Optimal Multi-unit Allocation with Costly Verification*

Geoffrey A. Chua[†] Gaoji Hu[‡] Fang Liu[§]

February 2, 2021

Abstract

A principal has n homogeneous objects to allocate to $I > n$ agents. The principal can allocate at most one good to an agent and each agent values the good. Agents have private information about the principal's payoff of allocating the goods. There are no monetary transfers but the principal can costly check any agent's value. We characterize the mechanism which maximizes the principal's net expected payoff. Such an optimal mechanism is easily implementable by a dynamic game which has an equilibrium in obviously dominant strategies. We also compare the optimal mechanism with an alternative mechanism that allocates the goods one by one, where the single-good optimal mechanism is used in each step. The optimal mechanism dominates such an alternative in a particular way: Under any value profile, the two mechanisms allocate the goods to the same set of agents, but the optimal mechanism checks agents less frequently.

Keywords: mechanism design, costly verification, multiple goods

*We are grateful to Jiangtao Li, Bart Lipman, Joel Sobel, Satoru Takahashi, Xi Weng and Yongchao Zhang for their helpful comments and suggestions. This research was supported in part by Singapore Ministry of Education Academic Research Fund MOE2015-T2-2-046. Hu also acknowledges the sponsorship by Shanghai Pujiang Program (Grant No. 2020PJC052). All remaining errors are our own.

[†]Nanyang Business School, Nanyang Technological University. gbachua@ntu.edu.sg

[‡]School of Economics, Shanghai University of Finance and Economics. hugaoji@sufe.edu.cn

[§]Academy of Mathematics and Systems Science, Chinese Academy of Sciences. kliu@amt.ac.cn

1 Introduction

Consider a principal who allocates n identical goods to $I > n$ agents. The principal can allocate at most one good to an agent and the agents strictly prefer to receive a good than not receiving one. Each agent i has private information about the value to the principal, denoted by t_i, if he receives the good. The principal cannot incentivize the agents to report their private information through monetary transfers; however, she is able to check the value of each agent at a cost c_i that may vary across agents. Many examples from various industries and organizations fit the environment described above. A humanitarian organization, such as IFRC (International Federation of Red Cross and Red Crescent Societies), may need to decide which region to support after a disaster, e.g. in which town(s) to rebuild a hospital or a primary school. The ministry of education may need to decide to which research teams to allocate the research funds such as the Academic Research Fund (AcRF) in Singapore. A dean may have several job openings to allocate to different departments in the university. A chief procurement officer (CPO) and his team may decide which suppliers to invest for sustainability programs.

In this paper, we construct a class of mechanisms, the n-ascending mechanisms (n-ASMs), that maximize the expected net value to the principal, that is, the total value from allocating the goods minus the checking costs. Such an optimal mechanism is a hybrid between two extremes. One extreme allocates the goods efficiently, i.e., to the n agents who have the highest values, which requires agents being checked and leads to very high checking cost; the other incurs no checking cost by allocating the goods without learning agents' value, which results in very large allocation inefficiency. The n-ASMs balance the benefit of efficient allocation and the cost of checking agents. Moreover, the trade-off between the two extremes is determined ex post, in response to each valuation profile $\mathbf{t} = (t_1, \ldots, t_I)$. See Theorem 1 for more details.

More precisely, an n-ASM specifies n agents, e.g. $i = 1, \ldots, n$, and a threshold v_i^* for each agent.[1] These agents are referred to as the favored agents, and they are "endowed" with the goods. The other agents are the free agents. The n-ASM specifies for each valuation profile an allocation rule and a checking policy that can be implemented by at most n steps. In step 1, if no free agent reports a net value $t_i - c_i$ higher than v_n^*, then allocate all goods to the first n agents without checking and the allocation process ends. Otherwise, check the agent with the highest reported net value among the free agents and agent n, and allocate a good to that agent. Under equilibrium, all agents report truthfully and hence the allocation rules are consistent with the checking outcomes. We remove the agent who has received the good, and the system is left with $I - 1$ agents and $n - 1$ goods. Agent n becomes a

[1] The agents $i = 1, \ldots, n$ may not be the default first n agents, i.e., we may relabel agents according to their value distributions. See Section 3.2.

free agent if he is not removed. Then, we proceed to step 2 which is very similar to step 1 but with the threshold v^*_{n-1} and the updated set of free agents. In Section 3.3, we argue that the process, as a dynamic game, has an equilibrium in obviously dominant strategies (Li, 2017).

We next compare the n-ASM to an alternative mechanism which we term the n-descending mechanism (n-DSM). The n-DSM allocates the goods one by one according to the optimal single-good favored-agent mechanism in Ben-Porath et al. (2014). Since the n-ASM is optimal, it certainly delivers a weakly higher payoff to the principal than the n-DSM. Nevertheless, our comparison says that the n-ASM not only dominates the n-DSM, but also dominates it in a particular way: Under any type profile, the two mechanisms allocate the n goods to the same set of agents, while the checking cost incurred in the n-ASM is weakly smaller than that in the n-DSM. See Theorem 2 for more details. We also show that under very general conditions, the n-ASM is strictly superior than the n-DSM. See Theorem 3.

The rest of this section reviews the literature. Section 2 introduces the model. Section 3 presents our result for the n-ASM. Section 4 compares the n-ASM with the n-DSM. Section 5 concludes.

The Related Literature

Townsend (1979) initiated the literature on the principal-agent model with costly state verification.[2] Recently, Ben-Porath et al. (2014) extends the costly state verification framework to allow for multiple agents when monetary transfers are not possible.[3] Particularly, they study the allocation of one good to multiple agents, which is a special case of our paper. Thus, our optimal mechanism, the n-ASM, reduces to their optimal favored-agent mechanism when $n = 1$.

In the multi-unit setting, our paper is most closely related to Ben-Porath et al. (2019), which examines mechanism design with evidence in a very general environment.[4] The evidence model differs significantly from the costly-verification model. However, Ben-Porath et al. (2019) show that a large class of costly-verification models can be studied using the equilibrium analysis of an associated model of evidence. More precisely, their approach can be used to solve the multi-unit allocation problem as we do but in a model with finitely many type profiles.[5] It can also be used to characterize optimal mechanisms for the model of Erlanson and Kleiner (2020), which studies costly verification in collective decisions. We draw more detailed connections with Ben-Porath et al. (2019) in what follows.

First of all, focusing on the costly-verification aspect, there is only one modeling difference between

[2] See also Gale and Hellwig (1985), Border and Sobel (1987), and Mookherjee and Png (1989). Our work is also related to, among others, Glazer and Rubinstein (2004, 2006), Green and Laffont (1986), Bull and Watson (2007), Deneckere and Severinov (2008), Ben-Porath and Lipman (2012), Kartik and Tercieux (2012), Sher and Vohra (2015) and Doval (2018) in the same sense as Ben-Porath et al. (2014); we refer the readers to their paper.

[3] See also Lipman (2015) and Erlanson and Kleiner (2019).

[4] See Dye (1985) and Jung and Kwon (1988) for earlier work on the evidence models.

[5] See Section 3.2 and Appendix C of Ben-Porath et al. (2019) for how their approach works when there are only finitely many types. Our approach for the continuous-type model differs significantly from theirs. See Appendix B.

two papers: Ben-Porath et al. (2019) assume that each agent's value distribution is over finitely many types, while we assume that each agent's value distribution has an interval support. In the discrete-type setting, Ben-Porath et al. (2019) build the mathematical link between the costly-verification model and the evidence model by changing variables. Using their results in the evidence model, one can easily derive an optimal allocation rule for the costly-verification model, which in turn can induce an optimal (reduced-form) checking policy. Unsurprisingly, their allocation rule and (reduced-form) checking policy have exactly the same formats as those of our n-ASM. In this sense, our paper serves as an extension of Ben-Porath et al. (2019), from the discrete-type setting to the continuous-type setting. Moreover, our analysis complements theirs in specifying the *ex post*, instead of the reduced-form, checking policies.[6]

Second, we provide an easy implementation of the optimal allocation mechanisms in both the continuous-type setting and the discrete-type setting of Ben-Porath et al. (2019). Namely, to implement an optimal mechanism, the principal can first endow the n favored agents each with a good, and then allow those agents who do not have a good to challenge a favored agent who is still holding an endowment. At each round of challenging, the principal only checks the agent who reports the highest net value among those agents involved.

The third contribution of our paper is that we provide a detailed comparison between the optimal n-ASM and the intuitive n-DSM. Ben-Porath et al. (2019) have mentioned the n-DSM as an optimal mechanism for the multi-unit allocation problem with evidence, but it is not necessarily optimal in the costly-verification model due to the modeling differences.

The literature on state verification grows rapidly in recent years. In addition to those discussed above, Mylovanov and Zapechelnyuk (2017) study the allocation of an indivisible good with ex-post verification and limited punishment. More precisely, in Ben-Porath et al. (2014) as well as the current paper, once the agents are found lying, the principal can punish them by allocating nothing to them. This is seen as unlimited punishment. In contrast, agents in Mylovanov and Zapechelnyuk (2017) may still get some benefit even if they are found lying. Li (2020) investigates a single-good allocation problem as Ben-Porath et al. (2014), but with limited punishment as in Mylovanov and Zapechelnyuk (2017).[7] Erlanson and Kleiner (2020) study how a principal should optimally choose between implementing a new policy and maintaining the status quo when information relevant for the decision is privately held by agents. Again, there is no monetary transfers but costly verification is possible. Epitropou and Vohra (2019) consider a single-good allocation problem where agents arrive on-line. It is a dynamic mechanism design problem where the decision to allocate the good to an agent must be made upon his arrival and is irreversible.

[6]See Section 3.2 for more discussions.

[7]See also Li (2021) for the allocation of goods to financially constrained agents.

2 The Model

Preliminaries. The set of agents is $\mathcal{I} = \{1, 2, \ldots, I\}$. There are n identical indivisible goods to be allocated among the agents, where $1 \leq n \leq I - 1$. Each agent can receive at most one good. The value to the principal of assigning one good to agent i depends on the information which is known only to i. Formally, that value is referred as agent i's type, denoted by t_i, and it is assumed to be private information of i. We normalize so that types are always nonnegative and the value to the principal of assigning the object to no one is zero. We assume that the t_is are independently distributed. The distribution of t_i has a strictly positive density f_i over the interval $T_i = [\underline{t}_i, \bar{t}_i]$ where $0 \leq \underline{t}_i < \bar{t}_i < \infty$. We use F_i to denote the corresponding distribution function and F to denote the joint distribution. Let $\mathbf{T} = \prod_i T_i$ be the set of type profiles.

The principal can *check* the type of agent i at a cost $c_i > 0$. We interpret checking as obtaining information (e.g., by requesting documentation, interviewing the agent, or hiring outside evaluators) which perfectly reveals the type of the agent being checked. The cost to the agent who provides information is assumed to be zero. We assume that every agent strictly prefers receiving one good to receiving nothing. Consequently, we can take the payoff to an agent to be the probability with which he receives one good. The intensity of the agents' preferences plays no role in the analysis, so these intensities may or may not be related to the types. We also assume that each agent's reservation utility is less than or equal to his utility from not receiving the good. Since monetary transfers are not allowed, not receiving the good delivers the worst payoff to an agent. Consequently, individual rationality constraints do not bind and so are disregarded throughout.

Direct Mechanisms. A general mechanism in our setting is a game that specifies (i) how each agent may act, (ii) which agent is checked (and when), and (iii) who receives the goods. From the online appendix of Ben-Porath et al. (2014), we know that the revelation principle holds. Therefore, we restrict our attention to direct mechanisms.

Formally, a *direct mechanism* consists of (i) a checking policy that maps each reported type profile $\mathbf{t} \in \mathbf{T}$ to a profile of checking probabilities $\mathbf{q}(\mathbf{t}) := (q_1(\mathbf{t}), \ldots, q_I(\mathbf{t}))$, and (ii) an allocation rule that assigns the goods to the agents according to their reported type profile and the checking outcomes. The timeline of this game is specified as follows. First, each agent i reports a number $t_i \in T_i$. Then, the checking policy $\mathbf{q} : \mathbf{T} \mapsto [0, 1]^I$ specifies a checking probability $q_i(\mathbf{t}) \in [0, 1]$ for each i and agents are checked. Finally, the goods are allocated according to the agents' reports and checking outcomes. Specifically, the allocation depends on whether or not each agent i is checked, and if checked, whether he is truth-telling or lying. If agent i is not checked, then we denote $p_i^{nc}(\mathbf{t})$ as the probability of assigning a good to agent i conditioning on no checking. As a result, $(1 - q_i(\mathbf{t}))p_i^{nc}(\mathbf{t})$ is the probability of agent

i receiving a good without being checked.

If agent i is checked, then there may be two outcomes: truth-telling or lying. To provide the incentive for agent i to tell the truth, the probability of assigning a good to him conditioning on checking and lying has to be zero.[8] Since checking is costly, the assignment probability conditioning on checking and truth-telling is 1; otherwise, the principal can reduce the checking cost by only checking agent i if she decides to assign a good to him (conditioning on truth-telling). Therefore, $q_i(\mathbf{t})$ is the probability of checking and assigning a good to agent i. Thus, the total probability of assigning a good to agent i, assuming that he is truth-telling, is

$$p_i(\mathbf{t}) := q_i(\mathbf{t}) + (1 - q_i(\mathbf{t}))p_i^{nc}(\mathbf{t}).$$

Since $(p_i(\mathbf{t}), q_i(\mathbf{t}))$ can be derived from $(p_i^{nc}(\mathbf{t}), q_i(\mathbf{t}))$ and vice versa, we refer to a simplified direct mechanism as $(\mathbf{p}, \mathbf{q})$, where $\mathbf{p}(\mathbf{t}) = (p_1(\mathbf{t}), \ldots, p_I(\mathbf{t}))$.

The Principal's Problem. She selects $(\mathbf{p}, \mathbf{q})$ to maximize her expected net payoff.

$$\max_{\mathbf{p},\mathbf{q}} \mathbf{E}_t \left[\sum_i [p_i(\mathbf{t})t_i - q_i(\mathbf{t})c_i] \right] \tag{1}$$

$$\text{s.t. } p_i : \mathbf{T} \to [0,1],\; q_i : \mathbf{T} \to [0,1], \tag{2}$$

$$q_i(\mathbf{t}) \le p_i(\mathbf{t}), \forall \mathbf{t} \in \mathbf{T},\; \sum_i p_i(\mathbf{t}) \le n, \forall \mathbf{t} \in \mathbf{T}, \quad \text{and} \tag{3}$$

$$\mathbf{E}_{\mathbf{t}_{-i}} [p_i(t_i, \mathbf{t}_{-i})] \ge \mathbf{E}_{\mathbf{t}_{-i}} \left[p_i(t_i', \mathbf{t}_{-i}) - q_i(t_i', \mathbf{t}_{-i}) \right], \forall t_i, t_i' \in T_i, \forall i \in \mathcal{I}. \tag{4}$$

Constraints (2)-(3) are the feasibility constraints. Constraint (4) is the incentive compatibility constraint to ensure that agents always prefer truth-telling to lying.

Given a mechanism $(\mathbf{p}, \mathbf{q})$, we define $\hat{p}_i(t_i) := \mathbf{E}_{\mathbf{t}_{-i}} p_i(t_i, \mathbf{t}_{-i})$ and $\hat{q}_i(t_i) := \mathbf{E}_{\mathbf{t}_{-i}} q_i(t_i, \mathbf{t}_{-i})$. The $2I$ tuple of functions $(\hat{\mathbf{p}}, \hat{\mathbf{q}})$ is the *reduced form* of the mechanism $(\mathbf{p}, \mathbf{q})$. We say that two mechanisms $(\mathbf{p}^1, \mathbf{q}^1)$ and $(\mathbf{p}^2, \mathbf{q}^2)$ are *equivalent* if they have essentially the same reduced form, i.e., for each $i \in \mathcal{I}$, $\hat{p}_i^1(t_i) = \hat{p}_i^2(t_i)$ and $\hat{q}_i^1(t_i) = \hat{q}_i^2(t_i)$ for almost all $t_i \in T_i$.[9] It is easy to see that we can write the incentive compatibility constraints and the objective function of the principal as a function only of the reduced form of the mechanism. Hence if $(\mathbf{p}^1, \mathbf{q}^1)$ is an optimal incentive compatible mechanism, $(\mathbf{p}^2, \mathbf{q}^2)$ must be as well. In this paper, we only identify the optimal mechanism up to such equivalence.

3 Characterization of the Optimal Mechanisms

In this section, we present our first result, which characterizes the optimal mechanisms for the principal, i.e., the solutions for the problem (1)-(4). We will provide a formal definition of the n-ascending mechanism in Section 3.1. Then, in Section 3.2, we characterize the optimal mechanisms by essential randomizations over the n-ascending mechanisms. Finally, Section 3.3 discusses the implementation of

[8] See the online appendix of Ben-Porath et al. (2014) for more discussions.

[9] The word "essential" means "up to measure-zero subsets of $\mathbf{T}$." This applies throughout the paper.

the optimal mechanism in a dynamic game.

3.1 Definition of the n-Ascending Mechanisms

First of all, let us introduce a threshold v_i^* for each agent $i \in \mathcal{I}$. The threshold v_i^* is defined as a variant of the critical value t_i^* in Ben-Porath et al. (2014):

$$\mathbf{E}\left(t_i\right) = \mathbf{E}\left(\max\left\{t_i - c_i, v_i^*\right\}\right). \tag{5}$$ [10]

Notably, the threshold v_i^* is determined by the distribution F_i of agent i, which is simple in the sense that it is independent of (i) the number of agents, (ii) the number of goods, and (iii) the value distribution of other agents.

Our interpretation of v_i^* is as follows. Suppose agent i is "endowed" with one good ex ante regardless of his valuation. Suppose also that the principal is not committed to the such an allocation and, at this point, considers reallocating the good to pursue potentially higher ex post valuation. If she maintains the ex ante "endowment," her expected payoff is $\mathbf{E}(t_i)$. If she chooses to acquire ex post valuation by checking, then there will be two possible outcomes. First, the good still goes to agent i, in which case the principal obtains net value $t_i - c_i$. Second, the good goes to another agent and brings a net value, say, v to the principal. Obviously, the principal will obtain the higher net value between the two, i.e., $\max\left\{t_i - c_i, v\right\}$. Note that $\mathbf{E}\left(\max\left\{t_i - c_i, v\right\}\right)$ is increasing in v. Therefore, v_i^* is the minimal net value requested by the principal to trigger reallocation.

This interpretation of v_i^* does not depend on a particular reallocation rule. For example, the reallocation could happen between i and the agent who has the *highest* reported net value among $\mathcal{I} \setminus \{i\}$; it could also happen between i and the agent who has the *second highest* reported net value among $\mathcal{I} \setminus \{i\}$; etc.[11] When the reallocation is between agent i and the agent who has the highest reported net value among $\mathcal{I} \setminus \{i\}$, our interpretation reduces to that of Ben-Porath et al. (2014).

For notational convenience, we rank and relabel agents in the decreasing order of v_i^*, i.e., $v_1^* \geq v_2^* \geq \cdots \geq v_I^*$. The n-ascending mechanism is formally defined below, which is unique given the labelling of agents.

Definition 1. *A mechanism* $(\mathbf{p}, \mathbf{q})$ *is said to be the* n***-ascending mechanism (**n***-ASM)*** *if for all* $\mathbf{t} \in \mathbf{T}$*, the rules* $(\mathbf{p}(\mathbf{t}), \mathbf{q}(\mathbf{t}))$ *are defined by the algorithm below.*

Ascending Algorithm

Set $\mathcal{I}^n = \mathcal{I} \setminus \{1, \ldots, n-1\}$*. For* $k = n, \ldots, 1$*,*

Step k**.** *Consider two mutually exclusive cases.*

[10] Ben-Porath et al. (2014) defines a critical value t_i^* implicitly by $\mathbf{E}\left(t_i\right) = \mathbf{E}\left(\max\left\{t_i, t_i^*\right\}\right) - c_i$. It is straightforward to see that $v_i^* = t_i^* - c_i$.

[11] See Lemma 7 in Appendix B for a case where the reallocation is between agent i and the agent who has the *second highest* reported net value among $\mathcal{I} \setminus \{i\}$. For other reallocation rules that are of interest, see Step (ii) of the proof of Proposition 4 in Chua et al. (2019).

Case 1. $\max_{i \in \mathcal{I}^k \setminus \{k\}} \{t_i - c_i\} \leq v_k^*$.

Set $p_1(\mathbf{t}) = \cdots = p_k(\mathbf{t}) = 1$, $q_1(\mathbf{t}) = \cdots = q_k(\mathbf{t}) = 0$, *and* $p_i(\mathbf{t}) = q_i(\mathbf{t}) = 0$ *for all* $i \in \mathcal{I}^k \setminus \{k\}$.

Stop.

Case 2. $\max_{i \in \mathcal{I}^k \setminus \{k\}} \{t_i - c_i\} > v_k^*$.

Define $m := \min \{\arg\max_{i \in \mathcal{I}^k} \{t_i - c_i\}\}$ *and set* $p_m(\mathbf{t}) = q_m(\mathbf{t}) = 1$.[12]

If $k > 1$, *then set* $\mathcal{I}^{k-1} := \mathcal{I}^k \cup \{k-1\} \setminus \{m\}$ *and go to Step* $k-1$. *If* $k = 1$, *then set* $p_i(\mathbf{t}) = q_i(\mathbf{t}) = 0$ *for all* $i \in \mathcal{I}^k \setminus \{m\}$ *and stop.*

The above algorithm can be described by the following process. After ranking the agents in the decreasing order of v_i^*, the first n agents will be "endowed" with one good each, while the remaining agents will be free agents. If no free agent challenges the favored agents by reporting a net value (i.e., $t_i - c_i$, where $i \in \mathcal{I} \setminus \{1, \ldots, n\}$) higher than v_n^*, then allocate the goods according to the "endowment" without checking anyone and the allocation process ends. If some free agent challenges the favored agents, then find the agent in $\mathcal{I} \setminus \{1, \ldots, n-1\}$ who has the highest net value, check him (the checking outcome will be truth-telling on equilibrium path) and allocate a good to him. At the same time, let the set of free agents include agent n and exclude the agent who has obtained one good. Then, we proceed with v_{n-1}^*. This continues until all goods are allocated. (It is worth emphasizing that the Ascending Algorithm may end at any step between 1 and n, depending on agents' type profile $\mathbf{t}$.)

We provide some intuition of the n-ASM. Given the principal's objective function (1), an optimal mechanism should balance the benefit of efficient allocation and the cost of checking agents. On the one hand, allocating all units according to the "endowments" of agents results in the least checking cost but very large allocation inefficiency: the value of the reported valuations are not incorporated. On the other hand, allocating all units based on valuations, i.e., agents with the n highest values always get the goods, results in no allocation inefficiency but very high checking cost. As such, an optimal mechanism is expected to allocate some goods based on only agents' distributions while other goods based on their valuations. Take the "endowments" as a benchmark. As the Ascending Algorithm proceeds, the allocation inefficiency decreases in the number of steps but the checking costs increases. The algorithm stops at a balanced point of those two effects.

The following observation on the allocation rule of the n-ASM will be useful. It says that given any type profile $\mathbf{t} \in \mathbf{T}$, unless there are ties, the n-ASM allocates the n goods to the n agents who have the n highest values of $w_i := \max\{v_i^*, t_i - c_i\}$.

Lemma 1. *For each* $\mathbf{t} \in \mathbf{T}$ *and each* $i \in \mathcal{I}$, $p_i(\mathbf{t}) = 1$ *under the* n*-ASM if and only if*

$$|\{j \in \mathcal{I} : w_j > w_i\}| + |\{j \in \{1, \ldots, i-1\} : w_j = w_i\}| \leq n - 1. \tag{6}$$

[12]Taking the smallest index is only for the tie-breaking purpose.

3.2 The Characterization Theorem

Theorem 1 below is our first result, which is a full characterization the optimal mechanisms for the multi-unit allocation problem, if not unique. One reason for the existence of multiple optimal mechanisms is that whenever a mechanism is optimal, any mechanism that is equivalent to it must be optimal as well. A second reason is that the v_i^*'s may be tied such that, say, $v_1^* = v_2^*$, in which case we have at least two optimal mechanisms according to different labelling of agents. A third reason is that, since the objective function and the constraints (1)-(4) are all linear in $\mathbf{p}$ and $\mathbf{q}$, any randomization over optimal mechanisms must also be optimal. It turns out that those three are the only reasons to have multiple optimal mechanisms. Theorem 1 says that the n-ASM is the unique optimal mechanism up to equivalence, up to relabelling of agents according to the decreasing order of v_i^*'s and up to randomization (hereafter, equivalence-relabelling-randomization).

Theorem 1. *A mechanism $(\mathbf{p}, \mathbf{q})$ is optimal if and only if it is essentially a randomization over the n-ascending mechanisms.*

We hasten to emphasize that the algorithm-based n-ASM has an equivalent one-shot definition. Note that the Ascending Algorithm which defines the n-ASM potentially takes n steps. However, the key information that the algorithm conveys is merely a cut-off index $k \leq n+1$ for allocation (i) with neither value-comparison nor checking or (ii) with both of them: (i) each of the agents with smaller index than k receives one good without being checked, regardless of their net values; and (ii) the $n-(k-1)$ agents in $\{k, \ldots, I\}$ who have the highest net values will each receive one good after being checked.[13] More precisely, given a type profile $\mathbf{t} \in \mathbf{T}$, the cut-off index is $k = n+1$ if $\max_{i>n}\{t_i - c_i\} \leq v_n^*$; otherwise, the cut-off index k is defined as the smallest index such that the number of agents among $\{k+1, \ldots, I\}$ who have net values larger than v_k^* is no less than $n-(k-1)$:

$$k := \min\left\{\kappa \in \mathbb{Z}_+ : |\{i : i > \kappa, t_i - c_i > v_\kappa^*\}| \geq n - (\kappa - 1)\right\}. \tag{7}$$

Then, for each $i \leq k-1$, $p_i(\mathbf{t}) = 1$ and $q_i(\mathbf{t}) = 0$. For each $i \geq k$,

$$p_i(\mathbf{t}) = q_i(\mathbf{t}) = \begin{cases} 1, & \text{if } |\{l \geq k : t_l - c_l \geq t_i - c_i\}| \leq n - (k-1) \\ 0, & \text{otherwise.} \end{cases}$$

Remark 1 (Single-good allocation). *We briefly discuss the special case of single-good allocation, i.e., $n = 1$, which is studied in Ben-Porath et al. (2014). Precisely, assuming away tie-breaking for simplicity, our n-ASM is as follows: for all $\mathbf{t} \in \mathbf{T}$, if no one other than agent 1 has a net value that is higher than*

[13] This is a generic interpretation. However, when the $n-(k-1)$-th highest net values for agents in $\{k, \ldots, I\}$ is tied, our mechanism withholds some goods. As alternatives, we can adjust the mechanism so that goods are allocated arbitrarily in this tied case. Nevertheless, since the set of type profiles that admit ties has a measure of zero, the allocation rule and checking policy on such a set have no effect on the principal's payoff and, thus, on our analysis.

v_1^*, i.e., $k = n+1 = 2$, then agent 1 obtains the good without being checked, and none of the other agents obtains a good or is checked; if someone other than agent 1 has a net value that is higher than v_1^*, i.e., $k = 1$, then the agent who has the highest net value is checked and obtains the good, and other agents are not checked and obtain no good. This is exactly the optimal favored-agent mechanism in Ben-Porath et al. (2014) that favors agent 1 with threshold v_1^*.

Remark 2 (Strategy-proofness)**.** *Ben-Porath et al. (2014) has argued that the optimal favored agent mechanism for single-good allocation is ex post incentive compatible. And it can be made strategy-proof by slightly modifying the mechanism off the equilibrium path, without changing the generated allocation and checking cost. With multiple units, the n-ASM preserves these properties.*

Remark 3 (Discrete-type model)**.** *Consider a similar model where $\mathbf{T}$ contains only finitely many type profiles. Such a discrete-type model can be solved by the approach of Ben-Porath et al. (2019). They show that the optimal mechanism has to allocate the n goods to the n agents who have the n highest values of $\max\{v_i^*, t_i - c_i\}$.*[14] *Hence, by Lemma 1, the optimal allocation rules in the discrete-type model and the continuous-type model are identical in format. Once the allocation rules coincide, the reduced forms of checking policies have to coincide because of the following relation:*

$$\hat{q}_i(t_i) = \hat{p}_i(t_i) - \inf_{t_i' \in T_i} \hat{p}_i(t_i'), \quad \forall t_i \in T_i,$$

which says that the reduced form of the optimal checking policy is pinned down by the optimal allocation rule. This is equation (4) in Ben-Porath et al. (2014) and it still holds in our multi-unit case by the same argument. Therefore, in terms of format, the n-ASM is equivalent to the optimal mechanism in the discrete-type model considered by Ben-Porath et al. (2019).

3.3 Implementation of the n-ASM

The n-ASM is not only an optimal direct mechanism but also contains an easy implementation in its definition. Particularly, the principal can run a dynamic game according to the Ascending Algorithm. She first announces the n agents who will have endowments and their thresholds v_i^*s. Then free agents simultaneously report their types. The game continues only if an agent with an endowed good is challenged, in which case the challenged agent should report his type; otherwise no additional action is taken by any agent. We refer to this dynamic game as the *ascending game*. In what follows, we will keep our discussion intuitive, without introducing complicated notations, but self-contained.

Compared with the direct mechanism which asks all agents to report their types and then executes $(\mathbf{p}(\mathbf{t}), \mathbf{q}(\mathbf{t}))$, the ascending game has a notable advantage. Namely, it is "obviously strategy-proof." The concept of obvious strategy-proofness is first introduced by Li (2017). A strategy is obviously dominant

[14]See their Example 2 and Appendix C for details.

if, for any deviation, at any information set where both strategies first diverge, the best outcome under the deviation is no better than the worst outcome under the dominant strategy. A mechanism is obviously strategy-proof (OSP) if it has an equilibrium in obviously dominant strategies.

As a direct mechanism, the n-ASM is not OSP. For example, in the favored agent mechanism ($n = 1$), under some type profiles, the best outcome of lying for the favored agent is to still get the good without being checked, but the worst outcome of truth-telling is getting no good. Hence, the optimal direct mechanism is not OSP. Now we argue that the ascending game is OSP. Whenever an agent is required to report his type, he can get a good only if he is checked. Therefore, the best outcome of lying is getting no good. Obviously, the worst outcome of truth-telling is also getting no good. As a result, truth-telling whenever a report is required is obviously dominant for every agent, i.e., the ascending game is OSP.

Remark 4 (Implementation of the optimal mechanism in the discrete-type model)**.** *The optimal mechanism in the discrete-type model considered by Ben-Porath et al. (2019) can also be implemented by the ascending game. In this sense, our analysis which specifies an ex post checking policy complements that of Ben-Porath et al. (2019).*

4 Descending Mechanisms

In this section, we formally define the n-descending mechanism (n-DSM) and compare it with the n-ASM. The n-DSM allocates the n goods one by one. Particularly, given a type profile, the mechanism specifies the allocation rule and the checking policy by n steps. In each step, one good is allocated according to the optimal favored-agent mechanism.

4.1 Definition of the n-Descending Mechanism

The mechanism is formally defined below. Recall that $v_1^* \geq v_2^* \geq \cdots \geq v_I^*$.

Definition 2. *A mechanism* $(\mathbf{p}, \mathbf{q})$ *is said to be the* n***-descending mechanism (***n***-DSM)*** *if for all* $\mathbf{t} \in \mathbf{T}$*, the rules* $(\mathbf{p}(\mathbf{t}), \mathbf{q}(\mathbf{t}))$ *are defined by the algorithm below.*

Descending Algorithm

Set $\tilde{\mathcal{I}} = \mathcal{I}$*. For* $k = 1, 2, \ldots, n$*,*

Step k**.** *Let* $v^* := \max_{i \in \tilde{\mathcal{I}}} v_i^*$ *and* $j = \min\left\{\arg\max_{i \in \tilde{\mathcal{I}}} v_i^*\right\}$*.*[15] *Consider two mutually exclusive cases.*

Case 1. $\max_{i \in \tilde{\mathcal{I}} \setminus \{j\}} \{t_i - c_i\} \leq v^*$.

Set $p_j(\mathbf{t}) = 1$ *and* $q_j(\mathbf{t}) = 0$*. Set* $\tilde{\mathcal{I}} := \tilde{\mathcal{I}} \setminus \{j\}$*.*

Case 2. $\max_{i \in \tilde{\mathcal{I}} \setminus \{j\}} \{t_i - c_i\} > v^*$.

Define $m := \min\left\{\arg\max_{i \in \tilde{\mathcal{I}}} \{t_i - c_i\}\right\}$ *and set* $p_m(\mathbf{t}) = q_m(\mathbf{t}) = 1$*. Set* $\tilde{\mathcal{I}} := \tilde{\mathcal{I}} \setminus \{m\}$*.*

[15] Taking the lowest index is only for the tie-breaking purpose.

If $k < n$, then go to Step $k+1$; otherwise, set $p_i(\mathbf{t}) = q_i(\mathbf{t}) = 0$ for all $i \in \tilde{\mathcal{I}}$ and stop.

The above algorithm describes the following process. Within each step, the lowest-labelled remaining agent, i.e., agent j, is favored in the sense that she is "endowed" with a good. Other agents are free. We implement the mechanism as follows. First, if no free agent challenges the favored agents by reporting a net value (i.e., $t_i - c_i$, where $i \in \tilde{\mathcal{I}} \setminus \{j\}$) higher than v_j^*, then allocate the a good to agent j without checking her. If some free agent challenges the favored agent j, then find the agent in $\tilde{\mathcal{I}}$ who has the highest net value, check (the checking outcome will be truth-telling on equilibrium path) and allocate a good to him. At the same time, let the set of free agents include agent j and exclude the agent who has obtained one good. Next, proceed with the updated set of agents. This continues until all goods are allocated.

4.2 Comparison between the n-ASM and the n-DSM

To compare the n-ASM and the n-DSM, let us first present an example which illustrates their difference in checking cost, in the setting with two goods and three agents. Both the allocation rule and the checking cost will be compared generally after the example.

Example 1. *Consider an allocation of two goods among three agents. Suppose $v_1^* > v_2^* > v_3^*$. We list scenarios in which the 2-ASM and the 2-DSM are different in checking cost.*

Cases	*Checking Costs*	
	2-ASM	*2-DSM*
$t_3 - c_3 < v_2^* < v_1^* < t_2 - c_2$ *and* $t_1 - c_1 < t_2 - c_2$	0	c_2
$t_3 - c_3 < v_2^* < v_1^* < t_2 - c_2$ *and* $t_2 - c_2 < t_1 - c_1$	0	c_1
$t_2 - c_2 < v_2^* < v_1^* < t_3 - c_3 < t_1 - c_1$	c_3	$c_3 + c_1$

It is straightforward albeit tedious to verify that for all other $\mathbf{t}$'s that are not listed in the table, the checking costs of the 2-ASM and the 2-DSM are the same. Therefore, agents are checked weakly more often under the 2-DSM than under the 2-ASM.

However, the allocation rules of the n-ASM and the n-DSM are the same. To see this, we first recall Lemma 1 which says that given a type profile $\mathbf{t} \in \mathbf{T}$, the n-ASM allocates the n goods to the n agents who have the n highest values of $\max\{v_i^*, t_i - c_i\}$. On the other hand, in each step of the Descending Algorithm, one good is allocated to the agent who has the highest value of $\max\{v_i^*, t_i - c_i\}$ among the remaining agents in $\tilde{\mathcal{I}}$. Again, n goods are allocated to the n agents who have the n highest values of $\max\{v_i^*, t_i - c_i\}$. Therefore, for any type profile $\mathbf{t} \in \mathbf{T}$, an agent receives one good under the n-ASM if and only if he receives one good under the n-DSM.

Denote by $\left(\mathbf{p}^{ASM}, \mathbf{q}^{ASM}\right)$ the allocation rule and checking policy for the n-ASM, and by $\left(\mathbf{p}^{DSM}, \mathbf{q}^{DSM}\right)$ the allocation rule and checking policy for the n-DSM. The following theorem compares those two mechanisms.

THEOREM 2. $\mathbf{p}^{ASM}(\mathbf{t}) = \mathbf{p}^{DSM}(\mathbf{t})$ *and* $\mathbf{q}^{ASM}(\mathbf{t}) \leq \mathbf{q}^{DSM}(\mathbf{t})$ *for all* $\mathbf{t} \in \mathbf{T}$.

By Theorem 1, we know that the n-ASM certainly dominates any other mechanism, including the n-DSM, in terms of maximizing the principal's payoff. The key implication of Theorem 2 is that the n-ASM dominates the n-DSM in a very particular way. That is, the n-ASM checks agents less frequently than the n-DSM, which is the only difference between the two mechanisms.

4.3 Exact Difference between the n-ASM and the n-DSM when $n \geq 2$

In this subsection, we study the exact difference between the n-ASM and the n-DSM, in terms of the checking costs, and identify conditions under which the n-ASM delivers a strictly higher payoff. We first introduce a benchmark mechanism $(\mathbf{p}, \mathbf{q})$ to facilitate the comparison, where $\mathbf{p}(\mathbf{t}) = \mathbf{p}^{ASM}(\mathbf{t}) = \mathbf{p}^{DSM}(\mathbf{t})$ and $\mathbf{q}(\mathbf{t}) = \mathbf{p}(\mathbf{t})$ for all $\mathbf{t} \in \mathbf{T}$, i.e., an agent is checked whenever she gets a good. Then comparing the checking costs of the n-ASM and the n-DSM is equivalent to comparing the checking-cost savings relative to the benchmark mechanism. We proceed to explicitly write down the checking-cost savings for both mechanisms, and examine the difference. Let $\mathcal{J}^i$ be the set of agents who have larger identities than i, i.e., $\mathcal{J}^i := \mathcal{I} \setminus \{1, \ldots, i\}$. Let $\bar{F}(x) := 1 - F(x)$ for a cumulative distribution function $F(\cdot)$ and any x in its support.

First, the checking-cost saving of the n-ASM is given by:

$$\sum_{i=1}^{n} c_i \cdot \left\{ \sum_{\tilde{n}=0}^{n-i} \sum_{i_1,\ldots,i_{\tilde{n}} \in \mathcal{J}^i} \prod_{j \in \{i_1,\ldots,i_{\tilde{n}}\}} \bar{F}_j(v_i^* + c_j) \prod_{k \in \mathcal{J}^i \setminus \{i_1,\ldots,i_{\tilde{n}}\}} F_k(v_i^* + c_k) \right\}, \tag{8}$$

where $i_1, \ldots, i_{\tilde{n}}$ are mutually distinct and $\{i_1, \ldots, i_{\tilde{n}}\} = \emptyset$ if $\tilde{n} = 0$. In words, only agents in $\{1, \ldots, n\}$ may receive a good without being checked. For each agent $i = 1, \ldots, n$, we consider $n - i + 1$ cases where agent i is allocated one good without being checked. Particularly, a generic case is as follows: among agents who have larger identities than i, i.e., agents in $\mathcal{J}^i$, there are $\tilde{n}$ agents, denoted by $i_1, \ldots, i_{\tilde{n}}$, whose net values are higher than v_i^*, and the rest have net values lower than v_i^*. In such cases, the cut-off identity k in (7) must be larger than i. Therefore, agent i receives a good but he is not checked, i.e., the checking cost c_i is saved relative to the benchmark mechanism.

Secondly, the checking-cost saving of the n-DSM is given by:

$$\sum_{i=1}^{n} c_i \cdot \left\{ \sum_{\tilde{n}=0}^{n-i} \sum_{i_1,\ldots,i_{\tilde{n}} \in \mathcal{J}^i} \begin{pmatrix} \int_{v_i^*+c_i}^{\bar{t}_i} \prod_{j \in \{i_1,\ldots,i_{\tilde{n}}\}} \bar{F}_j(t_i - c_i + c_j) dF_i(t_i) \\ + F_i(v_i^* + c_i) \prod_{j \in \{i_1,\ldots,i_{\tilde{n}}\}} \bar{F}_j(v_i^* + c_j) \end{pmatrix} \prod_{k \in \mathcal{J}^i \setminus \{i_1,\ldots,i_{\tilde{n}}\}} F_k(v_i^* + c_k) \right\}, \tag{9}$$

where $i_1, \ldots, i_{\tilde{n}}$ are mutually distinct and $\{i_1, \ldots, i_{\tilde{n}}\} = \emptyset$ if $\tilde{n} = 0$. In words, again, only agents in $\{1, \ldots, n\}$ may receive a good without being checked. For each agent $i = 1, \ldots, n$, we consider $n - i + 1$

cases where agent i is allocated one good without checking. Particularly, an agent i gets a good without being checked only if v_i^* serves as the threshold, which means that each of the agents with smaller identities has obtained one good and been excluded from $\tilde{\mathcal{I}}$. Therefore, $n-(i-1)$ goods are left available for agents in $\mathcal{J}^i$. A generic case out of the $n-i+1$ ones, indexed by $\tilde{n}$ where $\tilde{n} \in \{0, 1, \dots, n-i\}$, is that $\tilde{n}$ goods are taken by $\tilde{n}$ challengers in $\mathcal{J}^i$ while v_i^* serves as the threshold. Since $\tilde{n} < n-(i-1)$, at least one good is left for agent i while v_i^* still serves as the threshold. When $t_i - c_i > v_i^*$, the challengers need to beat agent i to receive one good. Then, the precise condition for agent i to get one good without being checked is that there are $\tilde{n}$ agents in $\mathcal{J}^i$ whose net values are higher than $t_i - c_i$, and the rest of the agents in $\mathcal{J}^i$ have net values lower than v_i^*. When $t_i - c_i < v_i^*$, the challengers necessarily beat agent i by definition. Then the precise condition for agent i to get one good without being checked is that there are $\tilde{n}$ agents in $\mathcal{J}^i$ whose net values are higher than v_i^*, and the rest of the agents in $\mathcal{J}^i$ have net values lower than v_i^*.

In the appendix, we will show that (8) $\geq$ (9), where the inequality is strict if (i) $\bar{F}_i(v_i^* + c_i) > 0$ and if (ii) $c_i > 0$, (iii) $\prod_{k \in \mathcal{J}^i \setminus \{i_1, \dots, i_{\tilde{n}}\}} F_k(v_i^* + c_k) > 0$, and (iv) $\prod_{j \in \{i_1, \dots, i_{\tilde{n}}\}} \bar{F}_j(v_i^* + c_j) > 0$. To interpret these conditions, we first recall that our threshold v_i^* is simply a variant of the critical value t_i^* in Ben-Porath et al. (2014). That is, t_i^* is defined implicitly by the equation $\mathbf{E}(t_i) = \mathbf{E}(\max\{t_i, t_i^*\}) - c_i$ and $t_i^* = v_i^* + c_i$. It is easy to see that t_i^* is increasing in c_i. Then condition (i) says that $t_i^* < \bar{t}_i$, i.e., the checking cost of agent i is not extremely large to push the critical value beyond the highest type of i.[16] Now we consider conditions (iii) and (iv). Obviously, they would hold if $0 < F_i(v_j^* + c_i) < 1$ for all $i, j \in \mathcal{I}$, which is equivalent to saying that $\underline{t}_i < t_j^* < \bar{t}_i$, i.e., agents' critical values are not too extreme relative to other agents' types.[17] To sum up, the n-ASM and the n-DSM are strictly different if every agent's checking cost is not very extreme relative to her own distribution and every agent's critical value is not very extreme relative to other agents' distributions.

The follow theorem says that under very general conditions which only exclude uninteresting cases, the n-ASM is strictly superior to the n-DSM.

Theorem 3. *Suppose $n \geq 2$. The n-ASM delivers a strictly higher payoff to the principal than the n-DSM, unless for every $i = 1, \dots, n$, either one of the following conditions holds:*

1. *$c_i = 0$ or $\bar{F}_i(v_i^* + c_i) = 0$.*
2. *For each $\tilde{n} = 0, \dots, n-i$, and each $\{i_1, \dots, i_{\tilde{n}}\} \subset \mathcal{J}^i$, either $\prod_{k \in \mathcal{J}^i \setminus \{i_1, \dots, i_{\tilde{n}}\}} F_k(v_i^* + c_k) = 0$ or $\prod_{j \in \{i_1, \dots, i_{\tilde{n}}\}} \bar{F}_j(v_i^* + c_j) = 0$.*

[16] By equation (5), $\bar{F}_i(v_i^* + c_i) = 0$ if and only if $c_i \geq \bar{t}_i - \mathbb{E}[t_i]$. If $c_i \geq \bar{t}_i - \mathbb{E}[t_i]$ for all i, an optimal mechanism is to (randomly) allot the good without checking.

[17] Particularly, if all the agents are ex ante identical, then this condition holds if the checking cost is positive but not extremely large.

Remark 5 (Quantification of the difference between two mechanisms.). *A precise quantification of the difference necessarily depends on the specific distributions of agent types. Nevertheless, the difference in terms of the number of agents checked can be as large as $n-1$, which is significant given that the maximal number of agents checked is n under both mechanisms. For an example where such a difference arises, we can consider the following generic situation:*

$$\begin{aligned} t_1 - c_1 > t_2 - c_2 &\quad > v_1^* \\ t_2 - c_2 > t_3 - c_3 &\quad > v_2^* \\ \cdots &\quad \cdots \\ t_{n-1} - c_{n-1} > t_n - c_n &\quad > v_{n-1}^* \qquad \textit{and} \\ t_n - c_n &\quad > v_n^* \quad > \max_{i>n} t_i - c_i. \end{aligned}$$

Obviously, in this situation, the n-ASM checks no agent while the n-DSM checks the first $n-1$ agents.

5 Concluding Remarks

In this paper, we characterize the mechanisms which maximize the net expected value to the principal from allocating multiple identical goods. Such an optimal mechanism specifies a cut-off for each type profile, which divides the agents into two groups. In the first group, goods are allocated to these agents without the principal learning agents' values, which incurs no checking cost but may result in allocation inefficiency. In the second group, the rest of the goods are allocated efficiently to the agents who have the highest values, which requires these agents being checked and leads to some checking cost. Our optimal mechanism balances the benefit of efficient allocation and the cost of checking agents. The optimal mechanism reduces to the optimal favored-agent mechanism in Ben-Porath et al. (2014) when we have only one good, and provide an extension of Ben-Porath et al. (2019) when their approach is applied to allocation problems with costly verification. Moreover, the optimal mechanism is straightforward to implement: the principal only needs to run an allocation process according to the algorithm which defines the optimal mechanism. Lastly, the optimal mechanism generally delivers a strictly higher payoff to the principal than allocating the goods one by one.

Three of our assumptions are of further interest. First, the goods are identical, which significantly simplifies the incentive compatibility constraints. Particularly, the constraints are formulated merely by the probability of obtaining a good and the probability of being checked, which only depend on agents' reported types instead of their true types. Second, agents' types are independently distributed. This assumption echoes the use of agent-specific thresholds, and it allows for interpretations (properties) of the thresholds that hold regardless of other agents' types. Third, the principal learns an agent's type once she inspects the agent. This allows the principal to punish an agent who lies in the most efficient way: once found lying, the agent certainly gets nothing. Again, it significantly simplifies the problem. Relaxing any of those assumptions may lead to a future research direction that brings this topic to another new scope.

Appendix A Proof of Lemma 1

The "only-if" part. Suppose $p_i(\mathbf{t}) = 1$. According to the ASCENDING ALGORITHM, there are two possibilities for agent i to obtain a good. First, $i \leq n$ and he is not challenged. Then within the group $\{i+1, \ldots, I\}$, only agents who left with a good may have had a net value that is larger than v_i^*. Therefore, $|\{j \in \{i+1, \ldots, I\} : t_j - c_j > v_i^*\}| \leq n - i$. Since $v_j^* \leq v_i^*$ for all $j \in \{i+1, \ldots, I\}$, we have

$$|\{j \in \{i+1, \ldots, I\} : w_j > w_i\}| \leq n - i. \tag{10}$$

Obviously,

$$|\{j \in \{1, \ldots, i-1\} : w_j \geq w_i\}| \leq i - 1. \tag{11}$$

Note that

$$\begin{aligned} &|\{j \in \mathcal{I} : w_j > w_i\}| + |\{j \in \{1, \ldots, i-1\} : w_j = w_i\}| \\ =&|\{j \in \{i+1, \ldots, I\} : w_j > w_i\}| + |\{j \in \{1, \ldots, i-1\} : w_j \geq w_i\}|. \end{aligned}$$

Hence, the desired inequality (6) is derived by summing up (10) and (11).

Second, i obtains a good by successfully challenging some j with $j \leq n$ and $j < i$. Then $t_i - c_i > \max\{v_j^*, t_j - c_j\}$. Moreover, within the group $\{j+1, \ldots, I\}$, only agents who left with a good may have had a net value that is larger than $t_i - c_i$ (or equal to $t_i - c_i$ in the tied case); otherwise, i can never be the successful challenger. Hence,

$$|\{k \in \{j+1, \ldots, I\} : t_k - c_k > t_i - c_i\}| + |\{k \in \{j+1, \ldots, i-1\} : t_k - c_k = t_i - c_i\}| \leq n - j.$$

Since $v_k^* \leq v_j^* < t_i - c_i$ for all $k \in \{j+1, \ldots, I\}$, we have

$$|\{k \in \{j+1, \ldots, I\} : w_k > w_i\}| + |\{k \in \{j+1, \ldots, i-1\} : w_k = w_i\}| \leq n - j. \tag{12}$$

Obviously,

$$|\{k \in \{1, \ldots, j-1\} : w_k \geq w_i\}| \leq j - 1. \tag{13}$$

Note that $t_i - c_i > w_j = \max\{v_j^*, t_j - c_j\}$ implies $w_i > w_j$, which, together with (12) and (13), leads to the desired inequality (6).

The "if" part. Now suppose (6) holds for agent i under $\mathbf{t}$. If $w_i = v_i^*$, i.e., $v_i^* \geq t_i - c_i$, then $i \leq n$. Obviously, we have $v_j^* \geq v_i^*$ for all $j \in \{1, \ldots, i-1\}$, which implies $w_j \geq w_i$. Therefore, $|\{j \in \{1, \ldots, i-1\} : w_j \geq w_i\}| = i - 1$. By (6), we have $|\{j \in \{i+1, \ldots, I\} : w_j > w_i\}| \leq n - i$. This in turn implies $|\{j \in \{i+1, \ldots, I\} : t_j - c_j > v_i^*\}| \leq n - i$, i.e., agent i is not challenged. Hence, $p_i(\mathbf{t}) = 1$.

If $w_i = t_i - c_i$, i.e., $t_i - c_i > v_i^*$, we need to consider two cases. When $i \leq n$ and i is not successfully challenged, we are done as agent i gets a good. When either $i > n$ or $i \leq n$ but i is successfully challenged, agent i has to challenge some other agent j with $j \leq n$ and $j < i$. In this case, we have $t_i - c_i > v_j^*$ and that i was not successful when challenging l with $j < l \leq n$, i.e,

$$|\{k \in \{j+1, \ldots, I\} : t_k - c_k > t_i - c_i\}| + |\{k \in \{j+1, \ldots, i-1\} : t_k - c_k = t_i - c_i\}| \geq n - j.$$

Since $t_i - c_i > v_j^* \geq v_k^*$ for all $k \in \{j+1, \ldots, I\}$ and $v_j^* > v_i^*$, we have

$$|\{k \in \{j+1, \ldots, I\} : w_k > w_i\}| + |\{k \in \{j+1, \ldots, i-1\} : w_k = w_i\}| \geq n - j,$$

By (6), we know that $|\{k \in \{1, \ldots, j\} : w_k \geq w_i\}| \leq j - 1$ and thus

$$|\{k \in \{1, \ldots, j\} : w_k \geq t_i - c_i\}| \leq j - 1. \tag{14}$$

Since (14) holds for any j that i challenges, such challenges cannot always fail, i.e., for at least one $k \in \{1, \ldots, j\}$, we have $t_i - c_i > w_k = \max\{t_k - c_k, v_k^*\}$. Hence, $p_i(\mathbf{t}) = 1$.

Appendix B Proof Ideas of Theorem 1 ($n = 2$)

We focus on the special case of $n = 2$ in this appendix. The proof of Theorem 1 for this case demonstrates the key ideas for the general cases, which are visualizable when $I = 3$ and $n = 2$.[18] Precisely, when we have three agents as in Examples 2-5, the set of type profiles $\mathbf{T}$ is a subset of $\mathbb{R}^3$. Thus, we can illustrate each step of the proof in a 3-dimensional diagram. Technical details omitted in this appendix can be found in the Online Appendix E. We refer the readers to Chua et al. (2019) for the case of $n > 2$, where the proof is more tedious but the ideas remain the same.

We first provide a roadmap, illustrated in Figure 1, before proceeding with more details. Mathematically, the principal's problem (1)-(4) is an infinite dimensional linear programming with infinitely many constraints. Our basic idea is to reduce the dimensions. We first identify a set of binding constraints such that $\mathbf{q}$ can be represented by $\mathbf{p}$, which means that the choice variable is simplified from $(\mathbf{p}, \mathbf{q})$ to $\mathbf{p}$. Second, we classify the set of feasible mechanisms into convex and compact subsets. Then, within each of these convex and compact subsets, we reduce the infinite dimensional problem to a finite dimensional problem. The link between them is what we called *threshold mechanisms*, in which an I dimensional vector pins down the entire infinite dimensional mechanism $\mathbf{p}$. This is justified by

[18] The techniques to solve the continuous-type model is significantly different from that for the discrete-type model. According to the technical details of Ben-Porath et al. (2019) (the definition of $\tilde{v}_i(t_i)$ at p.565), it is not obvious how their approach can be used to solve a continuous-type model. One may conjecture that some limiting argument will work. A seemingly possible route may be as follows. We first approximate/discretize the continuum type space by a finite type space, say, with a parameter measuring the size of a grid. Once we find the optimal allocation rule in the finite type space model, taking limit of the parameter will lead us to an optimal allocation rule for the continuum type space. However, taking limit requires that the solution to the discretized optimization problem is continuous in the parameter. To the best of our knowledge, we do not have a reliable mathematical result, e.g. the Berge Maximum Theorem (Aliprantis and Border, 2006, p. 570), to justify this limiting argument.

Nevertheless, once we characterize the optimal mechanisms as the n-ASMs, we do *prove* an approximation result indirectly: discretizing the continuous distributions and using the results of Ben-Porath et al. (2019) will indeed lead us to an approximately optimal mechanism for the continuous-type allocation problem. That is why we view our characterization as a technical contribution and a nontrivial extension for Ben-Porath et al. (2019).

showing that it is sufficient to restrict attentions to the threshold mechanisms. Finally, we examine the extreme points of each convex and compact subsets, and find the optimal ones among the extreme threshold mechanisms.

More precisely, the first step of simplifying $(\mathbf{p}, \mathbf{q})$ to $\mathbf{p}$ is done in Appendix B.1. Second, we classify the set of feasible $\mathbf{p}$'s in Appendix B.2, where we have stylized convex and compact subsets of $\mathbf{p}$'s. Third, for each subset of the feasible $\mathbf{p}$'s, we introduce a class of threshold mechanisms within it (see Appendix B.3). Fourth, we show that a mechanism is optimal only if it is an extreme threshold mechanism up to equivalence-relabelling-randomization (see Propositions 1-1$'$ in Appendix B.4). Therefore, we can restrict our attention to extreme threshold mechanisms. In Appendix B.5, we search for the optimal extreme threshold mechanisms. The principal's choice among extreme threshold mechanisms is equivalent to choosing two distinct agents i and j, and respectively two thresholds v and v' such that $v \leq v'$. The choices of (i, v) and that of (j, v') are relatively independent in the sense that the only links are $i \neq j$ and $v \leq v'$. We drop the constraint $v \leq v'$ and study them separately. Finally, we build the link and solve the principal's problem: We show that the optimal choice of $(j, v'; i, v)$ is $(1, v_1^*; 2, v_2^*)$, which is the unique optimum up to equivalence-relabelling-randomization (see Proposition 2 in Appendix B.5). It is straightforward to verify that the mechanism represented by $(1, v_1^*; 2, v_2^*)$ is equivalent to the 2-ASM. Therefore, the 2-ASM is the unique optimal extreme threshold mechanism up to equivalence-relabelling-randomization, i.e., a mechanism is optimal if and only if it is essentially a randomization over the 2-ASMs.

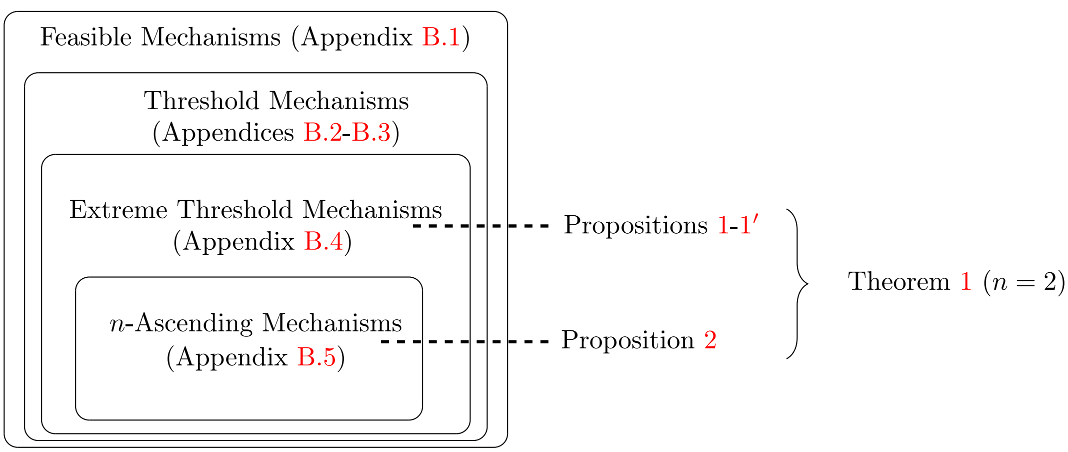


Figure 1: Proof Sketch of Theorem 1.

B.1 Simplify the Principal's Problem

In this subsection, we simplify the principal's problem (1)-(4). The goal is to eliminate $\mathbf{q}$ from the choice variable such that the principal only needs to choose $\mathbf{p}$. We follow Ben-Porath et al. (2014) to define $\varphi_i = \inf_{t_i' \in T_i} \mathbf{E}_{\mathbf{t}_{-i}} [p_i(t_i', \mathbf{t}_{-i})]$, i.e., φ_i is the infimum of agent i's expected total assignment probability. Then by the same argument as in (Ben-Porath et al., 2014, pp. 3793-4), the principal's problem can be simplified to

$$\max_{\mathbf{p}} \ \mathbf{E}\left(\sum_{i\in\mathcal{I}} p_i(\mathbf{t})(t_i - c_i) + \sum_{i\in\mathcal{I}} \varphi_i c_i\right) \tag{15}$$

$$\text{s.t. } p_i : \mathbf{T} \to [0,1], \forall i \in \mathcal{I}; \sum_{i\in\mathcal{I}} p_i(\mathbf{t}) \le 2, \forall \mathbf{t} \in \mathbf{T};$$

$$\text{and } \varphi_i = \inf_{t_i' \in T_i} \mathbf{E}_{\mathbf{t}_{-i}}\left[p_i(t_i', \mathbf{t}_{-i})\right], \forall i \in \mathcal{I}, \tag{16}$$

which is independent of $\mathbf{q}$. From now on, we also refer to a mechanism as $\mathbf{p}$.

We say a mechanism $\mathbf{p}$ is *feasible* if $p_i : \mathbf{T} \to [0,1]$ for all i and $\sum_{i\in\mathcal{I}} p_i(\mathbf{t}) \le 2$ for all $\mathbf{t}$. Denote P as the set of feasible mechanisms, i.e.,

$$P := \left\{\mathbf{p} : p_i(\mathbf{t}) \in [0,1], \forall \mathbf{t} \in \mathbf{T}, \forall i \in \mathcal{I} \text{ and } \sum_{i\in\mathcal{I}} p_i(\mathbf{t}) \le 2, \forall \mathbf{t} \in \mathbf{T}\right\}.$$

A vector $\boldsymbol{\varphi}$ is said to be *feasible* if there exists a feasible mechanism $\mathbf{p}$ such that $\varphi_i = \inf_{t_i'\in T_i} \mathbf{E}_{\mathbf{t}_{-i}}\left[p_i(t_i', \mathbf{t}_{-i})\right]$ for all i, i.e., (16) holds. The following lemma characterizes the set of feasible $\boldsymbol{\varphi}$'s.

Lemma 2. *$\boldsymbol{\varphi}$ is feasible if and only if $\varphi_i \in [0,1]$ for all $i \in \mathcal{I}$ and $\sum_{i\in\mathcal{I}} \varphi_i \le 2$.*

Denote by Φ the set of feasible $\boldsymbol{\varphi}$'s, i.e., $\Phi := \left\{\boldsymbol{\varphi} : \varphi_i \in [0,1], \forall i \in \mathcal{I}, \text{ and } \sum_{i\in\mathcal{I}} \varphi_i \le 2\right\}$, which is illustrated in Example 2.

Example 2. *We consider the allocation problem of two goods to three agents, i.e., $n = 2$ and $I = 3$. The set Φ in $\mathbb{R}^3$ is drawn in Figure 2, which is independent of the domain of type profiles $\mathbf{T}$ and the distributions F_i's.*

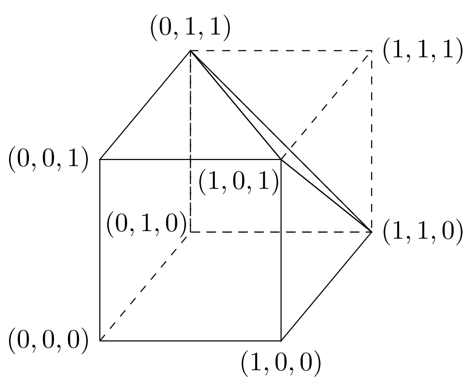


Figure 2: A 3-Dimensional Illustration of Φ.

Lemma 2 suggests that we can solve the principal's problem by first finding the optimal $\mathbf{p}$ for a given (subset of) vector $\boldsymbol{\varphi}$ and then solving for the optimal $\boldsymbol{\varphi} \in \Phi$, which is done in what follows.

B.2 Classification of Φ and P

We classify Φ into $I + 1$ subsets and denote Φ_j, $j \in \mathcal{I} \cup \{\emptyset\}$, as the j-th class. A short intuition for $\boldsymbol{\varphi} \in \Phi_j$ is that the j-th dimension φ_j is "relatively larger" than φ_i's, $i \ne j$. Similarly, for $\boldsymbol{\varphi} \in \Phi_\emptyset$,

φ_i's are "relatively even." The classes will be further divided into slices which are convex and compact subsets of Φ.

We first introduce some notations for different subsets of $\mathbf{T}$, which is illustrated in Example 3. Let $\underline{v} = \min_{i\in\mathcal{I}}\{\underline{t}_i - c_i\}$ and $\bar{v} = \min_{i\in\mathcal{I}}\{\bar{t}_i - c_i\}$. For each $v \in [\underline{v}, \bar{v}]$ and each $j \in \mathcal{I} \cup \{\emptyset\}$, we define $K^j(v)$ as the set of type profiles such that agent j (nobody when $j = \emptyset$) has a net value which is higher than v and all other agents have net values lower than v, i.e.,

$$K^j(v) := \{\mathbf{t} \in \mathbf{T} : t_j - c_j \geq v \geq t_l - c_l, \forall l \neq j\}.$$

We call $K^{\emptyset}(v)$ the *palm*, $K^i(v)$ the i-th *finger* to facilitate our verbal description. Define the *claw* as $K(v) := K^{\emptyset}(v) \bigcup \bigcup_{i\in\mathcal{I}} K^i(v)$.

Example 3 (Example 2 revisited). *We consider the allocation of two goods among three agents as in Example 2. Particularly, we have $\mathbf{T} = \prod_{i=1}^{3} T_i \subseteq \mathbb{R}^3$. We assume $\underline{t}_1 - c_1 = \underline{t}_2 - c_2 = \underline{t}_3 - c_3 = \underline{v}$ and $\bar{t}_1 - c_1 = \bar{t}_2 - c_2 = \bar{t}_3 - c_3 = \bar{v}$. In Figure 3, we draw $K^i(v)$'s for a given $v \in [\underline{v}, \bar{v}]$. The point O stands for the type profile $(\underline{t}_1 - c_1, \underline{t}_2 - c_2, \underline{t}_3 - c_3)$, C for $(\bar{t}_1 - c_1, \bar{t}_2 - c_2, \bar{t}_3 - c_3)$, and F for (v, v, v). In the figure, $K^{\emptyset}(v)$ and $K^i(v)$ are the cube OF and the column $D_iG_{-\mathbf{i}}$ respectively, where $i = 1, 2, 3$.*

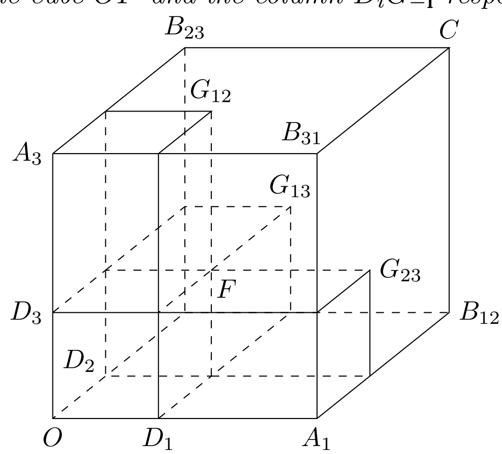


Figure 3: A 3-Dimensional Illustration of $\mathbf{T}$, $K^{\emptyset}(v)$ and $K^i(v)$ for all $i \in \mathcal{I}$.

Now we define the classes Φ_j, $j \in \mathcal{I} \cup \{\emptyset\}$ and illustrate them with Figure 2. The class for $j = \emptyset$ consists of slices indexed by $v \in [\underline{v}, \bar{v}]$, i.e., $\Phi_{\emptyset} = \cup_{v\in[\underline{v},\bar{v}]}\Phi_{\emptyset}(v)$.[19] For each $v \in (\underline{v}, \bar{v}]$, the slice $\Phi_{\emptyset}(v)$ is defined as

$$\Phi_{\emptyset}(v) := \left\{ \boldsymbol{\varphi} \in \Phi : \begin{array}{c} \sum_{i\in\mathcal{I}} \varphi_i F_i(v + c_i) = \int_{K(v)} dF + \int_{K^{\emptyset}(v)} dF, \\ \text{for every } i \in \mathcal{I}, \varphi_i F_i(v + c_i) \leq \int_{K(v)\setminus K^i(v)} dF \end{array} \right\}.$$

Since all $\boldsymbol{\varphi} \in \Phi_{\emptyset}(v)$ satisfy a linear equation, a slice $\Phi_{\emptyset}(v)$ is a subset of a $(I-1)$-dimensional hyperplane in $\mathbb{R}^I$. The corner slice for $\underline{v}$ is a singleton defined by $\Phi_{\emptyset}(\underline{v}) := \{\mathbf{0}\}$.

The class for each $j \in \mathcal{I}$ consists of slices indexed by a pair of net values $(v, v') \in [\underline{v}, \bar{v}] \times [v, \bar{v}]$, i.e.,

[19]Those slices may not be disjoint. We will include all the limit elements such that all the slices are compact sets. The same footnote applies to slices of Φ_j for $j \in \mathcal{I}$.

$\Phi_j := \bigcup_{(v,v')\in[\underline{v},\bar{v}]\times[v,\bar{v}]} \Phi_j(v,v')$. For each $j \in \mathcal{I}$ and each pair $(v,v') \in (\underline{v},\bar{v}] \times [v,\bar{v}]$, the slice $\Phi_j(v,v')$ is defined as

$$\Phi_j(v,v') := \left\{ \boldsymbol{\varphi} \in \Phi : \begin{array}{ll} \varphi_j = \prod_{k\in\mathcal{I}\setminus\{j\}} F_k(v'+c_k) + \sum_{i\neq j} [1-F_i(v'+c_i)] \prod_{k\neq i,j} F_k(v'+c_k); \text{ and} & \\ \sum_{i\in\mathcal{I}\setminus\{j\}} \varphi_i F_i(v+c_i) = \int_{K^{\emptyset}(v)\cup K^j(v)} dF, & \text{if } |i\in\mathcal{I}\setminus\{j\} : \varphi_i > 0| \geq 2, \\ \varphi_i = \prod_{k\neq i,j} F_k(v+c_k), & \text{if } \varphi_i > 0 \text{ and } \varphi_k = 0, \forall k \neq i,j \end{array} \right\}.$$

We hasten to clarify the close connection between $\Phi_{\emptyset}(v)$ and $\Phi_j(v,v')$. Note that when $F_j(v'+c_j) > 0$, we have

$$\prod_{k\in\mathcal{I}\setminus\{j\}} F_k(v'+c_k) + \sum_{i\neq j} [1-F_i(v'+c_i)] \prod_{k\neq i,j} F_k(v'+c_k) = \frac{1}{F_j(v'+c_j)} \int_{K(v')\setminus K^j(v')} dF;$$

and that when $F_i(v+c_i) > 0$, we have

$$\prod_{k\neq i,j} F_k(v+c_k) = \frac{1}{F_i(v+c_i)} \int_{K^{\emptyset}(v)\cup K^j(v)} dF.$$

Therefore, when $v' = v$, it is straightforward to verify that $\Phi_j(v,v) \subseteq \Phi_{\emptyset}(v)$. As a corollary, for each $j \in \mathcal{I}$, the class Φ_j has nonempty intersection with $\Phi_{\emptyset}$.

Since for all $\boldsymbol{\varphi} \in \Phi_j(v,v')$, the j-th entries are fixed and the other $I-1$ entries satisfy a linear equation, we know that a slice $\Phi_j(v,v')$ is a subset of a $I-2$ dimensional hyperplane in $\mathbb{R}^I$. For each $j \in \mathcal{I}$ and each $v' \in (\underline{v},\bar{v}]$, we define by $\Phi_j(\underline{v},v')$ the boundary cases such that

$$\Phi_j(\underline{v},v') := \left\{ \boldsymbol{\varphi} \in \Phi : \begin{array}{c} \varphi_i = 0, \forall i \neq j \text{ and} \\ \varphi_j = \prod_{k\in\mathcal{I}} F_k(v'+c_k) + \sum_{i\neq j} [1-F_i(v'+c_i)] \prod_{k\neq i} F_k(v'+c_k) \end{array} \right\}.$$

Finally, we define the corner "slice", which is a singleton, as $\Phi_j(\underline{v},\underline{v}) := \{\mathbf{0}\}$.

The following lemma says that any $\boldsymbol{\varphi} \in \Phi$ must belong to one of the slices in Φ_j for some j, i.e.,

$$\left[\bigcup_{v\in[\underline{v},\bar{v}]} \Phi_{\emptyset}(v) \right] \bigcup \left[\bigcup_{j\in\mathcal{I},(v,v')\in[\underline{v},\bar{v}]\times[v,\bar{v}]} \Phi_j(v,v') \right] = \Phi.$$

Lemma 3. *For any vector $\boldsymbol{\varphi} \in \Phi$, there either exists a net value $v \in [\underline{v},\bar{v}]$ such that $\boldsymbol{\varphi} \in \Phi_{\emptyset}(v)$, or exist $j \in \mathcal{I}$ and a pair of net values $(v,v') \in [\underline{v},\bar{v}] \times [v,\bar{v}]$ such that $\boldsymbol{\varphi} \in \Phi_j(v,v')$.*

The following lemma characterizes the extreme points of $\Phi_{\emptyset}(v)$ and $\Phi_j(v,v')$, when not a singleton.

Lemma 4. *For each $v \in (\underline{v},\bar{v}]$, the set of extreme points of $\Phi_{\emptyset}(v)$ is given by*

$$\Phi_{\emptyset}^{example}(v) := \left\{ \boldsymbol{\varphi} \in \Phi : \exists \text{ distinct } i,j \in \mathcal{I} \text{ s.t. } \begin{array}{c} \varphi_k = 0, \forall k \in \mathcal{I}\setminus\{i,j\}, \\ \varphi_j F_j(v+c_j) = \int_{K(v)\setminus K^j(v)} dF, \text{ and} \\ \varphi_i F_i(v+c_i) = \int_{K^{\emptyset}(v)\cup K^j(v)} dF. \end{array} \right\}.$$

For each $j \in \mathcal{I}$ and each pair of $(v,v') \in (\underline{v},\bar{v}] \times [v,\bar{v}]$, the set of extreme points of $\Phi_j(v,v')$ is given by

$$\Phi_j^{example}(v,v') := \left\{ \boldsymbol{\varphi} \in \Phi : \exists i \in \mathcal{I}\setminus\{j\} \text{ s.t. } \begin{array}{c} \varphi_k = 0, \forall k \in \mathcal{I}\setminus\{i,j\}, \\ \varphi_j F_j(v'+c_j) = \int_{K(v')\setminus K^j(v')} dF, \text{ and} \\ \varphi_i F_i(v+c_i) = \int_{K^{\emptyset}(v)\cup K^j(v)} dF. \end{array} \right\}.$$

We close this subsection by classifying P. By Lemma 2, we know that a classification of Φ induces a classification of P. To be precise, for each $j \in \mathcal{I} \cup \{\emptyset\}$, the set of mechanisms that corresponds to Φ_j is defined as follows:

$$P_j := \left\{ \mathbf{p} \in P : \exists \boldsymbol{\varphi} \in \Phi_j \text{ s.t. } \varphi_i = \inf_{t_i' \in T_i} \mathbf{E}_{\mathbf{t}_{-i}} \left[p_i(t_i', \mathbf{t}_{-i}) \right], \forall i \right\}. \tag{17}$$

The slices of P_j, i.e., $P_\emptyset(v)$'s and $P_j(v, v')$'s, are defined analogous to slices of Φ_j.

B.3 Threshold Mechanisms: Definitions

In this subsection, we introduce a *threshold mechanism* within each slice of each class of P. Let $\boldsymbol{\varphi}^{\mathbf{p}} \in \Phi$ be the vector that is induced from $\mathbf{p}$ in the sense of (16). The advantage of threshold mechanisms is that the infinite dimensional $\mathbf{p}$ is pinned down by an I-dimensional vector $\boldsymbol{\varphi}^{\mathbf{p}}$.

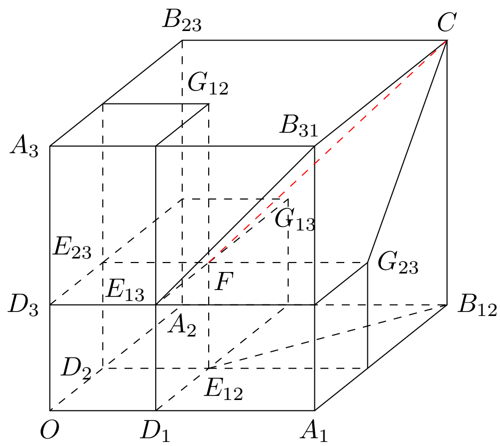
We provide a brief intuition for the threshold mechanisms. The principal's objective function $\mathbf{E}\left(\sum_i p_i(\mathbf{t})(t_i - c_i) + \sum_i \varphi_i c_i\right)$ represents a trade-off between allocation efficiency and checking-cost saving, where $\sum_i \varphi_i c_i$ is the saved cost. When goods are allocated without checking agents' values, agents with low value may receive the goods with the presence of high-valued agents, i.e., there may be efficiency loss. The idea of threshold mechanisms is to accommodate the efficiency loss at low value profiles in the sense that either (i) every net value $t_i - c_i$ is low or (ii) the number of dimensions which have high net values is small. Particularly, the first case corresponds to the palm $K^\emptyset(v)$, where $t_i - c_i < v$ for all $i \in \mathcal{I}$; and the second case corresponds to the fingers $K^j(v)$, where $t_j - c_j > v$ and other agents all have low net values. Here is the intuition to save checking cost and accommodate efficiency loss: For value profiles in $K^\emptyset(v)$, agents' values are too low to be worth distinguishing, in which case we would allocate the two goods without checking. For value profiles in $K^j(v)$, obviously, agent j deserves a good. However, the other agents' values are too low to be worth distinguishing, in which case we would allocate the other good without checking.

Below we first define threshold mechanisms for $j = \emptyset$, then for $j \in \mathcal{I}$. They are (partially) illustrated in Examples 4 and 5 respectively.

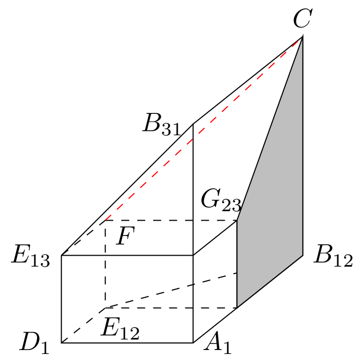
Definition 3. *A feasible mechanism $\mathbf{p} \in P_\emptyset(v)$ is called a $\emptyset$-**threshold mechanism** if the following three conditions hold:*

1. *For each $\mathbf{t} \notin K(v)$, two goods are allocated to the two agents with the highest net values, i.e.,*
$$p_i(t_i, \mathbf{t}_{-i}) = \begin{cases} 1, & \text{if } |\{j \in \mathcal{I} : t_j - c_j \geq t_i - c_i\}| \leq 1; \\ 0, & \text{otherwise.} \end{cases}$$
2. *For each $\mathbf{t} \in K^i(v)$, agent i gets one good for sure, i.e., $p_i(\mathbf{t}) = 1$.*
3. *For each $i \in \mathcal{I}$ and each t_i with $\underline{t}_i - c_i \leq t_i - c_i \leq v$, $\mathbf{E}_{\mathbf{t}_{-i}}[p_i(t_i, \mathbf{t}_{-i})] = \varphi_i^{\mathbf{p}}$.*

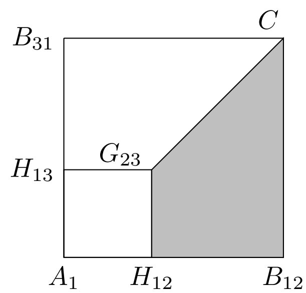
Example 4 (Example 3 revisited)**.** *We consider the allocation of two goods among three agents as in Example 3. The entire domain $\mathbf{T}$ is included in Figure 4a for completeness. Let F be the profile of net values (v, v, v). In the subdomain indicated by Figure 4b, agent 1 receives one good for sure according* to rule 1 and rule 2 (since he has the highest net value). Figure 4c further illustrates rule 1 (fixing $t_1 - c_1 = \bar{v}$), which says that the other good goes to the agent who has the second highest net value. Particularly, agent 2 receives the other good in region $H_{12}C$ and agent 3 receives the other good in region $H_{13}C$.



(a) The Domain $\mathbf{T}$.



(b) $\mathbf{t} : t_1 - c_1 > \max\{v^*, t_2 - c_2, t_3 - c_3\}$.



(c) Allocation between Agents 2 and 3.

Figure 4: A 3-Dimensional Illustration of $\mathbf{t}$ for $\emptyset$-threshold mechanisms.

Definition 4. *For each $j \in \mathcal{I}$, a mechanism $\mathbf{p}$ in $P_j(v, v')$ is called a j**-threshold mechanism** if the following conditions hold:*

1. *For each $\mathbf{t} \notin K(v')$, two agents with the highest net values obtain the goods, i.e.,*

$$p_j(\mathbf{t}) = \begin{cases} 1, & \text{if } |\{k \in \mathcal{I} : t_k - c_k \geq t_j - c_j\}| \leq 1; \\ 0, & \text{otherwise.} \end{cases}$$

2. *For each $\mathbf{t} \in K(v') \setminus \left[K^{\emptyset}(v') \cup K^j(v')\right]$, agent j receives one good and the agent who has the highest net value receives one good. For each $\mathbf{t} \in \left[K^{\emptyset}(v') \cup K^j(v')\right] \setminus \left[K^{\emptyset}(v) \cup K^j(v)\right]$, agent j receives one good, and the other good is allocated to the agent who has the highest net value among agents in $\mathcal{I} \setminus \{j\}$, i.e., for each $i \in \mathcal{I} \setminus \{j\}$,*

$$p_i(\mathbf{t}) = \begin{cases} 1, & \text{if } t_i - c_i > t_k - c_k \text{ for all } k \neq i, j; \\ 0, & \text{otherwise.} \end{cases}$$

   Finally, for each $\mathbf{t} \in \left[K^{\emptyset}(v) \cup K^j(v)\right]$, agent j receives one good.

3. *For each $i \in \mathcal{I} \setminus \{j\}$ and each t_i with $\underline{t}_i - c_i \leq t_i - c_i \leq v$, we have $\mathbf{E}_{\mathbf{t}_{-i}}\left[p_i(t_i, \mathbf{t}_{-i})\right] = \varphi_i^{\mathbf{p}}$. Moreover, for each t_j with $\underline{t}_j - c_j \leq t_j - c_j \leq v'$, we have $\mathbf{E}_{\mathbf{t}_{-j}}\left[p_j(t_j, \mathbf{t}_{-j})\right] = \varphi_j^{\mathbf{p}}$.*

Example 5 (Example 3 revisited). *We consider the allocation of two goods among three agents as in Example 3. Let F be the net-value profile (v, v, v) and H be the net-value profile (v', v', v'). Two key features, in addition to what already exist in Definition 3, are as follows: First, agent $j = 1$ gets one good for sure in the claw $K(v')$. Second, for each $\mathbf{t} \in \left[K^{\emptyset}(v') \cup K^{j}(v')\right] \setminus \left[K^{\emptyset}(v) \cup K^{j}(v)\right]$, that is, the subdomain in the right panel of Figure 5, the second good is allocated to the agent with the highest net value among agents in $\mathcal{I} \setminus \{j\}$.*

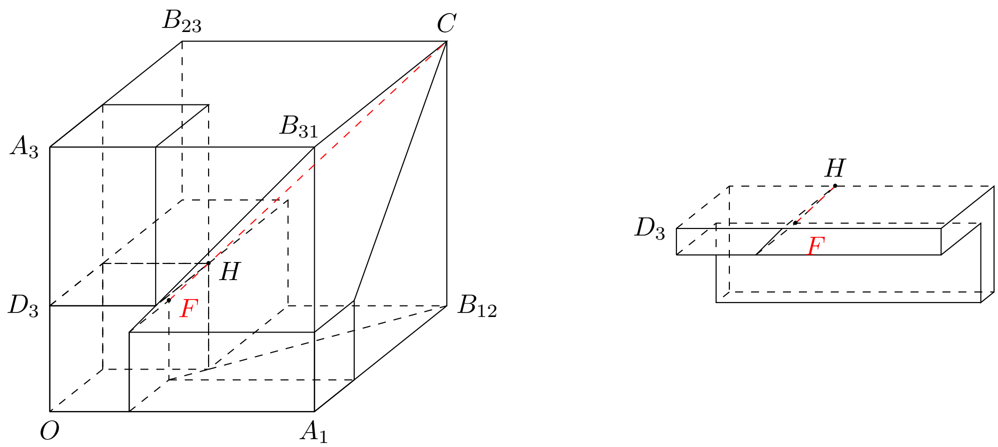


Figure 5: A 3-Dimensional Illustration of j-Threshold Mechanisms.

The following lemma guarantees that for every $\mathbf{p}$ in P, there exists a threshold mechanism that induces the same $\boldsymbol{\varphi}$ vector as induced by $\mathbf{p}$.

Lemma 5. *For each $v \in [\underline{v}, \bar{v}]$ and each $\mathbf{p} \in P_{\emptyset}(v)$, there exists a $\emptyset$-threshold mechanism $\mathbf{p}'$ in $P_{\emptyset}(v)$ such that $\boldsymbol{\varphi}^{\mathbf{p}'} = \boldsymbol{\varphi}^{\mathbf{p}}$. For each $j \in \mathcal{I}$, each $(v, v') \in [\underline{v}, \bar{v}] \times [v, \bar{v}]$ and each $\mathbf{p} \in P_j(v, v')$, there exists a j-threshold mechanism $\mathbf{p}'$ in $P_j(v, v')$ such that $\boldsymbol{\varphi}^{\mathbf{p}'} = \boldsymbol{\varphi}^{\mathbf{p}}$.*

B.4 Threshold Mechanisms: Necessity

In this subsection, we will show that without loss of generality, we can restrict our attention to threshold mechanisms, particularly a subset of threshold mechanisms called *extreme threshold mechanisms*. We say a mechanism $\mathbf{p}$ is extreme if the induced $\boldsymbol{\varphi}^{\mathbf{p}}$ is an extreme point either in $\Phi_{\emptyset}(v)$ for some v or in $\Phi_j(v, v')$ for some (v, v'). We formally define extreme threshold mechanisms below, where $\Phi_{\emptyset}^{example}(v)$ and $\Phi_j^{example}(v, v')$ are given in Lemma 4.

Definition 5. *For each $v \in [\underline{v}, \bar{v}]$, an $\emptyset$-threshold mechanism $\mathbf{p} \in P_{\emptyset}(v)$ is an **extreme $\emptyset$-threshold mechanism** if $\boldsymbol{\varphi}^{\mathbf{p}} \in \Phi_{\emptyset}^{example}(v)$. For each $j \in \mathcal{I}$ and each pair of $(v, v') \in [\underline{v}, \bar{v}] \times [v, \bar{v}]$, a j-threshold mechanism $\mathbf{p} \in P_j(v, v')$ is an **extreme j-threshold mechanism** if $\boldsymbol{\varphi}^{\mathbf{p}} \in \Phi_j^{example}(v, v')$.*

The following proposition says that a mechanism is optimal only if it is a randomization over the optimal extreme threshold mechanisms.[20]

[20]See Ben-Porath et al. (2014) and Mylovanov and Zapechelnyuk (2017) for reduced ($n = 1$) or similar threshold

Proposition 1. *A mechanism is optimal only if it is essentially a randomization over the optimal extreme threshold mechanisms.*

By Lemma 4, we know that any extreme $\emptyset$-threshold mechanism $\mathbf{p} \in P_\emptyset(v)$ must also be an extreme j-threshold mechanism with thresholds (v, v) for some $j \in \mathcal{I}$. Then, the following proposition is an immediate corollary of Proposition 1.

Proposition 1′. *A mechanism is optimal only if it is essentially a randomization over the optimal extreme threshold mechanisms within $P \setminus P_\emptyset$.*

B.5 Optimal (Extreme Threshold) Mechanisms

In this subsection, we solve for the optimal extreme threshold mechanisms within $P \setminus P_\emptyset$. Within the extreme threshold mechanisms in $P \setminus P_\emptyset$, by Definition 5, the principal's problem is simply to choose optimal extreme $\boldsymbol{\varphi}$'s in $\Phi \setminus \Phi_\emptyset$ and, then, follow the threshold mechanisms in Definition 4 and Lemma 5.

By Lemma 4, choosing extreme $\boldsymbol{\varphi}$'s is equivalent to choosing two distinct agents i and j, and respectively two thresholds v and v' such that $v \leq v'$. Therefore, the principal's problem is written as below, in which (i) the objective function (18) is written according to Definition 4 and (ii) the constraints (19) and (20) indicate that we are indeed choosing an extreme $\boldsymbol{\varphi}$ in $\Phi \setminus \Phi_\emptyset$. For notational convenience, we let $(t-c)|_{\mathcal{I}}^{(1)}$ be the highest net value reported by agents in $\mathcal{I}$, let $(t-c)|_{\mathcal{I}}^{(2)}$ be the second highest net value reported by agents in $\mathcal{I}$, and let $(t-c)|_{\mathcal{I}\setminus\{j\}}^{(1)}$ be the highest net value reported by agents in $\mathcal{I} \setminus \{j\}$.

$$\begin{aligned}
\max_{i,j,v,v':i\neq j,v\leq v'} \quad & \varphi_i c_i + \varphi_j c_j + \int_{\mathbf{T}\setminus K(v')} (t-c)|_{\mathcal{I}}^{(1)} + (t-c)|_{\mathcal{I}}^{(2)} dF \\
& + \int_{K(v')\setminus\left[K^\emptyset(v')\cup K^j(v')\right]} (t_j - c_j) + (t-c)|_{\mathcal{I}}^{(1)} dF \\
& + \int_{\left[K^\emptyset(v')\cup K^j(v')\right]\setminus\left[K^\emptyset(v)\cup K^j(v)\right]} (t_j - c_j) + (t-c)|_{\mathcal{I}\setminus\{j\}}^{(1)} dF \\
& + \int_{K^\emptyset(v)\cup K^j(v)} (t_j - c_j) + (t_i - c_i) dF \qquad (18) \\
\text{s.t. } \; & \varphi_j F_j(v' + c_j) = \int_{K(v')\setminus K^j(v')} dF \text{ and} \qquad (19) \\
& \varphi_i F_i(v + c_i) = \int_{K^\emptyset(v)\cup K^j(v)} dF. \qquad (20)
\end{aligned}$$

Let $T_{\mathcal{I}}^j$ be the set of type profiles such that the net value of agent j is the largest among agents in $\mathcal{I}$, i.e., $T_{\mathcal{I}}^j := \{\mathbf{t} \in \mathbf{T} : t_j - c_j > t_k - c_k, \forall k \in \mathcal{I} \setminus \{j\}\}$ and $T_{\mathcal{I}\setminus\{j\}}^i$ be analogously defined. Rearranging

mechanisms.

the objective function, the principal's objective function is equivalent to the following one:

$$\max_{i,j,v,v':i\neq j,v\leq v'} \quad \varphi_j c_j + \int_{\mathbf{T}\setminus\left[K(v')\cup T^j_{\mathcal{I}}\right]}(t-c)|^{(2)}_{\mathcal{I}}dF + \int_{K(v')\cup T^j_{\mathcal{I}}}(t_j-c_j)dF$$
$$+\varphi_i c_i + \int_{\mathbf{T}\setminus\left[K^{\emptyset}(v)\cup K^j(v)\right]}(t-c)|^{(1)}_{\mathcal{I}\setminus\{j\}}dF + \int_{K^{\emptyset}(v)\cup K^j(v)}(t_i-c_i)dF.$$

The choice of v and i is actually a subproblem given v' and j, and the only constraints are $i \neq j$ and $v \leq v'$; analogously, the choice of v' and j given v and i is also a subproblem. In what follows, we first drop the constraint $v \leq v'$ and analyze the two relaxed subproblems separately. Then we aggregate the solutions for the relaxed subproblems to obtain the solution for the original problem (18)-(20).

(Relaxed) Optimal v and i. We first solve for the optimal v and i by dropping $v \leq v'$.

$$\max_{i,v:i\neq j} \varphi_i c_i + \int_{\mathbf{T}\setminus\left[K^{\emptyset}(v)\cup K^j(v)\right]}(t-c)|^{(1)}_{\mathcal{I}\setminus\{j\}}dF + \int_{K^{\emptyset}(v)\cup K^j(v)}(t_i-c_i)dF \text{ s.t. } (20). \tag{21}$$

Since $\int_{K^{\emptyset}(v)\cup K^j(v)} dF = \prod_{k\neq j} F_k(v+c_k)$, we know that $\varphi_i = \prod_{k\neq i,j} F_k(v+c_k)$. Moreover, it is straightforward to see that the objective function is independent of the type of agent j. Therefore, the subproblem is a single-good allocation problem with a set of agents $\mathcal{I}\setminus\{j\}$. By Theorem 3 of Ben-Porath et al. (2014), we know that the subproblem has the following solution.

Lemma 6. *The relaxed problem (21) has the following solution:*

1. *When $j = 1$, the optimal $i = 2$ and the optimal $v = v^*_2$.*
2. *When $j \neq 1$, the optimal $i = 1$ and the optimal $v = v^*_1$.*

(Relaxed) Optimal v' and j. Now we investigate the optimal v' and j by dropping $v \leq v'$.

$$\max_{j,v':j\neq i} \varphi_j c_j + \int_{\mathbf{T}\setminus\left[K(v')\cup T^j_{\mathcal{I}}\right]}(t-c)|^{(2)}_{\mathcal{I}}dF + \int_{K(v')\cup T^j_{\mathcal{I}}}(t_j-c_j)dF \text{ s.t. } (19). \tag{22}$$

The problem is essentially a one good allocation problem with the following rules: within the big claw defined by v' and the region where agent j's net value is above v' and is the highest among all agents, the good is allocated to agent j; otherwise, the good is allocated to the agent who has the *second highest* net value. As for the checking policies, the constraint (19) amounts to say the following: for all type profiles such that at most one agent has a net value that is above v', i.e., $|\{k \neq j : t_k - c_k > v'\}| \leq 1$, no one is checked; for all other type profiles, the agent who gets the good is checked.

This observation facilitates the comparison of the principal's payoffs under different choices of v''s, which in turn gives us a unique optimal v' as claimed in the following lemma.

Lemma 7. *For any fixed j in the relaxed problem (22), the unique optimal choice of v' is v^*_j.*

It is worth emphasizing that v^*_j is defined in the same way as in Ben-Porath et al. (2014). Nevertheless, such a threshold has a richer interpretation than illustrated in their paper; see equation (5) and the interpretation of v^*_j in Section 3.2.

(Relaxed) Small Index First. Collect the choice variables of (18)-(20) in a tuple $(j, v'; i, v)$. We proceed to compare the principal's payoffs under two tuples $(1, v_1^*; j, v_j^*)$ and $(j, v_j^*; 1, v_1^*)$. The former represents an extreme 1-threshold mechanism. Although the latter may not satisfy the constraint $v' \geq v$, we will still refer it as a j-threshold mechanism for convenience and our argument will go through. We aim to show that the former mechanism delivers a higher payoff to the principal than the latter, where the former satisfies the constraint $v' \geq v$.

Lemma 8. *For any agent $j \neq 1$, the 1-threshold mechanism $(1, v_1^*; j, v_j^*)$ delivers a higher payoff than the j-threshold mechanism $(j, v_j^*; 1, v_1^*)$, where the comparison is strict if $v_1^* > v_j^*$.*

Optimality and Uniqueness. The following proposition states the optimality of the extreme 1-threshold mechanism $(1, v_1^*; 2, v_2^*)$, with which we can prove Theorem 1 for the case $n = 2$.[21]

Proposition 2. *The extreme 1-threshold mechanism $(1, v_1^*; 2, v_2^*)$ is an optimal choice of $(j, v'; i, v)$ for (18)-(20). Moreover, it is the unique optimal mechanism up to equivalence and relabelling of agents according to decreasing order of v_i^*.*

Proof. Suppose $j \neq 1$. Then by Lemma 6, we know that in the relaxed problem $i = 1$ and $v = v_1^*$. By Lemma 7, we know that $v' = v_j^*$ in the relaxed problem. Thus, the solution for the relaxed problem is $(j, v_j^*; 1, v_1^*)$, which must perform weakly better than the optimal mechanism in the original problem. Suppose $v_1^* > v_j^*$. Lemma 8 tells us that in the relaxed problem, $(1, v_1^*; j, v_j^*)$ performs strictly better than $(j, v_j^*; 1, v_1^*)$. Then, $(1, v_1^*; j, v_j^*)$ performs strictly better than the solution to the original problem. Since $(1, v_1^*; j, v_j^*)$ is a feasible mechanism satisfying $v' > v$, we have arrived at a contradiction. Therefore, we must have $j = 1$ or $v_1^* = v_j^*$, where in the latter case we can relabel the agents to have $j = 1$. Without loss of generality, we have $j = 1$. Again, by Lemma 6, we know that $i = 2$ with $v = v_2^*$. By Lemma 7, $j = 1$ has an optimal threshold v_1^*. Hence, $(1, v_1^*; 2, v_2^*)$ is an optimal mechanism for the relaxed problem. Since $(1, v_1^*; 2, v_2^*)$ satisfies $v' \geq v$, it is optimal in the original problem. $\square$

Proof of Theorem 1 for $n = 2$. We prove the sufficiency part first. Denote the allocation rule of $(1, v_1^*; 2, v_2^*)$ by $\mathbf{p}$. By comparing Definition 1 and Definition 4, it is straightforward to verify that $\mathbf{p}$ is equivalent to $\mathbf{p}^{ASM}$. Next, we retrieve the checking policy of the mechanism $(1, v_1^*; 2, v_2^*)$, denoted by $\mathbf{q}$, by requiring that $\mathbf{E}_{\mathbf{t}_{-i}}[q_i(t_i, \mathbf{t}_{-i})] = \mathbf{E}_{\mathbf{t}_{-i}}[p_i(t_i, \mathbf{t}_{-i})] - \varphi_i^{\mathbf{p}}, \forall i \in \mathcal{I}, \forall t_i \in T_i$. Again, it is straightforward to verify that any eligible $\mathbf{q}$ that satisfies the above requirement must be equivalent to $\mathbf{q}^{ASM}$. Then, $(\mathbf{p}, \mathbf{q})$ is equivalent to the 2-ASM. By Proposition 2, the extreme 1-threshold mechanism $(1, v_1^*; 2, v_2^*)$,

[21] The proof for the general case, i.e., $n \geq 2$, is in the online appendix. Although the general proof is technically more complicated, the key ideas are all illustrated in the proof for $n = 2$, i.e., the current section.

or $(\mathbf{p}, \mathbf{q})$, is optimal. Therefore, the 2-ASM is optimal. As a result, essential randomizations of 2-ASMs are also optimal.

Now suppose a mechanism is optimal. By Proposition 1 or 1′, it is essentially one (or a randomization) of the extreme threshold mechanisms. Since the 2-ASM is the *unique* optimal extreme threshold mechanism up to equivalence and relabelling (Proposition 2), we know that the optimal mechanism must be essentially one (or a randomization) of the 2-ASMs. □

Appendix C Proof of Theorem 2

We have verified the first part of the theorem, i.e., $\mathbf{p}^{ASM}(\mathbf{t}) = \mathbf{p}^{DSM}(\mathbf{t})$, before the statement. For the second part, i.e., $\mathbf{q}^{ASM}(\mathbf{t}) \leq \mathbf{q}^{DSM}(\mathbf{t})$, it suffices to argue that given a type profile $\mathbf{t} \in \mathbf{T}$, if an agent is checked in the Ascending Algorithm, then he must also be checked in the Descending Algorithm. Suppose an agent i is checked in step k in the Ascending Algorithm. Then we must have $i \geq k$. Moreover, agent i's net value $t_i - c_i$ must be higher than v_k^* and satisfy

$$t_i - c_i > t_j - c_j \text{ for all } j \geq k \text{ and } j \neq i. \tag{23}$$

Now let us focus on the Descending Algorithm. Note that the threshold used is decreasing. When agent 1 is favored with threshold v_1^*, we know by (23) that one of the agents in $\{1, \ldots, k-1, i\}$ receives a good and leaves. As long as agent i remains, we know by (23) that one of the agents in $\{1, \ldots, k-1, i\}$ receives a good and leaves. This can happen for at most k times because $|\{1, \ldots, k-1, i\}| = k$. Therefore, agent i receives a good in one of the first k steps, where the threshold must be in $\{v_1^*, \ldots, v_k^*\}$. Since $i \geq k$, agent i is either not favored (when $i > k$) or challenged (when $i = k$ and the threshold is v_k^*). In both cases, agent i is checked when he receives a good. This completes the proof.

Appendix D Proof of Theorem 3

We need to compare (8) and (9).[22] Note that for each i and each set of distinct agents $\{i_1, \ldots, i_{\tilde{n}}\} \subset \mathcal{J}^i$, the only difference between (8) and (9) is the one between

$$\prod_{j \in \{i_1, \ldots, i_{\tilde{n}}\}} \bar{F}_j(v_i^* + c_j)$$

and

$$\int_{v_i^* + c_i}^{\bar{t}_i} \prod_{j \in \{i_1, \ldots, i_{\tilde{n}}\}} \bar{F}_j(t_i - c_i + c_j) dF_i(t_i) + F_i(v_i^* + c_i) \prod_{j \in \{i_1, \ldots, i_{\tilde{n}}\}} \bar{F}_j(v_i^* + c_j).$$

We refer the two expressions as the checking-exemption probabilities conditioning on agent i not being challenged by agents in $\mathcal{J}^i \setminus \{i_1, \ldots, i_{\tilde{n}}\}$, where the condition is represented by

$$\prod_{k \in \mathcal{J}^i \setminus \{i_1, \ldots, i_{\tilde{n}}\}} F_k(v_i^* + c_k).$$

[22] The two mechanisms are identical when $n = 1$, in which case (8) and (9) are equal.

Since $F_j(t_i - c_i + c_j) > F_j(v_i^* + c_j)$ for all $t_i > v_i^* + c_i$ and all $j \in \{i_1, \dots, i_{\tilde{n}}\}$, we have

$$\begin{aligned}
&\int_{v_i^*+c_i}^{\bar{t}_i} \prod_{j\in\{i_1,\dots,i_{\tilde{n}}\}} \bar{F}_j(t_i - c_i + c_j) dF_i(t_i) + F_i(v_i^* + c_i) \prod_{j\in\{i_1,\dots,i_{\tilde{n}}\}} \bar{F}_j(v_i^* + c_j) \\
\leq& \bar{F}_i(v_i^* + c_i) \prod_{j\in\{i_1,\dots,i_{\tilde{n}}\}} \bar{F}_j(v_i^* + c_j) + F_i(v_i^* + c_i) \prod_{j\in\{i_1,\dots,i_{\tilde{n}}\}} \bar{F}_j(v_i^* + c_j) \\
=& \prod_{j\in\{i_1,\dots,i_{\tilde{n}}\}} \bar{F}_j(v_i^* + c_j).
\end{aligned}$$

Hence, we know that (8) $\geq$ (9), where the inequality is strict if $t_i - c_i > v_i^*$ with a positive measure and if all coefficients of the conditional checking-exemption probabilities are positive, i.e., if (i) $\bar{F}_i(v_i^* + c_i) > 0$ and if (ii) $c_i > 0$, (iii) $\prod_{k\in\mathcal{J}^i\setminus\{i_1,\dots,i_{\tilde{n}}\}} F_k(v_i^* + c_k) > 0$, and (iv) $\prod_{j\in\{i_1,\dots,i_{\tilde{n}}\}} \bar{F}_j(v_i^* + c_j) > 0$.

References

Aliprantis, C. D. and Border, K. C. (2006). *Infinite Dimensional Analysis: A Hitchhiker's Guide*. Springer.

Ben-Porath, E., Dekel, E., and Lipman, B. L. (2014). Optimal allocation with costly verification. *The American Economic Review*, 104(12):3779–3813.

Ben-Porath, E., Dekel, E., and Lipman, B. L. (2019). Mechanisms with evidence: Commitment and robustness. *Econometrica*, 87(2):529–566.

Ben-Porath, E. and Lipman, B. L. (2012). Implementation with partial provability. *Journal of Economic Theory*, 147(5):1689–1724.

Border, K. C. and Sobel, J. (1987). Samurai accountant: A theory of auditing and plunder. *The Review of Economic Studies*, 54(4):525–540.

Bull, J. and Watson, J. (2007). Hard evidence and mechanism design. *Games and Economic Behavior*, 58(1):75–93.

Chua, G. A., Hu, G., and Liu, F. (2019). Optimal multi-unit allocation with costly verification. *Available at SSRN: https://ssrn.com/abstract=3407031.*

Deneckere, R. and Severinov, S. (2008). Mechanism design with partial state verifiability. *Games and Economic Behavior*, 64(2):487–513.

Doval, L. (2018). Whether or not to open pandora's box. *Journal of Economic Theory*, 175:127–158.

Dye, R. A. (1985). Disclosure of nonproprietary information. *Journal of Accounting Research*, pages 123–145.

Epitropou, M. and Vohra, R. (2019). Optimal on-line allocation rules with verification. In *International Symposium on Algorithmic Game Theory*, pages 3–17. Springer.

Erlanson, A. and Kleiner, A. (2019). A note on optimal allocation with costly verification. *Journal of Mathematical Economics*, 84:56–62.

Erlanson, A. and Kleiner, A. (2020). Costly verification in collective decisions. *Theoretical Economics*, 15(3):923–954.

Gale, D. and Hellwig, M. (1985). Incentive-compatible debt contracts: The one-period problem. *The Review of Economic Studies*, 52(4):647–663.

Glazer, J. and Rubinstein, A. (2004). On optimal rules of persuasion. *Econometrica*, 72(6):1715–1736.

Glazer, J. and Rubinstein, A. (2006). A study in the pragmatics of persuasion: a game theoretical approach. *Theoretical Economics*, 1:395–410.

Green, J. R. and Laffont, J.-J. (1986). Partially verifiable information and mechanism design. *The Review of Economic Studies*, 53(3):447–456.

Jung, W.-O. and Kwon, Y. K. (1988). Disclosure when the market is unsure of information endowment of managers. *Journal of Accounting Research*, pages 146–153.

Kartik, N. and Tercieux, O. (2012). Implementation with evidence. *Theoretical Economics*, 7(2):323–355.

Li, S. (2017). Obviously strategy-proof mechanisms. *American Economic Review*, 107(11):3257–87.

Li, Y. (2020). Mechanism design with costly verification and limited punishments. *Journal of Economic Theory*, 186.

Li, Y. (2021). Mechanism design with financially constrained agents and costly verification. *Theoretical Economics*.

Lipman, B. (2015). An elementary proof of the optimality of threshold mechanisms. *Working Paper*.

Mookherjee, D. and Png, I. (1989). Optimal auditing, insurance, and redistribution. *The Quarterly Journal of Economics*, 104(2):399–415.

Mylovanov, T. and Zapechelnyuk, A. (2017). Optimal allocation with ex post verification and limited penalties. *American Economic Review*, 107(9):2666–94.

Sher, I. and Vohra, R. (2015). Price discrimination through communication. *Theoretical Economics*, 10(2):597–648.

Townsend, R. M. (1979). Optimal contracts and competitive markets with costly state verification. *Journal of Economic Theory*, 21(2):265–293.

For Online Publication Only:

Appendix E Technical Details for Appendix B

E.1 Proof of Lemma 2.

Proof. (Only-if Part.) Suppose $\boldsymbol{\varphi}$ is feasible, i.e., there exists a feasible mechanism $\mathbf{p}$ such that (16) holds. It is straightforward to see that $\varphi_i \in [0,1]$ since $p_i(\mathbf{t}) \in [0,1]$ for all $\mathbf{t}$. Moreover,

$$\begin{aligned}\sum_i \varphi_i =& \sum_i \inf_{t_i' \in T_i} \mathbf{E}_{\mathbf{t}_{-i}} \left[p_i(t_i', \mathbf{t}_{-i})\right] \\ \leq& \sum_i \int_{T_i} \mathbf{E}_{\mathbf{t}_{-i}} \left[p_i(t_i', \mathbf{t}_{-i})\right] dF_i(t_i') \\ =& \sum_i \mathbf{E}_t \left[p_i(\mathbf{t})\right] \\ =& \mathbf{E}_t \left[\sum_i p_i(\mathbf{t})\right] \\ \leq& 2.\end{aligned}$$

(If Part.) Suppose $\varphi_i \in [0,1]$ for all i and $\sum_i \varphi_i \leq 2$. Then the mechanism $\mathbf{p}$ satisfying $p_i(\mathbf{t}) = \varphi_i$ for all $\mathbf{t}$ and all i is a feasible mechanism that derives $\boldsymbol{\varphi}$. $\square$

E.2 Proofs for Appendix B.2

Proof of Lemma 3. We consider three mutually exclusive cases.

Case 1. $\boldsymbol{\varphi} = \mathbf{0}$. Obviously, we have $\boldsymbol{\varphi} \in \Phi_\emptyset(\underline{v})$.

Case 2. There exists $j \in \mathcal{I}$ such that $\varphi_j > 0$ and $\varphi_i = 0$ for all $i \neq j$.

We investigate the following equation:

$$\varphi_j = \prod_{k \in \mathcal{I} \setminus \{j\}} F_k(v' + c_k) + \sum_{i \neq j} \left[1 - F_i(v' + c_i)\right] \prod_{k \neq i,j} F_k(v' + c_k) \tag{24}$$

Since the right-hand-side is strictly increasing in v' once it leaves 0 and before it reaches 1, (24) has a unique solution. If $v' < \underline{t}_j - c_j$, then $\boldsymbol{\varphi} \in \Phi_j(\underline{v}, v')$. If $v' \geq \underline{t}_j - c_j$, then v' is also a solution for

$$\varphi_j F_j(v' + c_j) = \int_{K(v') \setminus K^j(v')} dF.$$

In this case, again, we have $\boldsymbol{\varphi} \in \Phi_j(\underline{v}, v')$.

Case 3. There exist distinct $j, k \in \mathcal{I}$ such that $\varphi_j > 0$ and $\varphi_k > 0$.

We consider the following equation:

$$\sum_{i \in \mathcal{I}} \varphi_i F_i(\hat{v} + c_i) = \int_{K(\hat{v})} dF + \int_{K^\emptyset(\hat{v})} dF. \tag{25}$$

Obviously, $\underline{v}$ is a solution for (25), which we refer as the trivial solution. We proceed to argue that (25) has a nontrivial solution $\hat{v} > \underline{v}$. On the one hand, at $\hat{v} = \bar{v}$, we have

$$\sum_{i \in \mathcal{I}} \varphi_i F_i(\bar{v} + c_i) = \sum_{i \in \mathcal{I}} \varphi_i \leq 2 = \int_{K(\bar{v})} dF + \int_{K^\emptyset(\bar{v})} dF.$$

On the other hand, denote the second highest lower bound of the net values by, say, $\underline{t}_s - c_s$. At $\hat{v} = \underline{t}_s - c_s$, the plus slope of the right-hand-side of (25) with respect to $\hat{v}$ is zero. In contrast, at least one of $\varphi_j F_j(\hat{v} + c_j)$ and $\varphi_k F_k(\hat{v} + c_k)$ has a strictly positive plus slope at $\hat{v} = \underline{t}_s - c_s$. Therefore, for a small enough $\epsilon > 0$, we have

$$\sum_{i \in \mathcal{I}} \varphi_i F_i(\underline{t}_s - c_s + \epsilon + c_i) > \int_{K(\underline{t}_s - c_s + \epsilon)} dF + \int_{K^{\emptyset}(\underline{t}_s - c_s + \epsilon)} dF.$$

Since both sides of (25) are continuous in $\hat{v}$, we know that there exists a solution $\hat{v}$ for (25) such that $\underline{v} \leq \underline{t}_s - c_s < \hat{v} \leq \bar{v}$.

If for every $i \in \mathcal{I}$, we have

$$\varphi_i F_i(\hat{v} + c_i) \leq \int_{K(\hat{v}) \setminus K^i(\hat{v})} dF,$$

then $\boldsymbol{\varphi} \in \Phi_{\emptyset}(\hat{v})$.

Suppose there exists j such that

$$\varphi_j F_j(\hat{v} + c_j) > \int_{K(\hat{v}) \setminus K^j(\hat{v})} dF. \tag{26}$$

We define v' such that v' is the solution for

$$\varphi_j F_j(v' + c_j) = \int_{K(v') \setminus K^j(v')} dF \tag{27}$$

in $(\hat{v}, \bar{v}]$, which exists for the following reasons. On the one hand, at $v = \bar{v}$, we have

$$\varphi_j F_j(\bar{v} + c_j) = \varphi_j \leq 1 = \int_{K(\bar{v}) \setminus K^j(\bar{v})} dF.$$

On the other hand, we have (26). Since both sides of (27) are continuous in $\hat{v}$,, we know that there exists a solution for (27) in $(\hat{v}, \bar{v}]$.

Consider two mutually exclusive subcases.

Subcase 3.1. There exists $i \in \mathcal{I} \setminus \{j\}$ such that $\varphi_i > 0$ and $\varphi_k = 0$ for all $k \neq i, j$. We define v such that

$$\varphi_i = \prod_{k \neq i,j} F_k(v + c_k). \tag{28}$$

Similar argument as for the nontrivial solution for (25) shows that we have a nontrivial solution for (28), i.e., $v > \underline{v}$. If $v > \underline{t}_i - c_i$, then we know that

$$\varphi_i F_i(v + c_i) = \int_{K^{\emptyset}(v)} dF + \int_{K^j(v)} dF,$$

which implies $v < \hat{v}$ since (25) and (26) imply that

$$\varphi_i F_i(\hat{v} + c_i) < \int_{K^{\emptyset}(\hat{v})} dF + \int_{K^j(\hat{v})} dF.$$

In this case, we have $v < v'$ and thus $\boldsymbol{\varphi} \in \Phi_j(v, v')$.

If $v < \underline{t}_i - c_i$, then we know that

$$\int_{K^{\emptyset}(v)} dF + \int_{K^j(v)} dF = \prod_{k \neq j} F_k(v + c_k) = 0,$$

which implies $v < \hat{v}$ since (25) and (26) imply that

$$0 \leq \varphi_i F_i(\hat{v} + c_i) < \int_{K^{\emptyset}(\hat{v})} dF + \int_{K^j(\hat{v})} dF = \prod_{k \neq j} F_k(\hat{v} + c_k).$$

In this case, again, we have $v < v'$ and thus $\boldsymbol{\varphi} \in \Phi_j(v, v')$.

Subcase 3.2. There exist distinct $i, k \in \mathcal{I} \setminus \{j\}$ such that $\varphi_i > 0$ and $\varphi_k > 0$. We define v as the solution for

$$\sum_{i \in \mathcal{I} \setminus \{j\}} \varphi_i F_i(v + c_i) = \int_{K^{\emptyset}(v) \cup K^j(v)} dF \tag{29}$$

in $(\underline{v}, \hat{v})$, where the existence follows from similar argument as for the nontrivial solution of (25). Therefore, we have $\boldsymbol{\varphi} \in \Phi_j(v, v')$. $\square$

Proof of Lemma 4. We characterize the extreme points of $\Phi_{\emptyset}(v)$ with $v \in (\underline{v}, \bar{v})$. The characterization for the extreme points of $\Phi_j(v, v')$ is similar and much easier, which is omitted.

We first show that every $\boldsymbol{\varphi} \in \Phi_{\emptyset}^{example}(v)$ is an extreme point. Suppose to the contrary that there exist $\boldsymbol{\varphi}^1 \in \Phi_{\emptyset}(v)$ and $\boldsymbol{\varphi}^2 \in \Phi_{\emptyset}(v)$ such that $\boldsymbol{\varphi}^1 \neq \boldsymbol{\varphi}^2$ and there exists some $\lambda \in (0, 1)$ such that $\lambda \boldsymbol{\varphi}^1 + (1 - \lambda)\boldsymbol{\varphi}^2 = \boldsymbol{\varphi}$. Then we have $\varphi_k^1 = \varphi_k^2 = 0$ for all $k \neq i, j$ and that

$$\lambda \varphi_k^1 + (1 - \lambda)\varphi_k^2 = \varphi_k, k = i, j.$$

If $\varphi_j^1 \neq \varphi_j^2$, then w.l.o.g.,

$$\varphi_i^1 F_j(v + c_j) < \int_{K(v) \setminus K^j_{\mathcal{I} \setminus \{j\}}(v)} dF < \varphi_j^2 F_j(v + c_j),$$

which is a contradiction to $\boldsymbol{\varphi}^2 \in \Phi_{\emptyset}(v)$. Thus,

$$\varphi_i^1 F_j(v + c_j) = \varphi_j^2 F_j(v + c_j) = \int_{K(v) \setminus K^j_{\mathcal{I} \setminus \{j\}}(v)} dF.$$

Since $\boldsymbol{\varphi}^1 \neq \boldsymbol{\varphi}^2$, we must have $\varphi_i^1 \neq \varphi_i^2$. Then w.l.o.g.,

$$\varphi_i^1 F_i(v + c_i) < \int_{K^{\emptyset}(v)} dF + \int_{K^j_{\mathcal{I} \setminus \{j\}}(v)} dF < \varphi_i^2 F_i(v + c_i),$$

which implies that

$$\varphi_j^2 F_j(v + c_j) + \varphi_i^2 F_i(v + c_i) > \int_{K^{\emptyset}(v)} dF + \int_{K(v)} dF,$$

a contradiction to $\boldsymbol{\varphi}^2 \in \Phi_{\emptyset}(v)$.

Next we show that there is no other extreme points. Fix any feasible $\boldsymbol{\varphi}' \in \Phi_{\emptyset}(v)$. We proceed to show that $\boldsymbol{\varphi}'$ is a convex combination of the points in $\Phi_{\emptyset}^{example}(v)$. Let $\boldsymbol{\varphi}(j, i)$ denote the point such that $\varphi_k = 0$, for all $k \neq i, j$,

$$\varphi_j F_j(v + c_j) = \int_{K(v) \setminus K^j_{\mathcal{I} \setminus \{j\}}(v)} dF \text{ and } \varphi_i F_i(v + c_i) = \int_{K^{\emptyset}(v)} dF + \int_{K^j_{\mathcal{I} \setminus \{j\}}(v)} dF.$$

To simplify notations, we write F_i for $F_i(v+c_i)$ and $\int_R$ for $\int_{K(v)} dF$. We define a feasible $\boldsymbol{\varphi}^1$ as

$$\boldsymbol{\varphi}^1 := \begin{pmatrix} \frac{1}{F_1(v+c_1)} \int_{R\setminus K^1_{\mathcal{I}\setminus\{1\}}} \\ \frac{\varphi'_2}{\sum_{l\neq 1}\varphi'_l F_l}\left(\int_{K^1_{\mathcal{I}\setminus\{1\}}} + \int_{K_{\mathcal{I}}}\right) \\ \cdots \\ \frac{\varphi'_I}{\sum_{l\neq 1}\varphi'_l F_l}\left(\int_{K^1_{\mathcal{I}\setminus\{1\}}} + \int_{K_{\mathcal{I}}}\right) \end{pmatrix} = \begin{pmatrix} \frac{1}{F_1(v+c_1)} \int_{R\setminus K^1_{\mathcal{I}\setminus\{1\}}} \\ \frac{\left(\int_{K^1_{\mathcal{I}\setminus\{1\}}} + \int_{K_{\mathcal{I}}}\right)}{\sum_{l\neq 1}\varphi'_l F_l}\varphi'_2 \\ \cdots \\ \frac{\left(\int_{K^1_{\mathcal{I}\setminus\{1\}}} + \int_{K_{\mathcal{I}}}\right)}{\sum_{l\neq 1}\varphi'_l F_l}\varphi'_I \end{pmatrix}.$$

Then we have

$$\boldsymbol{\varphi}^1 = \frac{\varphi'_2 F_2}{\sum_{l\neq 1}\varphi'_l F_l}\boldsymbol{\varphi}(1,2) + \cdots + \frac{\varphi'_I F_I}{\sum_{l\neq 1}\varphi'_l F_l}\boldsymbol{\varphi}(1,I),$$

i.e., $\boldsymbol{\varphi}^1$ is a convex combination of $\boldsymbol{\varphi}(i,j)$'s. Similarly, we define $\boldsymbol{\varphi}^i$ for other i's. Then, the matrix $M := (\boldsymbol{\varphi}^1, \ldots, \boldsymbol{\varphi}^I)$ has full rank unless $\boldsymbol{\varphi}$ is identical to one of the $\boldsymbol{\varphi}^i$'s, in which case we are done. In the other case, full rank of M implies that $M\beta = \boldsymbol{\varphi}'$ has a solution β. Moreover, $\beta_i \geq 0$ since $\varphi_i F_i(v+c_i) \leq \int_{R\setminus K^i_{\mathcal{I}\setminus\{i\}}}$, which means that $\boldsymbol{\varphi}'$ is a convex combination of the $\boldsymbol{\varphi}$'s in $\Phi^{example}_{\emptyset}(v)$. ☐

E.3 Proofs for Appendix B.3

Proof of Lemma 5. For $j = \emptyset$, we prove the lemma by two parts.

Part 1. Construction We construct $\mathbf{p}'$ according to rule *1* of Definition 3 on the domain $\mathbf{T}\setminus K(v)$. On the domain $K(v)$, $\mathbf{p}'$ is constructed by rule *2* and the algorithm below, which "squeezes" $\sum_{i\in\mathcal{I}}\varphi_i F_i(v+c_i)$ into the "capacity-constrained claw." The number $\varphi_i F_i(v+c_i)$ for each agent i represents a probability mass with which agent i receives one good when her net value is no more than v. The *capacity constraints* within the claw mean that (i) for net-value profiles in the palm, two goods are available to be allocated; and (ii) for net-value profiles in each of the fingers, only one good is available. The capacity-constrained claw has a weighted probability mass $\int_{K(v)} dF + \int_{K^{\emptyset}(v)} dF$.

$\boldsymbol{\varphi}^{\mathbf{p}}$ SATISFYING ALGORITHM.

Set $\boldsymbol{\phi} := \boldsymbol{\varphi}^{\mathbf{p}}$. *Set*

$$\mathbf{O} := \left(\int_{K^1(v)} dF, \ldots, \int_{K^i(v)} dF, 2\int_{K^{\emptyset}(v)} dF\right).$$

For $i = 1, \ldots, I$,

Step i. *This step has up to* $I+1$ *rounds. For each round* $k = i+1, \ldots, I, 1, \ldots, i-1, \emptyset$, *we consider two mutually exclusive cases.*

1. *If* $\phi_i F_i(v+c_i) > O_k > 0$, *then set* $p'_i(\mathbf{t}) := \frac{O_k}{\int_{K^k(v)} dF}$ *for all* $\mathbf{t} \in K^k(v)$.

   Set $\phi_i := \phi_i - \frac{O_k}{F_i(v+c_i)}$. Set $O_k := 0$. If $\phi_i F_i(v+c_i) > 0$, proceed. If $\phi_i F_i(v+c_i) = 0$ and $i < I$, go to Step $i+1$; if $\phi_i F_i(v+c_i) = 0$ and $i = I$, stop.

2. *If* $\phi_i F_i(v+c_i) \in (0, O_k]$, *then set* $p'_i(\mathbf{t}) = \frac{\phi_i F_i(v+c_i)}{\int_{K^k(v)} dF}$ *for all* $\mathbf{t} \in K^k(v)$.

Set $O_k := O_k - \phi_i F_i(v+c_i)$. Set $\phi_i := 0$ and $k_i^* = k$, i.e., k_i^* is the round index where Step i is completed. If $i < I$, go to Step $i+1$; otherwise, stop.

Obviously, $p_i'(\mathbf{t}) \in [0,1]$ for all i and all $\mathbf{t} \in K^k(v)$. Since $\mathbf{p} \in P_\emptyset(v)$, we know that $\boldsymbol{\varphi}^{\mathbf{p}} \in \Phi_\emptyset(v)$. Therefore, as long as the 1-st step goes to round $k = \emptyset$, we must have $\phi_1 F_1(v+c_1) \leq \int_{K^\emptyset(v)} dF$. Induction shows that as long as the i-th step goes to round $k = \emptyset$, we must have $\phi_i F_i(v+c_i) \leq \min\left\{O_{I+1}, \int_{K^\emptyset(v)} dF\right\}$. Therefore, $\mathbf{p}'$ is feasible on $K(v)$. It is straightforward to verify that $\mathbf{p}'$ is feasible on $\mathbf{T} \setminus K(v)$. Thus, $\mathbf{p}'$ is a feasible mechanism.

Part 2. Verification We proceed to show rule *3* of Definition 3 and $\boldsymbol{\varphi}^{\mathbf{p}'} = \boldsymbol{\varphi}^{\mathbf{p}}$ are satisfied. For each i and each $t_i - c_i \in [\underline{v}, v]$, we know by rule *1* that agent i gets some fraction of one good only in the region $K(v) \setminus K^i(v)$. Since $\mathbf{E}_{\mathbf{t}_{-i}}[p_i'(t_i, \mathbf{t}_{-i})]$ is independent of t_i in the algorithm, we know that

$$\begin{aligned}\mathbf{E}_{\mathbf{t}_{-i}}\left[p_i'(t_i, \mathbf{t}_{-i})\right] =& \frac{1}{F_i(v+c_i)} \sum_{k=i+1}^{k_i^*-1} \int_{K^k} \frac{O_k}{\int_{K^k(v)} dF} dF + \int_{K^{k_i^*}} \frac{\phi_i F_i(v+c_i)}{\int_{K^{k_i^*}(v)} dF} dF \\ =& \varphi_i^{\mathbf{p}},\end{aligned} \tag{30}$$

where the summation goes by the order $k = i+1, \ldots, I, 1, \ldots, i-1, \emptyset$. Hence, rule *3* is satisfied.

For agent i, if $t_i - c_i \in (v, \bar{v}]$, then agent i gets one good if and only if it has the highest or second highest net value. Thus,

$$\begin{aligned}\mathbf{E}_{\mathbf{t}_{-i}}[p_i'(t_i, \mathbf{t}_{-i})] =& \prod_{k \in \mathcal{I}\setminus\{i\}} F_k(t_i - c_i + c_k) + \sum_{j \in \mathcal{I}\setminus\{i\}} [1 - F_j(t_i - c_i + c_j)] \prod_{k \in \mathcal{I}\setminus\{i,j\}} F_k(t_i - c_i + c_k) \\ \geq& \prod_{k \in \mathcal{I}\setminus\{i\}} F_k(v + c_k) + \sum_{j \in \mathcal{I}\setminus\{i\}} [1 - F_j(v + c_j)] \prod_{k \in \mathcal{I}\setminus\{i,j\}} F_k(v + c_k) \\ \geq& \varphi_i^{\mathbf{p}},\end{aligned} \tag{31}$$

where the second inequality follows from $\boldsymbol{\varphi}^{\mathbf{p}} \in \Phi_\emptyset(v)$ and the first inequality follows because the right-hand-side expression is increasing in $t_i - c_i$. Therefore, (30)-(31) imply that $\boldsymbol{\varphi}^{\mathbf{p}'} = \boldsymbol{\varphi}^{\mathbf{p}}$, which also implies that $\mathbf{p}'$ is a $\emptyset$-threshold mechanism with threshold v. This completes the proof for $j = \emptyset$.

For $j \neq \emptyset$, we prove the case where there are at least two agents other than j who have positive φ_i's. The other cases can be similarly proved with simpler arguments and, thus, are omitted. The proof also consists of two parts.

Part 1. Construction We define $\mathbf{p}'$ first. For every $\mathbf{t} \in K^\emptyset(v) \cup K^i(v)$, set

$$p_i'(\mathbf{t}) = \frac{\varphi_i}{\prod_{k \neq i,j} F_k(v + c_k)} \tag{32}$$

for all $i \in \mathcal{I} \setminus \{j\}$. For all $\mathbf{t} \notin K^\emptyset(v) \cup K^i(v)$, we define $\mathbf{p}'$ by applying rules *1-2* of Definition 4.

We proceed to argue that $\mathbf{p}'$ is feasible, i.e., $p_i'(\mathbf{t}) \in [0,1]$ for all i and $\mathbf{p}$, and $\sum_i p_i'(\mathbf{t}) \leq 2$ for all

$\mathbf{t}$. Note that by definition of $\Phi_j(v, v')$, we have

$$\prod_{k\neq j} F_k(v+c_k) = \sum_{i\in\mathcal{I}\setminus\{j\}} \varphi_i F_i(v+c_i) \geq \varphi_i F_i(v+c_i).$$

Thus, $p_i'(\mathbf{t}) \leq 1$ for all $i \in \mathcal{I}\setminus\{j\}$ and all $\mathbf{t} \in K^{\emptyset}(v)\cup K^i(v)$. Obviously, for agent j on $K^{\emptyset}(v)\cup K^i(v)$ and all agents on $\mathbf{T}\setminus\left[K^{\emptyset}(v)\cup K^i(v)\right]$, $p_j(\mathbf{t})$ or $p_i(\mathbf{t})$ is either 0 or 1.

For each $\mathbf{t} \in K^{\emptyset}(v)\cup K^j(v)$, we know that $p_j'(\mathbf{t}) = 1$ and that

$$\begin{aligned}\sum_{i\in\mathcal{I}\setminus\{j\}} p_i'(\mathbf{t}) = & \sum_{i\in\mathcal{I}\setminus\{j\}} \frac{\varphi_i}{\prod_{k\neq i,j} F_k(v+c_k)} \\ = & \frac{1}{\prod_{k\neq j} F_k(v+c_k)} \sum_{i\in\mathcal{I}\setminus\{j\}} \varphi_i F_i(v+c_i) \\ = & 1,\end{aligned}$$

where the last equality follows from the definition of $\Phi_j(v, v')$. Therefore, $\mathbf{p}'$ is feasible on $K^{\emptyset}(v)\cup K^j(v)$. It is straightforward to verify that $\mathbf{p}'$ is feasible on $\mathbf{T}\setminus\left[K^{\emptyset}(v)\cup K^j(v)\right]$.

Part 2. Verification We proceed to show that $\boldsymbol{\varphi}^{\mathbf{p}'} = \boldsymbol{\varphi}^{\mathbf{p}}$. To simplify notations, we denote $\mathbf{E}_{\mathbf{t}_{-i}}[p_i(t_i, \mathbf{t}_{-i})]$ by $\hat{p}_i(t_i)$. For agent j, if $t_j - c_j \in [\underline{t}_j - c_j, v']$, then by Definition 4 we have

$$\hat{p}_j'(t_j) = \prod_{k\in\mathcal{I}\setminus\{j\}} F_k(v'+c_k) + \sum_{i\in\mathcal{I}\setminus\{j\}} \left[1 - F_i(v'+c_i)\right] \prod_{k\in\mathcal{I}\setminus\{i,j\}} F_k(v'+c_k) = \varphi_j^{\mathbf{p}},$$

where the second equality follows from the definition of v'. If $t_j - c_j \in (v', \bar{v}]$, then

$$\begin{aligned}\hat{p}_j'(t_j) = & \prod_{k\in\mathcal{I}\setminus\{j\}} F_k(t_j - c_j + c_k) + \sum_{i\in\mathcal{I}\setminus\{j\}} [1 - F_i(t_j - c_j + c_i)] \prod_{k\in\mathcal{I}\setminus\{i,j\}} F_k(t_j - c_j + c_k) \\ \geq & \prod_{k\in\mathcal{I}\setminus\{j\}} F_k(v'+c_k) + \sum_{i\in\mathcal{I}\setminus\{j\}} \left[1 - F_i(v'+c_i)\right] \prod_{k\in\mathcal{I}\setminus\{i,j\}} F_k(v'+c_k) \\ = & \varphi_j^{\mathbf{p}},\end{aligned}$$

where the inequality follows because the right-hand-side expression is increasing in $t_j - c_j$.

For each $i \in \mathcal{I}\setminus\{j\}$ and each t_i with $\underline{t}_i - c_i \leq t_i - c_i \leq v$, by (32), we know that $\hat{p}_i'(t_i) = \varphi_i^{\mathbf{p}}$, i.e., rule *3* is satisfied. For each $i \in \mathcal{I}\setminus\{j\}$ and each t_i with $v \leq t_i - c_i \leq v'$, agent i obtains one good if and only if i has the highest or second highest net value in $K^{\emptyset}(v')\cup K^j(v')$. Thus,

$$\begin{aligned}\hat{p}_i'(t_i) = & \prod_{k\in\mathcal{I}\setminus\{i\}} F_k(t_i - c_i + c_k) + [1 - F_j(t_i - c_i + c_j)] \prod_{k\in\mathcal{I}\setminus\{i,j\}} F_k(t_i - c_i + c_k) \\ = & \prod_{k\in\mathcal{I}\setminus\{i,j\}} F_k(t_i - c_i + c_k) \\ \geq & \prod_{k\in\mathcal{I}\setminus\{i,j\}} F_k(v + c_k) \\ \geq & \varphi_i^{\mathbf{p}}.\end{aligned}$$

The argument for t_i with $v' < t_i - c_i \leq \bar{v}$ is similar to that for agent j and thus omitted. In summary, we have $\boldsymbol{\varphi}^{\mathbf{p}'} = \boldsymbol{\varphi}^{\mathbf{p}}$, which also implies that $\mathbf{p}'$ is a j-threshold mechanism with thresholds (v, v'). $\square$

E.4 Proofs for Appendix B.4

The lemma below will be used to prove Proposition 1. Its proof is standard and, thus, is omitted.

Lemma 9. *The principal's problem (15)-(16) is equivalent to*

$$\max_{\substack{\varphi_i \in [0,1], \\ \sum_i \varphi_i \leq 2.}} \max_{\substack{p_i : \mathbf{t} \to [0,1], \\ \sum_i p_i(\mathbf{t}) \leq 2, \forall \mathbf{t} \in \mathbf{T}.}} \mathbf{E}_{\mathbf{t}} \left[\sum_i p_i(\mathbf{t})(t_i - c_i) + \sum_i \varphi_i c_i \right] \tag{33}$$

$$s.t. \ \mathbf{E}_{\mathbf{t}_{-i}} \left[p_i(t_i, \mathbf{t}_{-i}) \right] \geq \varphi_i, \forall t_i, \forall i. \tag{34}$$

Proof of Proposition 1. We first show that if an optimal mechanism $\mathbf{p}$ lies in P_j for some $j \in \mathcal{I} \cup \{\emptyset\}$, then it is a j-threshold mechanism. Then we argue that optimal j-threshold mechanisms must be essentially (randomizations of) extreme j-threshold mechanisms, which completes the proof of the proposition. We denote the corresponding threshold mechanism of $\mathbf{p}$ by $\mathbf{p}'$, in the sense that $\boldsymbol{\varphi}^{\mathbf{p}'} = \boldsymbol{\varphi}^{\mathbf{p}}$. To simplify notations, we denote $\mathbf{E}_{\mathbf{t}_{-i}}[p_i(t_i, \mathbf{t}_{-i})]$ by $\hat{p}_i(t_i)$. In what follows, we prove only the case of $j \in \mathcal{I}$ and omit the case of $j = \emptyset$, since the argument for the latter is similar to that for the former and much easier.

Part 1. *If* $\mathbf{p} \in P_j(v, v')$ *is optimal, then* $\mathbf{p}$ *is essentially a* j*-threshold mechanism.*

We proceed to show that (i) for almost all $t_j < v' + c_j$, $\hat{p}_j(t_j) = \hat{p}'_j(t_j)$, and, moreover, for each $i \neq j$ and almost all $t_i < v + c_i$, $\hat{p}_i(t_i) \geq \hat{p}'_i(t_i)$; (ii) for almost all $t_j > v' + c_j$, $\hat{p}_j(t_j) \leq \hat{p}'_j(t_j)$, and, moreover, for each $i \neq j$ and almost all $t_i > v + c_i$, $\hat{p}_i(t_i) \leq \hat{p}'_i(t_i)$; (iii) for each i and each t_i, $\hat{p}_i(t_i) = \hat{p}'_i(t_i)$; and finally (iv) rules 1 and 2 in Definition 4 are both satisfied. Then, $\mathbf{p}$ is essentially a j-threshold mechanism.

Step (i). *For almost all* $t_j < v' + c_j$, $\hat{p}_j(t_j) = \hat{p}'_j(t_j)$*, and, moreover, for each* $i \neq j$ *and almost all* $t_i < v + c_i$, $\hat{p}_i(t_i) \geq \hat{p}'_i(t_i)$.

Consider $i \neq j$. For almost all $t_i < v + c_i$, we argue $\hat{p}_i(t_i) \geq \hat{p}'_i(t_i)$. Firstly, since $\mathbf{p}$ is optimal, it is feasible in the sense that it satisfies (34), i.e., for each i and almost all t_i, we have $\hat{p}_i(t_i) \geq \varphi_i^{\mathbf{p}}$. Secondly, since $\mathbf{p}'$ is the canonical mechanism of $\mathbf{p}$, we know that $\varphi_i^{\mathbf{p}'} = \varphi_i^{\mathbf{p}}$. Finally, since $\mathbf{p}'$ is a j-threshold mechanism, we know that $\hat{p}'(t_i) = \varphi_i^{\mathbf{p}'}$ for all $t_i < v + c_i$. Therefore, $\hat{p}_i(t_i) \geq \hat{p}'_i(t_i)$ for almost all $t_i < v + c_i$. For similar reasons, we know that for almost all $t_j < v' + c_j$, $\hat{p}_j(t_j) \geq \hat{p}'_j(t_j)$.

Suppose to the contrary that $\hat{p}_j(t_j) > \hat{p}'_j(t_j)$ for a positive measure set of $t_i < v' + c_i$.

Claim 1. *For a positive measure set of* $t_j < v' + c_j$ *we have*

$$\int_{(t_j, \mathbf{t}_{-j}) \in K(v')} p_j(t_j, \mathbf{t}_{-j}) dF_{-j}(\mathbf{t}_{-j}) > \int_{(t_j, \mathbf{t}_{-j}) \in K(v')} p'_j(t_j, \mathbf{t}_{-j}) dF_{-j}(\mathbf{t}_{-j}). \tag{35}$$

Proof. Rewrite the inequation $\hat{p}_j(t_j) > \hat{p}'_j(t_j)$ as

$$\begin{aligned}&\int_{(t_j,\mathbf{t}_{-j})\in K(v')} p_j(t_j,\mathbf{t}_{-j})dF_{-j}(\mathbf{t}_{-j}) + \int_{(t_j,\mathbf{t}_{-j})\in \mathbf{T}\setminus K(v')} p_j(t_j,\mathbf{t}_{-j})dF_{-j}(\mathbf{t}_{-j})\\ > &\int_{(t_j,\mathbf{t}_{-j})\in K(v')} p'_j(t_j,\mathbf{t}_{-j})dF_{-j}(\mathbf{t}_{-j}) + \int_{(t_j,\mathbf{t}_{-j})\in \mathbf{T}\setminus K(v')} p'_j(t_j,\mathbf{t}_{-j})dF_{-j}(\mathbf{t}_{-j}).\end{aligned} \tag{36}$$

We claim that for almost all $t_i \in D_i$,

$$\int_{(t_j,\mathbf{t}_{-j})\in \mathbf{T}\setminus K(v')} p_j(t_j,\mathbf{t}_{-j})dF_{-j}(\mathbf{t}_{-j}) \leq \int_{(t_j,\mathbf{t}_{-j})\in \mathbf{T}\setminus K(v')} p'_j(t_j,\mathbf{t}_{-j})dF_{-j}(\mathbf{t}_{-j}). \tag{37}$$

Suppose not. Then, there exists a positive measure set of $(t_j,\mathbf{t}_{-j}) \in \mathbf{T}\setminus K(v')$ such that $p_j(t_j,\mathbf{t}_{-j}) > p'_j(t_j,\mathbf{t}_{-j})$. Note that at every type profile in $\mathbf{T}\setminus K(v')$ such that $t_j < v' + c_j$, $t_j - c_j$ is never among the highest two net values. Thus, we can improve $\mathbf{p}$ by moving some assignment probability from agent j to the agents with the two highest net values. Since $\hat{p}_j(t_j) > \hat{p}'_j(t_j) \geq \varphi_j^{\mathbf{p}}$, moving a small enough amount of assignment probability this way is feasible. Since this would give a mechanism strictly better for the principal than $\mathbf{p}$ and $\mathbf{p}$ is optimal, we know this is impossible. Hence, up to sets of measure zero of t_j, (37) holds. Therefore, (36) implies that for a positive measure set of $t_j < v' + c_j$, (35) holds.

Note that within $K(v')$, $p'_j(t_j,\mathbf{t}_{-j}) = 1$ by definition. Therefore, (35) implies that

$$\int_{(t_j,\mathbf{t}_{-j})\in K(v')} p_j(t_j,\mathbf{t}_{-j})dF_{-j}(\mathbf{t}_{-j}) > \int_{(t_j,\mathbf{t}_{-j})\in K(v')} dF_{-j}(\mathbf{t}_{-j}),$$

This in turn implies that $p_j(t_j,\mathbf{t}_{-j}) > 1$ for a positive measure of type profiles, which contradicts the feasibility of $\mathbf{p}$. Therefore, for almost all $t_j < v' + c_j$, we have $\hat{p}_j(t_j) = \hat{p}'_j(t_j)$.

Step (ii). *For almost all $t_j > v' + c_j$, $\hat{p}_j(t_j) \leq \hat{p}'_j(t_j)$, and, moreover, for each $i \neq j$ and almost all $t_i > v + c_i$, $\hat{p}_i(t_i) \leq \hat{p}'_i(t_i)$.*

Suppose to the contrary that for some $i \neq j$, we have $\hat{p}_i(t_i) > \hat{p}'_i(t_i)$ for a positive measure set of t_i such that $t_i - c_i > v$. Since a sufficient condition for agent i to receive one good under $\mathbf{p}'$ when $t_i - c_i > v$ is that she has one of the two highest net values, we know that there is a positive measure set of $\mathbf{t_{-i}}$ such that agent i receives a good under $\mathbf{p}$ even though her net value is not one of the two highest net values. Since $\hat{p}_i(t_i) > \hat{p}'_i(t_i) \geq \varphi_i^{\mathbf{p}}$, on such profiles $(t_i, \mathbf{t_{-i}})$ we can (a) improve $\mathbf{p}$ by moving some assignment probability from agent i to the two agents with the highest net values and (b) maintain the constraint (34). Since this would give a mechanism strictly better for the principal than $\mathbf{p}$ and $\mathbf{p}$ is optimal, we know this is impossible. Hence, up to sets of measure zero, $\hat{p}_i(t_i) \leq \hat{p}'_i(t_i)$ all t_i such that $t_i - c_i > v$. For similar reasons, we know that for almost all $t_j > v' + c_j$, $\hat{p}_j(t_j) \leq \hat{p}'_j(t_j)$.

Step (iii). *For each i and each t_i, $\hat{p}_i(t_i) = \hat{p}'_i(t_i)$.*

By **Step (i)** and **Step (ii)**, we know that for each i, (a) for almost all $t_i < v + c_i$, $\hat{p}_i(t_i) \geq \hat{p}'_i(t_i)$; (b) for almost all $t_i > v + c_i$, $\hat{p}_i(t_i) \leq \hat{p}'_i(t_i)$. An identical argument as in Lipman (2015) can show that $\hat{p}_i(t_i) = \hat{p}'_i(t_i)$.

Step (iv). *Rules* 1 *and* 2 *in Definition 4 are both satisfied.*

Clearly, up to sets of measure zero, we have for all $i \neq j$ that $\hat{p}_i(t_i) = \varphi_i^{\mathbf{p}}$ for all $t_i < v + c_i$. Suppose for some agent $i \neq j$ and a positive measure set of $t_i > v + c_i$ that $\hat{p}_i(t_i) = \hat{p}'_i(t_i)$ but that the allocation for some profiles $\mathbf{t_{-i}}$ (positive measure) differ across p_i and p'_i given t_i. Since by Definition 4, agent i with $t_i > v + c_i$ receives one good under $\mathbf{p}'$ if and only if she has one of the two highest net values, we know that on such profiles $(t_i, \mathbf{t_{-i}})$, the mechanism $\mathbf{p}$ can be improved. Particularly, note that for all $t_i > v + c_i$, we have $\hat{p}'_i(t_i) > \varphi_i^{\mathbf{p}}$, which follows from our construction and the atomlessness of distributions, which implies that $\hat{p}_i(t_i) > \varphi_i^{\mathbf{p}}$ for almost all $t_i > v + c_i$. As a result, we can (i) improve $\mathbf{p}$ by moving some assignment probability from agent i to the two agents who have the highest net values and (ii) maintain the constraint (34). Since this would give a mechanism strictly better for the principal than $\mathbf{p}$ and $\mathbf{p}$ is optimal, we know this is impossible. Hence, for all $i \neq j$ and almost all $t_i > v + c_i$, $p_i(\mathbf{t}) = p'_i(\mathbf{t})$.

Note that at every $\mathbf{t} \in \mathbf{T} \setminus K(v')$ such that $t_j < v' + c_j$, $t_j - c_j$ is never among the highest two net values. Thus, to satisfy $\hat{p}_j(t_j) \geq \varphi_j^{\mathbf{p}}$, we must have $p_j(\mathbf{t}) = 1$ for all type profiles in $K(v')$ such that $t_j < v' + c_j$, i.e., $p_j(\mathbf{t}) = p'_j(\mathbf{t})$. For similar reasons as in the argument for agents $i \neq j$, we know that for almost all type profiles such that $t_j > v' + c_j$, $p_j(\mathbf{t}) = p'_j(\mathbf{t})$. Therefore, $\mathbf{p}$ is essentially a j-threshold mechanism.

Part 2. *An optimal mechanism in P_j must be essentially randomizations of extreme j-mechanisms.*

The principal's objective function, restricted to threshold mechanisms, is determined by the vector $\boldsymbol{\varphi}$. In particular, it is

$$
\begin{aligned}
&\varphi_j c_j + \int_{\underline{t}_j}^{v'+c_j} \varphi_j (t_j - c_j) dF_j(t_j) \\
&+ \int_{v'+c_j}^{\bar{t}_j} \begin{bmatrix} \prod_{k\neq j} F_k(t_j - c_j + c_k) + \\ \sum_{m\neq j}[1 - F_m(t_j - c_j + c_m)] \prod_{k\neq j,m} F_k(t_j - c_j + c_m) \end{bmatrix} (t_j - c_j) dF_j(t_j) \\
&+ \sum_{i \in \mathcal{I}\setminus\{j\}} \varphi_i c_i + \sum_{i \in \mathcal{I}\setminus\{j\}} \int_{\underline{t}_i}^{v+c_i} \varphi_i (t_i - c_i) dF_i(t_i) \\
&+ \sum_{i \in \mathcal{I}\setminus\{j\}} \int_{v+c_i}^{v'+c_i} \begin{bmatrix} \prod_{k\neq i} F_k(t_i - c_i + c_k) + \\ [1 - F_j(t_i - c_i + c_j)] \prod_{k\neq i,j} F_k(t_i - c_i + c_m) \end{bmatrix} (t_i - c_i) dF_i(t_i) \\
&+ \sum_{i \in \mathcal{I}\setminus\{j\}} \int_{v'+c_i}^{\bar{t}_i} \begin{bmatrix} \prod_{k\neq i} F_k(t_i - c_i + c_k) + \\ \sum_{m\neq i}[1 - F_m(t_i - c_i + c_m)] \prod_{k\neq i,m} F_k(t_i - c_i + c_m) \end{bmatrix} (t_i - c_i) dF_i(t_i).
\end{aligned}
$$

It is obvious that the objective function is linear in $\boldsymbol{\varphi}$. Moreover, it is increasing in $\boldsymbol{\varphi}$. Thus, we only need to focus on the extreme points of $\Phi_j(v, v')$, which are characterized in Lemma 4. □

E.5 Proofs for Appendix B.5

The proof of Lemma 7 is similar to that of Theorem 2 in Ben-Porath et al. (2014). Particularly, we calculate the difference in payoffs brought by two alternatives: v_j^* and another v' such that $v' > v_j^*$. Let x be the second highest net value. Without loss of generality, this means that there exists k such that $t_k - c_k = x$ and that $|i : t_i - c_i > x| = 1$. The superiority of v_j^* is obtained by examining the principal's payoffs under v_j^* and v' in three subdomains such that $x < v_j^* < v'$, $v_j^* < x < v'$, or $v_j^* < v' < x$. Symmetrically, v_j^* is better than any $v' < v_j^*$. Thus, v_j^* is the unique optimal threshold. We refer the readers to Ben-Porath et al. (2014) for details.

Here we present a proof for Lemma 8 that is interpretable. Readers who are interested could directly decompose the principal's objective function and arrive at the same conclusion.

Proof of Lemma 8. We start with a benchmark mechanism and analyze the difference between each of the candidate mechanisms and the benchmark mechanism. Let the benchmark mechanism be as follows: for each type profile, the two goods always go to the two agents with the two highest net values. An agent is (partially) checked if and only if he (partially) gets a good.

To derive either of the two candidate mechanisms, we conduct the following modification to the benchmark mechanism, which is the same for two candidates.[23] For every type profile $\mathbf{t} \in K^{\emptyset}(v_1^*) \cup K^j(v_1^*)$, agent 1 gets one good with no check; for every type profile $\mathbf{t} \in K^{\emptyset}(v_j^*) \cup K^1(v_j^*)$, agent j gets one good with no check. To do this, we let agent 1 replace the agent who has the second highest net value when agent 1 did not get one good in the benchmark mechanism, within the region $K^{\emptyset}(v_1^*) \cup K^j(v_1^*)$; within the region $[K^{\emptyset}(v_j^*) \cup K^1(v_j^*)] \cap [K^{\emptyset}(v_1^*) \cup K^j(v_1^*)]$, we let agent j replaces the agent who has the highest net value when agent j did not get one good in the benchmark mechanism; within the region $[K^{\emptyset}(v_j^*) \cup K^1(v_j^*)] \setminus [K^{\emptyset}(v_1^*) \cup K^j(v_1^*)]$ we let agent j replace the agent who has the second highest net value when agent j did not get one good in the benchmark mechanism.

Now we deal with the difference between $(1, v_1^*; j, v_j^*)$ and $(j, v_j^*; 1, v_1^*)$. First, to derive $(1, v_1^*; j, v_j^*)$, we consider the following regions: for each $m \neq 1, j$, let $\bar{K}^m(v, 1)$ be the extended finger m in the direction of agent 1 such that

$$\bar{K}^m(v, 1) := \left\{ \mathbf{t} \in \mathbf{T} : t_m > v \text{ and } \max_{k \neq 1, m} t_k - c_k < v \right\}.$$

Then $(1, v_1^*; j, v_j^*)$ is derived by letting agent 1 get one good in $\bar{K}^m(v_1^*, 1)$, for all $m \neq 1, j$, with no check. This is done by replacing the agent who has the second highest net value whenever agent 1 did not get one good in the benchmark mechanism. In the extended finger $m \neq 1, j$, the payoff change

[23] The overlap with the benchmark mechanism makes the presentation simpler without changing the conclusion. Similar remarks apply to the rest of the proof.

from the benchmark mechanism to $(1, v_1^*; j, v_j^*)$ is given by

$$
\begin{aligned}
&[1 - F_m(v_1^* + c_m)] \left[\prod_{k \neq 1, m} F_k(v_1^* + c_k) \right] c_1 \\
&+ [1 - F_m(v_1^* + c_m)] \left[\prod_{k \neq 1, m} F_k(v_1^* + c_m) \right] \int_{\underline{t}_1}^{v_1^* + c_1} (t_1 - c_1) dF_1(t_1) \\
&- [1 - F_m(v_1^* + c_m)] \sum_{i \neq m} \int_{\underline{t}_i}^{v_1^* + c_i} \left[\prod_{k \neq i, m} F_k(t_i - c_i + c_k) \right] (t_i - c_i) dF_i(t_i),
\end{aligned} \tag{38}
$$

where the first term is the saved checking cost and the difference between the second term and the third term is the gain from changing assignment.

Second, to derive $(j, v_j^*; 1, v_1^*)$, we consider the following regions: for each $m \neq 1, j$, let $\bar{K}^m(v, j)$ be the extended finger m in the direction of agent j such that

$$
\bar{K}^m(v, l) := \left\{ \mathbf{t} \in \mathbf{T} : t_m > v \text{ and } \max_{k \neq j, m} t_k - c_k < v \right\}.
$$

Then $(j, v_j^*; 1, v_1^*)$ is derived by letting agent j get one good in $\bar{K}^m(v_j^*, j)$, for all $m \neq 1, j$, with no check. This is done by (i) replacing the agent who has the second highest net value whenever agent j did not get one good in the benchmark mechanism, and (ii) replacing the agent who has the highest net value whenever $t_m - c_m \in [v_j^*, v_1^*]$ and agent j did not get one good in the benchmark mechanism. In the extended finger $m \neq 1, j$, the payoff change from the benchmark mechanism to $(j, v_j^*; 1, v_1^*)$ is given by

$$
\begin{aligned}
&[1 - F_m(v_j^* + c_m)] \left[\prod_{k \neq j, m} F_k(v_j^* + c_k) \right] c_j \\
&+ [1 - F_m(v_j^* + c_m)] \left[\prod_{k \neq j, m} F_k(v_j^* + c_m) \right] \int_{\underline{t}_j}^{v_j^* + c_j} (t_j - c_j) dF_j(t_j) \\
&- \int_{v_j^* + c_m}^{v_1^* + c_m} \left[\prod_{k \neq m} F_k(t_m - c_m + c_k) \right] (t_m - c_m) dF_m(t_m) \\
&- [1 - F_m(v_1^* + c_m)] \sum_{i \neq m} \int_{\underline{t}_i}^{v_j^* + c_i} \left[\prod_{k \neq i, m} F_k(t_i - c_i + c_k) \right] (t_i - c_i) dF_i(t_i),
\end{aligned} \tag{39}
$$

where the first term is the saved checking cost, the second term is the payoff from the new assignment rule, the third term is the assignment loss for type profiles with $t_m - c_m \in [v_j^*, v_1^*]$ and the last terms is the assignment loss for type profiles with $t_m - c_m > v_1^*$.

To show that $(1, v_1^*; j, v_j^*)$ always deliver higher payoff $(j, v_j^*; 1, v_1^*)$, it suffices to show that (38) >

(39) for every $m \neq 1, j$. Note that the first two terms of (38) can be simplified as

$$\begin{aligned}
&[1 - F_m(v_1^* + c_m)] \left[\prod_{k \neq 1,m} F_k(v_1^* + c_k)\right] c_1 \\
&+ [1 - F_m(v_1^* + c_m)] \left[\prod_{k \neq 1,m} F_k(v_1^* + c_m)\right] \int_{\underline{t}_1}^{v_1^* + c_1} (t_1 - c_1) dF_1(t_1) \\
=& [1 - F_m(v_1^* + c_m)] \left[\prod_{k \neq m} F_k(v_1^* + c_k)\right] \left[\frac{\int_{\underline{t}_1}^{v_1^* + c_1} (t_1 - c_1) dF_1(t_1)}{F_1(v_1^* + c_1)} + \frac{c_1}{F_1(v_1^* + c_1)} - c_1\right] \\
=& [1 - F_m(v_1^* + c_m)] \left[\prod_{k \neq m} F_k(v_1^* + c_k)\right] v_1^*,
\end{aligned}$$

where the second equality follows from the definition of v_1^*, i.e., (5). Then (38) becomes

$$\begin{aligned}
&[1 - F_m(v_1^* + c_m)] \left[\prod_{k \neq m} F_k(v_1^* + c_k)\right] v_1^* \\
&- [1 - F_m(v_1^* + c_m)] \sum_{i \neq m} \int_{\underline{t}_i}^{v_1^* + c_i} \left[\prod_{k \neq i,m} F_k(t_i - c_i + c_k)\right] (t_i - c_i) dF_i(t_i) \\
=& [1 - F_m(v_1^* + c_m)] \left\{\left[\prod_{k \neq m} F_k(v_1^* + c_k)\right] v_1^* - \sum_{i \neq m} \int_{\underline{t}_i}^{v_1^* + c_i} \left[\prod_{k \neq i,m} F_k(t_i - c_i + c_k)\right] (t_i - c_i) dF_i(t_i)\right\},
\end{aligned}$$

where the difference is exactly the gain from replacing all values in the region

$$\left\{\mathbf{t} \in \mathbf{T} : t_m - c_m > v_1^* \text{ and } \max_{k \neq m} t_k - c_k < v_1^*\right\} \tag{40}$$

to their upper bound v_1^*.

Similarly, (39) becomes

$$\begin{aligned}
&\left[1 - F_m(v_j^* + c_m)\right] \left[\prod_{k \neq m} F_k(v_j^* + c_k)\right] v_j^* - \int_{v_j^* + c_m}^{v_1^* + c_m} \left[\prod_{k \neq m} F_k(t_m - c_m + c_k)\right] (t_m - c_m) dF_m(t_m) \\
&- [1 - F_m(v_1^* + c_m)] \sum_{i \neq m} \int_{\underline{t}_i}^{v_j^* + c_i} \left[\prod_{k \neq i,m} F_k(t_i - c_i + c_k)\right] (t_i - c_i) dF_i(t_i).
\end{aligned}$$

Since

$$\begin{aligned}
&\int_{v_j^* + c_m}^{v_1^* + c_m} \left[\prod_{k \neq m} F_k(t_m - c_m + c_k)\right] (t_m - c_m) dF_m(t_m) \\
&\geq \left[F_m(v_1^* + c_m) - F_m(v_j^* + c_m)\right] \left[\prod_{k \neq m} F_k(v_j^* + c_k)\right] v_j^*,
\end{aligned}$$

we know that

$$
\begin{aligned}
(39) \leq& \left[1 - F_m(v_j^* + c_m)\right] \left[\prod_{k\neq m} F_k(v_j^* + c_k)\right] v_j^* - \left[F_m(v_1^* + c_m) - F_m(v_j^* + c_m)\right] \left[\prod_{k\neq m} F_k(v_j^* + c_k)\right] v_j^* \\
&- [1 - F_m(v_1^* + c_m)] \sum_{i\neq m} \int_{\underline{t}_i}^{v_j^*+c_i} \left[\prod_{k\neq i,m} F_k(t_i - c_i + c_k)\right] (t_i - c_i) dF_i(t_i) \\
=& [1 - F_m(v_1^* + c_m)] \left[\prod_{k\neq m} F_k(v_j^* + c_k)\right] v_j^* \\
&- [1 - F_m(v_1^* + c_m)] \sum_{i\neq m} \int_{\underline{t}_i}^{v_j^*+c_i} \left[\prod_{k\neq i,m} F_k(t_i - c_i + c_k)\right] (t_i - c_i) dF_i(t_i) \\
=& [1 - F_m(v_1^* + c_m)] \left\{ \left[\prod_{k\neq m} F_k(v_j^* + c_k)\right] v_j^* - \sum_{i\neq m} \int_{\underline{t}_i}^{v_j^*+c_i} \left[\prod_{k\neq i,m} F_k(t_i - c_i + c_k)\right] (t_i - c_i) dF_i(t_i) \right\}.
\end{aligned}
$$

where the difference is exactly the gain from replacing all values in the region

$$
\left\{ \mathbf{t} \in \mathbf{T} : t_m - c_m > v_1^* \text{ and } \max_{k\neq m} t_k - c_k < v_j^* \right\} \tag{41}
$$

to their upper bound v_j^*. Since (40) is a superset of (41), we know that (38) > (39), as desired. □