (1 - p^\tau_i) \sum_{j \in N \setminus \{i\}} v_j(\tau, (0, p^\tau_{-i})) && (6)\\
& \quad - h_i(\theta_{-i}).
\end{aligned}$$

Since all valuations are multilinear in PoS, the sum of (3) and (4) is equal to $v_i(\tau, p^\tau)$, and the sum of (5) and (6) is $\sum_{j \in N \setminus \{i\}} v_j(\tau, p^\tau)$. Thus, the sum of (3), (4), (5) and (6) is the social welfare under allocation $\pi^*(\hat{\theta}_i, \theta_{-i})$. The social welfare is maximized when i reports truthfully because π^* maximizes social welfare (note that this is not the case when θ_{-i} is not truthfully reported). Moreover, $h_i(\theta_{-i})$ is independent of i's report and is the maximum social welfare that the others can achieve without i. Therefore, by reporting θ_i truthfully, i's utility is maximized. □

Theorem 1 shows that multilinearity in PoS is sufficient to truthfully implement (π^*, x^{PEV}) in an ex-post equilibrium (ex-post truthful), but not in dominant strategies (truthful). It has been shown in similar settings that ex-post truthfulness is the best we can achieve here [14, 15, 18, 4].

3.2 Multilinearity in PoS is also Necessary

In the above we showed that multilinearity in PoS is sufficient for (π^*, x^{PEV}) to be ex-post truthful. Here we show that the multilinearity is also necessary.

Theorem 2. *If (π^*, x^{PEV}) is ex-post truthful for all type profiles $\theta \in \Theta$, then for all $i \in N$, v_i is multilinear in PoS.*

Proof. By contradiction, assume that v_i of agent of type θ_i is not multilinear in PoS, i.e., there exist a θ_{-i}, an allocation $\tau \in T$, and a $j \in N$ (without loss of generality, assume that $j \neq i$) such that:

$$v_i(\tau, p^\tau) \neq p^\tau_j \times v_i(\tau, (1, p^\tau_{-j})) + (1 - p^\tau_j) \times v_i(\tau, (0, p^\tau_{-j})) \qquad (7)$$

Under the efficient allocation choice function π^*, it is not hard to find a type profile $\hat{\theta}_{-i}$ such that $\pi^*(\theta_i, \hat{\theta}_{-i}) = \tau$ and the PoS profile is the same between θ_{-i} and $\hat{\theta}_{-i}$. We can choose $\hat{\theta}_{-i}$ by setting $\hat{v}_j(\tau, p^\tau)$ to a sufficiently large value for each $j \neq i$.

Applying (π^*, x^{PEV}) on profile $(\theta_i, \hat{\theta}_{-i})$, when j finally successfully completes her tasks τ_j, her utility is $u^1_j = \hat{v}_j(\tau, (1, p^\tau_{-j})) - h_j((\theta_i, \hat{\theta}_{-i})_{-j}) + V^1_{-j}((\theta_i, \hat{\theta}_{-i}), \pi^*)$ and her utility if she fails is $u^0_j = \hat{v}_j(\tau, (0, p^\tau_{-j})) - h_j((\theta_i, \hat{\theta}_{-i})_{-j}) + V^0_{-j}((\theta_i, \hat{\theta}_{-i}), \pi^*)$. Thus, j's expected utility is (note that

$\hat{p}_j^\tau = p_j^\tau$):

$$
\begin{aligned}
p_j^\tau \times u_j^1 &+ (1-p_j^\tau) \times u_j^0 = \\
& p_j^\tau \times v_i(\tau, (1, p_{-j}^\tau)) \quad (8)\\
& + (1-p_j^\tau) \times v_i(\tau, (0, p_{-j}^\tau)) \quad (9)\\
& + p_j^\tau \sum_{k\in N\setminus\{i\}} \hat{v}_k(\tau, (1, p_{-j}^\tau)) \quad (10)\\
& + (1-p_j^\tau) \sum_{k\in N\setminus\{i\}} \hat{v}_k(\tau, (0, p_{-j}^\tau)) \quad (11)\\
& - h_j(\theta_{-j}).
\end{aligned}
$$

Given the assumption (7), terms (8) and (9) together can be written as $v_i(\tau, p^\tau) + \delta_i$ where $\delta_i = (8) + (9) - v_i(\tau, p^\tau)$. Similar substitutions can be carried out for all other agents $k \in N \setminus \{i\}$ in terms (10) and (11) regardless of whether v_k is mutlilinear in PoS. After this substitution, j's utility can be written as:

$$
\begin{aligned}
p_j \times u_j^1 &+ (1-p_j) \times u_j^0 = \\
& v_i(\tau, p^\tau) + \sum_{k\in N\setminus\{i\}} \hat{v}_k(\tau, p^\tau) \quad (12)\\
& + \sum_{k\in N} \delta_k \quad (13)\\
& - h_j(\theta_{-j}).
\end{aligned}
$$

Now consider a suboptimal allocation $\hat{\tau} \neq \tau$, if $\hat{\tau}$ is chosen by the mechanism, then j's utility can be written as:

$$
\begin{aligned}
\hat{u}_j &= \\
& v_i(\hat{\tau}, p^{\hat{\tau}}) + \sum_{k\in N\setminus\{i\}} \hat{v}_k(\hat{\tau}, p^{\hat{\tau}}) \quad (14)\\
& + \sum_{k\in N} \hat{\delta}_k \quad (15)\\
& - h_j(\theta_{-j}).
\end{aligned}
$$

In the above two utility representations, we know that terms (12) > (14) because π^* is efficient, but terms (13) and (15) can be any real numbers.

In what follows, we tune the valuation of j such that the optimal allocation is either τ or $\hat{\tau}$, and in either case j is incentivized to misreport.

In the extreme case where all agents except i's valuations are multilinear in PoS, we have $\delta_k = 0, \hat{\delta}_k = 0$ for all $k \neq i$ in (13) and (15). Therefore, $\sum_{k\in N} \delta_k = \delta_i \neq 0$ and $\sum_{k\in N} \hat{\delta}_k = \hat{\delta}_i$ (possibly $= 0$). It might be the case that $\delta_i = \hat{\delta}_i$, but there must exist a setting where $\delta_i \neq \hat{\delta}_i$, otherwise v_i is multilinear in PoS, because constant δ_i for any PoS does not violate the multilinearity definition.

1. If $\delta_i > \hat{\delta}_i$, we have $(12) + \delta_i > (14) + \hat{\delta}_i$. In this case, we can increase $\hat{v}_j(\hat{\tau}, p^{\hat{\tau}})$ such that $\hat{\tau}$ becomes optimal, i.e., (12) < (14), but $(12) + \delta_i > (14) + \hat{\delta}_i$ still holds. Therefore, if j's true valuation is the one that chooses $\hat{\tau}$ as the optimal allocation, then j would misreport to get allocation τ which gives her a higher utility.

2. If $\delta_i < \hat{\delta}_i$, we can easily modify $\hat{v}_j(\hat{\tau}, p^{\hat{\tau}})$ such that $(12) + \delta_i < (14) + \hat{\delta}_i$ but (12) > (14) still holds. In this case, if j's true valuation again is the one just modified, j would misreport to get allocation $\hat{\tau}$ with a better utility.

In both of the above situations, agent j is incentivized to misreport, which contradicts that (π^*, x^{PEV}) is ex-post truthful. Thus, v_i has to be multilinear in PoS. □

It is worth mentioning that Theorem 2 does not say that given a specific type profile θ, all v_i have to be multilinear in PoS for (π^*, x^{PEV}) to be ex-post truthful. Take the delivery example from Table 1 and change agent 2's valuation for τ to be $v_2(\tau, p^\tau) = (\mathbf{p_1^\tau})^{\mathbf{2}} \times p_2^\tau$ which is not multilinear in PoS. It is easy to check that under this change, no agent can gain anything by misreporting if the other agent reports truthfully. However, given each agent i of valuation v_i, to truthfully implement (π^*, x^{PEV}) in an ex-post equilibrium for all possible type profiles of the others, Theorem 2 says that v_i has to be multilinear in PoS, otherwise, there exist settings where some agent is incentivized to misreport.

3.3 Conditions for Achieving Individual Rationality

PEV-based mechanism is individually rational in Porter et al. [14]'s specific setting. However, in the general model we consider here, it may not guarantee this property. For example, there is an allocation where an agent has no task to complete in an allocation, but has a negative valuation for the completion of the tasks assigned to the others (i.e. she is penalised if the others complete their tasks). If that allocation is the optimal allocation and the allocation does not change with or without that agent, then she will get a zero payment therefore a negative utility.

Proposition 1 shows that by restricting agents' valuations to some typical constraint, PEV-based mechanism can be made individually rational. The constraint says if an agent is not involved in a task allocation (i.e., when the tasks assigned to her is empty), she will not be penalised by the completion of the others' tasks.

Proposition 1. *Mechanism (π^*, x^{PEV}) is individually rational if and only if for all $i \in N$, for all $\tau \in T$, if $\tau_i = \emptyset$, then $v_i(\tau, p^\tau) \geq 0$ for any $p^\tau \in [0,1]^N$.*

Proof. (If part) For all type profile $\theta \in \Theta$, for all $i \in N$, let $\tau = \pi^*(\theta)$ and $\hat{\tau} = \pi^*(\theta_{-i})$, i's utility is given by $\sum_{k\in N} v_k(\tau, p^\tau) - \sum_{k\in N\setminus\{i\}} v_k(\hat{\tau}, p_{-i}^{\hat{\tau}})$, where the first term is the optimal social welfare with i's participation and the second term is the optimal social welfare without i's participation. It is clear that $\hat{\tau}_i = \emptyset$ as $\hat{\tau}$ is the optimal allocation without i's participation. $\sum_{k\in N\setminus\{i\}} v_k(\hat{\tau}, p_{-i}^{\hat{\tau}}) + v_i(\hat{\tau}, p^{\hat{\tau}})$ is the social welfare for allocation $\hat{\tau}$. Since τ is optimal, we get that $\sum_{k\in N} v_k(\tau, p^\tau) \geq \sum_{k\in N\setminus\{i\}} v_k(\hat{\tau}, p_{-i}^{\hat{\tau}}) + v_i(\hat{\tau}, p^{\hat{\tau}})$. Thus, $\sum_{k\in N} v_k(\tau, p^\tau) - \sum_{k\in N\setminus\{i\}} v_k(\hat{\tau}, p_{-i}^{\hat{\tau}}) \geq v_i(\hat{\tau}, p^{\hat{\tau}}) \geq 0$, i.e. i's utility is non-negative.

(Only if part) If there exist an i of type θ_i, a τ, a $p^\tau \in [0,1]^N$ such that $\tau_i = \emptyset$ and $v_i(\tau, p^\tau) < 0$. We can always find a profile $\hat{\theta}_{-i}$ s.t. $\hat{p}^\tau = p^\tau$ and $\pi^*(\theta_i, \hat{\theta}_{-i}) = \pi^*(\hat{\theta}_{-i}) = \tau$. It is clear that the payment for i is 0 and her utility is $v_i(\tau, p^\tau) < 0$ (violates individual rationality). □

4. EXTENSION TO TRUST-BASED ENVIRONMENTS

So far, we have assumed that each agent can correctly predict her probability of success (PoS) for each task, but in some environments, an agent's PoS is not perfectly perceived by the agent alone. Instead, multiple other agents may have had prior experiences with a given agent and their experiences can be aggregated to create a more informed measure of the PoS for the given agent. This measure is termed the trust in the agent [15]. Ramchurn et al. have extended Porter et al.'s mechanism to consider agents' trust information and showed that the extension is still truthfully implementable in their settings.

Similarly, our general model can also be extended to handle the trust information by changing singleton p_i^τ to be a vector $p_i^\tau = (p_{i,1}^\tau, ..., p_{i,j}^\tau, ..., p_{i,n}^\tau)$ where $p_{i,j}^\tau$ is the probability that i believes j will complete j's tasks in τ. Agent i's aggregated/true PoS for task τ is given by a function $f_i^\tau : [0,1]^N \rightarrow [0,1]$ with input $(p_{1,i}^\tau, ..., p_{n,i}^\tau)$. Given this extension, for any type profile θ, let $\rho_i^\tau = f_i^\tau(p_{1,i}^\tau, ..., p_{n,i}^\tau)$, the social welfare of a task allocation τ is defined as:

$$\sum_{i \in N} v_i(\tau, \rho^\tau) \tag{16}$$

where $\rho^\tau = (\rho_1^\tau, ..., \rho_n^\tau)$.

As shown in [15], the PEV-based mechanism can be extended to handle this trust information by simply updating the efficient allocation choice function π^* with the social welfare calculation given by Equation (16). Let us call the extended mechanism $\mathcal{M}^{trust}$. Ramchurn et al. have demonstrated that $\mathcal{M}^{trust}$ is ex-post truthful in their settings when the PoS aggregation function is the following linear form:

$$f_i^\tau(p_{1,i}^\tau, ..., p_{n,i}^\tau) = \sum_{j \in N} \omega_j \times p_{j,i}^\tau \tag{17}$$

where constant $\omega_j \in [0,1]$ and $\sum_{j \in N} \omega_j = 1$.

Following the results in Theorems 1 and 2, we generalize Ramchurn et al.'s results to characterize all aggregation forms under which $\mathcal{M}^{trust}$ is ex-post truthful.

Definition 5. *A PoS aggregation $f_i = (f_i^\tau)_{\tau \in T}$ is* ***multilinear*** *if for all $j \in N$, for all $\tau \in T$, for all $\theta \in \Theta$, $f_i^\tau(p_{1,i}^\tau, ..., p_{j,i}^\tau, ..., p_{n,i}^\tau) = p_{j,i}^\tau \times f_i^\tau(p_{1,i}^\tau, ..., p_{j-1,i}^\tau, 1, p_{j+1,i}^\tau, ..., p_{n,i}^\tau) + (1 - p_{j,i}^\tau) \times f_i^\tau(p_{1,i}^\tau, ..., p_{j-1,i}^\tau, 0, p_{j+1,i}^\tau, ..., p_{n,i}^\tau)$.*

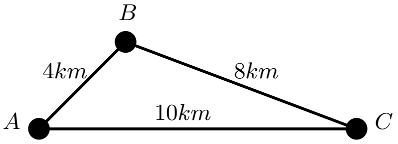
Definition 5 is similar to the multilinear in PoS definition given by Definition 4. Multilinear aggregations cover the linear form given by Equation (17), but also consist of many non-linear forms such as $\prod_{j \in N} p_{j,i}^\tau$. The following corollary directly follows Theorems 1 and 2. We omit the proof here. The basic idea of the proof is that given a multilinear function, if we substitute another multilinear function (with no shared variables) for one variable of the function, then the new function must be multilinear.

Corollary 1. *Trust-based mechanism $\mathcal{M}^{trust}$ is ex-post truthful if and only if for all $i \in N$, v_i is multilinear in PoS, and the PoS aggregation f_i is multilinear.*

For $\mathcal{M}^{trust}$ to be individually rational, the constraint specified in Proposition 1 is still sufficient and necessary, if we change h_{-i} in the payment definition (Equation (2)) to be the optimal social welfare that the others can achieve without i, but assume that i offered the worst trust in the others (see [15] for more details).

5. APPLYING PEV TO RIDESHARING

The literature has studied a few task allocation applications with execution uncertainty such as scheduling computational tasks to computing service providers or computing machines [18, 4]. In this section, we consider a ridesharing application such as *Uber* and *Lyft* in the increasingly important sharing economy, where the valuation setting has not been considered with execution uncertainty yet. The principle of ridesharing is to efficiently utilise the extra seats available in commuters' cars and therefore to reduce the number of cars on the road, travel costs and CO_2 emissions. However, existing ridesharing services such as *Uber* mainly allow professional drivers to share their cars, and these services are essentially not different from taxi services and even pull out more cars on the road (create more congestions).



Bob: $(A \rightarrow C, drive, cost: 15)$
Tom: $(A \rightarrow C, drive, cost: 10)$
Alice: $(B \rightarrow C, taxi, cost: 20)$

Allocation space T:
$\tau^1 = (\{Bob\}, \{Tom\}, \{Alice\})$
$\tau^2 = (\{Bob(drive), Tom\}, \{Alice\})$
$\tau^3 = (\{Bob, Tom(drive)\}, \{Alice\})$
$\tau^4 = (\{Bob\}, \{Tom, Alice\})$
$\tau^5 = (\{Bob, Alice\}, \{Tom\})$

Figure 2: Ridesharing example

To incentivise the whole community, not just professional drivers, to efficiently share their cars, it is essential to sufficiently evaluate the quality of the sharing services offered by individuals, and to establish a proper trust mechanism in the community[2]. Execution uncertainty is likely to have a strong impact on service quality. Following the above theoretical results, in what follows, we show how execution uncertainty can be modelled to increase service quality in ridesharing and how trust can be further incorporated.

We consider a ridesharing setting where there are multiple commuters, each of whom has a trip to finish via driving or taking other form of transport. Each commuter's trip is modelled by departure/arrival locations, travel time window, travel costs and etc (see [19] for an example). For the sake of clarity, we consider a simple ridesharing setting given in Figure 2. There are three commuters, Bob, Tom and Alice, who are planning a travel to attend a party at location C. Both Bob and Tom depart from location A and plan to drive with different costs, while Alice departs from location B and plans to take a taxi with a cost of 20 (the costs are based on the shortest route to finish their trips). Assume that they are flexible about their travel time, but all of them have to arrive at location C before the party starts. We further assume that both Bob and Tom have one extra seat in their cars that can be shared to others. We then get the sharing allocation space T presented in Figure 2 (people in one set share(s) a car).

For each task allocation $\tau \in T$, commuter i's probability of success (PoS) p_i^τ is the probability that i will finish the

[2] For example, *Maramoja* provides ridesharing/taxi services in Nairobi and relies on trust metrics to guarantee riders/drivers successful trips.

	v_b	v_t	v_a
τ^1	0	0	0
τ^2	$-\alpha \times p_b \times p_t$	$10 \times p_b \times p_t$	0
τ^3	$(15 - \alpha) \times p_b \times p_t$	0	0
τ^4	$-\beta \times p_t \times p_a$	$(-2 + \gamma \times p_a) \times p_t$	$20 \times p_a \times p_t$
τ^5	$(-3 + \delta \times p_a) \times p_b$	0	$20 \times p_a \times p_b$

Table 2: A valuation setting for the exmaple in Figure 2, where parameters $\alpha, \beta, \gamma, \delta > 0$.

	v_b	v_t	v_a
τ^1	0	0	0
τ^2	$-\alpha$	10	0
τ^3	$15 - \alpha$	0	0
τ^4	$-\beta$	$-2 + \gamma$	20
τ^5	$-3 + \delta$	0	20

Table 3: A valuation setting of Table 2 given $p_b = p_t = p_a = 1$.

trip and attend the party (without sharing). It is natural to assume that a commuter's intention to attend the party is the same among all sharing allocations, and let p_b, p_t and p_a be the PoS of Bob, Tom and Alice respectively.

We define their valuations v_b, v_t, v_a for Bob, Tom and Alice respectively according to Table 2. The basic intuition behind the valuation setting is that if a commuter takes a ride offered by others, then she will save her original trip cost (i.e., her valuation is positive), while if she offers a ride to others, then she will incur a detour cost if there is any (i.e., her valuation is negative). The detour costs for Bob and Tom to offer a ride to Alice are 15×0.2 and 10×0.2 respectively (as they travel 20% more than their shortest trip). In addition, commuters may have externalities. In this example, Bob has a higher preference to offer a ride to Alice than to Tom, and would prefer that Tom and Alice do not travel together (because, e.g., Bob is afraid of that Alice might discuss their joint business ideas with Tom). Bob gets a positive value δ if he shares with Alice, while receives a negative value $-\alpha$ to share with Tom. Bob also gets a negative value $-\beta$ if Tom shares with Alice. Tom also prefers Alice to Bob, but Alice has no preference between Bob and Tom. This valuation setting makes previous work insufficient, because there are externalities and there is no cost for a commuter to not execute her trip.

Without loss of generality, assume that Bob, Tom and Alice are all certain about attending the party, i.e., $p_b = p_t = p_a = 1$. Then, their valuations are simplified to be those in Table 3. Assume that τ^5 is the only efficient allocation and τ^4 is the second best, i.e., $17 + \delta > 18 - \beta + \gamma > 15 - \alpha$. Applying the PEV-based mechanism, the payment for Bob is the following (according to Equation (2)):

$$x_b^{PEV} = \begin{cases} (18 + \gamma) - 20 & \text{if Bob attended,} \\ (18 + \gamma) - 0 & \text{if Bob didn't attend.} \end{cases}$$

where $18 + \gamma$ is the social welfare that Tom and Alice can achieve without Bob (allocation τ^4), and 20 is the cost that Bob saved for Alice (allocation τ^5) if Bob attended the party (i.e., Bob finished his trip). The payment indicates that Bob will receive a reward $-(\gamma - 2)$ if he attends the party, while he will pay a penalty of $18 + \delta$ if he fails to attend. Thus, Bob's utility is $u_b = ((-3 + \delta) - (\gamma - 2)) \times p_b + (0 - (18 + \gamma)) \times (1 - p_b) = \delta - \gamma - 1$. Notice that Bob's utility might be negative because he is unhappy if Tom and Alice share together (see Proposition 1 for more details).

In what follows, we further consider how trust between Bob, Tom and Alice can be incorporated in the above ridesharing allocation. Apart from the probability of success p_b, p_t, p_a they have predicated for themselves, each of them also holds a measure of trust about how likely the others will attend the party (according to their past social experiences with each other). Then each commuter's final probability of attending the party is aggregated from all commuters' predications (see Section 4 for the detailed model). We have assumed that all of them are certain about their own trips, i.e., $p_b = p_t = p_a = 1$. However, Tom and Alice do not trust Bob, i.e., Tom and Alice believe that Bob may not attend the party in the end according to their experiences. If Bob's true probability of attending the party is averaged from all commuters' opinion, then Bob's probability of attending the party is $\hat{p}_b < 1$. Assume the other trust values are all one, i.e., the aggregated PoSs of Tom and Alice are both one. Then the social welfare of allocation τ^5 under trust is $(17 + \delta)\hat{p}_b$. If $(17 + \delta)\hat{p}_b$ is less than $18 - \beta + \gamma$ (the social welfare of allocation τ^4), Bob will not be allocated to share with Alice, which is different from what we had without considering trust in the above.

It is worth mentioning that in a complex ridesharing setting, a driver may take more than one rider and a rider may transfer between drivers to reach her destination. It is often quite challenging to compute an efficient allocation in a complex setting with many commuters, even without considering execution uncertainty and trust. To tackle this computational challenge, we may limit the allocation space by, for example, limiting the number of people can share in one car (e.g., *UberPool* only considers at most two passengers to share an Uber taxi) and limiting the number of transfers for a rider to get to her destination (e.g., no existing ridesharing service has considered transfer/multi-hop). Searching an optimal allocation under the limited allocation space will be a lot easier and more importantly the truthfulness property still holds [12].

Apart from ridesharing, there are many other sharing economy business models where execution uncertainty is unavoidable. For instance, in the knowledge sharing sector such as *Freelancer.com*, freelancers cannot always guarantee the completion of the tasks assigned to them and therefore rating/trust mechanism has been established to help employers to find the right freelancers for their tasks. In *Freelancer.com*, when an employer posts tasks, many freelancers will bid for the tasks with different completion speeds and costs. Some freelancers have good experiences (e.g., high completion rate and high employer rating) and some do not. Therefore, the employer faces the challenge of finding the best set of freelancers among all the bidders, which fits perfectly to the problem setting of this paper.

6. WHEN PEV IS NOT AVAILABLE

We have characterised and bounded the applicability of the post-execution verification (PEV)-based mechanism and its extension with trust in a general task allocation setting, and have further demonstrated its applicability in the real-world. The key feature of the PEV-based mechanism is that the mechanism can verify whether an agent has successfully completed the tasks assigned to her and pays her accordingly, which is feasible in many task allocation domains.

However, when post-execution verification is not available, the problem that arises requires the design of an efficient mechanism in a general interdependent valuation setting, where the interdependence comes from agents' execution uncertainty. Under a general interdependent valuation setting, Jehiel and Moldovanu [6] have demonstrated the difficulty of designing an efficient and Bayes-Nash truthful[3] mechanism (see also [9, 3]). They have proved a general impossibility and further identified a necessary condition for implementing an efficient and Bayes-Nash truthful mechanism (see Proposition 2). In the rest, we show how to model the task allocation problem as a general interdependent valuation setting and draw the implication of the necessary condition in the task allocation domain.

We first briefly introduce the general interdependent valuation setting based on [6, 3] and then show how to fit the task allocation problem to the general setting. The general interdependent valuation setting is the following:

- There are K social alternatives/outcomes and N agents.
- Each agent i has a (private) type (called *signal*) s_i drawn from a space $S_i \subseteq \mathbb{R}^{K\times N}$ according to a continuous density function $f_i(s_i) > 0$ and f_i is common knowledge. Coordinate $s^k_{i,j}$ of s_i influences the valuation of agent j in alternative k.
- One alternative k will be chosen, and i's valuation for k is defined as:

$$v^k_i(s^k_{1,i}, ..., s^k_{n,i}) = \sum_{j\in N} \alpha^k_{j,i} \times s^k_{j,i} \tag{18}$$

  where parameters $\alpha^k_{j,i} \geq 0$ are common knowledge.

It is evident that the interdependence between agents' valuations is caused by their private signals s_i, which is their probability of success in our task allocation setting.

The task allocation problem can be modelled as the above with the following setup:

- Let $K = T$, for all $k \in K$, $s^k_{i,j} = p^k_i$ for all $j \in N$, where p^k_i (i's PoS for k) is drawn from $[0, 1]$ with a density function f[4]. That is, there is only one signal from i for each k, which is i's PoS for k.
- Applying (18), i's valuation for allocation k is a linear function of all PoSs p^k_j for all $j \in N$, and parameter $\alpha^k_{i,j} \geq 0$ represents the value that j will get if i completes her tasks k_i in k.

We can see that the above model can only model a very small portion of our general task allocation, namely the settings in which the tasks between agents are independent (due to the linear form of their valuation functions) and also their valuations become public[5] (because parameters $\alpha^k_{i,j} \geq 0$ are public and not part of their type).

Even under this limited task allocation setting, Jehiel and Moldovanu [6] have showed that the following condition (Equation (19)) has to hold for implementing an efficient and Bayes-Nash truthful mechanism. We will discuss the implication of the necessary condition in the task allocation setting.

Proposition 2. *[Theorem 4.3 in [6]] If there exists an efficient and Bayes-Nash truthful mechanism, then the following must hold:*

$$\frac{\alpha^{\hat{k}}_{i,i}}{\alpha^k_{i,i}} = \frac{\sum_{j\in N} \alpha^{\hat{k}}_{i,j}}{\sum_{j\in N} \alpha^k_{i,j}} \tag{19}$$

for all i of type s_i, for all $k \neq \hat{k} \in K$, if $\alpha^k_{i,i} \neq 0$ and there exist $s_{-i} \neq \hat{s}_{-i}$ such that $\pi^*(s_i, s_{-i}) = k$ and $\pi^*(s_i, \hat{s}_{-i}) = \hat{k}$ where π^* is an efficient allocation.

In our model, condition (19) says that the ratio of the values that i gets from the completion of her own tasks in two different allocations $k, \hat{k}$ is equal to the ratio of the values that i brings to all agents including herself. In other words, if i prefers tasks k_i to $\hat{k}_i$, then all agents together also prefer k_i to $\hat{k}_i$ for i.

However, condition (19) is just one necessary condition and we are not aware of any efficient and Bayes-Nash truthful mechanism existing for the general interdependent valuation setting (except for a very simple setting where agents' signals are one-dimensional [6], in which case the efficient and truthful mechanism actually follows the VCG logic).

From the above analysis, we see that without post-execution verification, we can hardly design a truthful mechanism and the necessary condition for designing a truthful mechanism already restricts us to a very small portion of the whole valuation space. With post-execution verification, we are not restricted by the necessary condition (19). Our results showed that the PEV-based mechanism is ex-post truthful in the above general setting without any other condition, because the valuation setting (Equation (18)) is risk-neutral (multilinear in PoS) by definition.

7. CONCLUSIONS

We studied a general task allocation problem where multiple agents collaboratively accomplish a set of tasks, but they may fail to successfully complete tasks assigned to them. To design an efficient task allocation mechanism for this problem, we showed that post-execution verification based mechanism is truthfully implementable, if and only if all agents are risk-neutral with respect to their execution uncertainty. We also showed that trust information between agents can be integrated into the mechanism without violating its properties, if and only if the trust information is aggregated by a multilinear function. It is the first time that this characterisation and bound of the applicability of the post-execution verification based mechanism is studied. We further demonstrated the applicability of the mechanism in the real-world and showed the significance of the post-execution verification in the design. For future work, apart from investigating other interesting applications of the PEV-based mechanism, the results also indicate that we need to search different solutions to handle settings where agents are not risk-neutral (e.g., in ridesharing, a rider may not want to ride with a driver who has high execution uncertainty for important trips).

[3] *Bayes-Nash truthful* is weaker than ex-post truthful and it assumes that all agents know the correct probabilistic distribution of each agent's type.

[4] We can set f to be any random distribution, which is not important in our model.

[5] Public valuation does not affect our results because the main challenge in our setting is that their private PoS creates valuation interdependencies.

REFERENCES

[1] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia. A view of cloud computing. *Commun. ACM*, 53(4):50–58, Apr. 2010.

[2] R. Belk. You are what you can access: Sharing and collaborative consumption online. *Journal of Business Research*, 67(8):1595 – 1600, 2014.

[3] T. Börgers. *An introduction to the theory of mechanism design*. 2015.

[4] V. Conitzer and A. Vidali. Mechanism design for scheduling with uncertain execution time. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, pages 623–629. AAAI Press, 2014.

[5] T. Groves. Incentives in Teams. *Econometrica*, 41(4):617–31, July 1973.

[6] P. Jehiel and B. Moldovanu. Efficient Design with Interdependent Valuations. *Econometrica*, 69(5):1237–59, September 2001.

[7] P. Jehiel, B. Moldovanu, and E. Stacchetti. Multidimensional Mechanism Design for Auctions with Externalities. *Journal of Economic Theory*, 85(2):258–293, April 1999.

[8] F. Maniquet and Y. Sprumont. Sharing the cost of a public good: An incentive-constrained axiomatic approach. *Games and Economic Behavior*, 68(1):275 – 302, 2010.

[9] P. R. Milgrom and R. J. Weber. A Theory of Auctions and Competitive Bidding. *Econometrica*, 50(5):1089–1122, September 1982.

[10] R. B. Myerson. revelation principle. In S. N. Durlauf and L. E. Blume, editors, *The New Palgrave Dictionary of Economics*. Palgrave Macmillan, Basingstoke, 2008.

[11] N. Nisan and A. Ronen. Algorithmic mechanism design. *Games and Economic Behavior*, 35(1âĂŞ2):166 – 196, 2001.

[12] N. Nisan and A. Ronen. Computationally feasible vcg mechanisms. *J. Artif. Int. Res.*, 29(1):19–47, May 2007.

[13] N. Nisan, T. Roughgarden, Éva Tardos, and V. V. Vazirani. *Algorithmic Game Theory*. Cambridge University Press, 2007.

[14] R. Porter, A. Ronen, Y. Shoham, and M. Tennenholtz. Fault tolerant mechanism design. *Artif. Intell.*, 172(15):1783–1799, Oct. 2008.

[15] S. D. Ramchurn, C. Mezzetti, A. Giovannucci, J. A. Rodriguez-Aguilar, R. K. Dash, and N. R. Jennings. Trust-based mechanisms for robust and efficient task allocation in the presence of execution uncertainty. *J. Artif. Int. Res.*, 35(1):119–159, June 2009.

[16] T. Roughgarden. Routing games. In *Algorithmic Game Theory*. Cambridge University Press, 2007.

[17] T. Sandholm. An implementation of the contract net protocol based on marginal cost calculations. In *Proceedings of the Eleventh National Conference on Artificial Intelligence*, AAAI'93, pages 256–262. AAAI Press, 1993.

[18] S. Stein, E. Gerding, A. Rogers, K. Larson, and N. Jennings. Algorithms and mechanisms for procuring services with uncertain durations using redundancy. *Artificial Intelligence*, 175(14-15):2021–2060, 2011.

[19] D. Zhao, D. Zhang, E. H. Gerding, Y. Sakurai, and M. Yokoo. Incentives in ridesharing with deficit control. In *Proceedings of the 2014 International Conference on Autonomous Agents and Multi-agent Systems*, AAMAS '14, pages 1021–1028. International Foundation for Autonomous Agents and Multiagent Systems, 2014.