The Optimal Design of Countervailing Incentives*

Bing Liu†

October 31, 2025

PRELIMINARY AND INCOMPLETE

MOST UPDATED VERSION

Abstract

Countervailing incentives – arising when an agent has an incentive to understate their type at some realizations and to overstate it at others – are pervasive in many mechanism and market design settings such as resource pooling, auction with externalities, and multi-product monopoly pricing. This paper develops a unified framework for optimal mechanism design that explicitly accounts for countervailing incentives, presents an algorithm to compute the optimal mechanism and identifies conditions on agents' preferences under which the method applies. The algorithm enables empirical estimation of the designer's welfare weight by matching theoretical allocations to observed outcomes. We establish that leveraging countervailing incentives can increase designer surplus and that a 'quantity' premium is a general feature of optimal mechanisms when agents have linear utilities. Finally, we derive comparative statics for agents' worst-case payoffs: counterintuitively, uniformly worsening one agent's outside option can increase other agents' worst-case payoffs.

Keywords:
JEL-Classification:

*I thank Paul Milgrom, Ilya Segal, Ravi Jagadeesan and Alvin Roth for their invaluable mentorship and advice. I also thank Modibo Camara, Yi Chen for their especially helpful comments and suggestions.

†Department of Economics, Stanford University. Email: bingliu@stanford.edu.

1 Introduction

The market-design literature is well acquainted with *one-sided* incentives: buyers tend to understate their valuations and sellers tend to overstate their costs. We pay comparatively less attention to another kind of incentive. Consider that of a shareholder. If the buy/sell role is fixed exogenously, the shareholder has an incentive to understate their valuation when buying and to overstate it when selling. By contrast, if the buy/sell role is determined by the stated valuation, a *countervailing* incentive arises: overstating risks being assigned the buyer role and incurring a loss; understating risks being assigned the seller role and incurring a loss.

This paper argues that such countervailing incentives are prevalent and can be optimally created and designed in various market design settings by making participants de facto shareholders. The most direct example is in resource pooling. Consider land redevelopment. If the developer compensates affected residents in cash, the resident acts as a seller and has the incentive to overstate their loss. Instead, if the resident is required to report a single value for housing, and both their land contribution to project and their allocation of new units depends on that report, then the resident cannot simultaneously overstate losses and understate gains.[1] Even in environments such as one-sided auctions, where shareholding is not obvious, countervailing incentive may still exist. For example, when bidders experience externalities—as with telecommunications firms competing for spectrum licenses—a firm may understate its own valuation to reduce payment yet overstate the negative externalities from a rival's win. This suggests that the designer can strategically leverage these externalities to facilitate truthful reporting. Even when there are no externalities, the existences of multiple inside options such as differentiated products, can create useful countervailing incentives. For instance, a restaurant may offer meal combinations with varying portions of a healthy salad and a decadent crème brûlée; knowing that diners who value one tend to dislike the other, the restaurant can better elicit preferences with menus that balance opposing features.

In this paper, we develop a unified framework for optimal mechanism design that explicitly accounts for countervailing incentives, and we provide an algorithm to compute the optimal mechanism. In our model, each agent has single-dimensional private type that determines their value for an exogenous outside option, and the endogeneous inside

[1] While residents typically do not submit explicit value reports in practice, many land readjustment schemes—especially in Japan—share redevelopment gains in kind (serviced plots or new units) with affected owners, with allocations tied to each owner's contributed land (e.g., via a uniform reduction ratio). Functionally, this denominates both procurement and allocation, approximating the incentive effect described here. (Sorensen, 2000).

option – the mechanism's outcome (e.g., the level of procurement of potentially differentiated resources, the allocation of differentiated goods, and any externalities) chosen by the designer. The designer's objective is a weighted sum of the social welfare and the total transfers. Importantly, we impose a form of gross substitutability: if one agent's type decreases while all others' types remain fixed, the efficient mechanism does not reduce the welfare of the other agents. This condition is pivotal for preserving monotonicity and for ensuring Algorithm 1 effectively computes the optimal mechanism. We demonstrate that this condition holds in general partnership models, the monopoly pricing of differentiated goods, and models with externalities.

The optimal mechanism involves a point-wise *optimal* outcome based on the agents' virtual types, which is a function of their true type, and a *worst-off* type for each agent that both determines the shape of the agents' virtual types and aligns with the type where the participation constraint binds for this agent. The virtual type is a function of the agent's true type and reflects that type's marginal contribution to the designer's objective, subject to the incentive compatibility constraints. Above the worst-off type, agents have the incentive to pretend to be of a lower type, much like a buyer would pretend to have a lower value for a good; thus, the incentive compatibility constraint binds downwards, and the virtual type is lower than the true type. Below the worst-off type, agents have the incentive to pretend to be of a higher type, similar to a seller pretending to have a higher cost; therefore, the incentive compatibility constraint binds upwards, and the virtual type is higher than the true type.

Crucially, determining which type is an agent's worst-off type depends on the agent's virtual type through the point-wise optimal outcome. Conversely, the agent's virtual type is determined by their worst-off type. In classical mechanism design problems—such as auctioning a single good to agents with private values—resolving this intertwining relationship between the worst-off type and the virtual type (and thus the mechanism) is straightforward. First, higher types can always afford what lower types can, so the incentive constraint binds in only one direction. Second, when outside options are fixed and type-independent, participation constraints bind only at the lowest type: if the lowest type is willing to participate, then so are all higher types, who strictly prefer what the mechanism offers. Hence the lowest type is always the worst-off type. When outside options are type-dependent, this logic no longer applies. A higher type may decline to participate even if a lower type does—not because of affordability, but because their outside option is more attractive. In such environments, the worst-off type may lie in the interior of the type space, making the intertwining between the mechanism and worst-off types more complex. This intertwining between the mechanism and worst-off types

arises not only when outside options are type-dependent, but also when agents face multiple inside options. For instance, when the designer offers differentiated goods, an agent may be a high type for one good but a low type for another. While incentive constraints may bind in only one direction for extreme types—those who strongly prefer one good over others—types with more balanced preferences may have incentives to misreport in both directions. As with outside options, the challenge is that participation and incentive compatibility are jointly shaped by the mechanism, and the worst-off type can no longer be identified a priori.

The intertwined relationship between the mechanism and the worst-off type is resolved as follows: for each possible candidate worst-off type, one can derive the optimal mechanism assuming that this type binds the participation constraint. Among these candidate mechanisms, there exists one in which the assumed worst-off type is also the actual worst-off type under that candidate mechanism. This fixed point—a pair consisting of a mechanism and a consistent worst-off type—defines a saddle point. While deriving the optimal mechanism for a given worst-off type is often straightforward, proving the existence of such a saddle point is generally subtle and technically demanding. Moreover, although this approach has been applied in various settings, to the best of our knowledge, no existing work offers a unifying framework that identifies conditions under which the saddle point method is applicable. Additionally, beyond proving existence, no paper provides a constructive method for computing such mechanisms. This paper fills both gaps.

When there is a single-agent, searching for the saddle-point is straightforward: one can start from one end of the type space and systematically move to the other end. However, when there are multiple agents, each agent's worst-off type depends on the pointwise optimal outcome that hinges on all the agents' virtual types, which in turn depend on *all* the agents' worst-off types. Consider a scenario where the candidate 'worst-off' types fail the saddle point condition and lead two agent's worst-off types to be lower than his and her assumed worst-off types. If we attempt to increase the candidate worst-off type for one agent to correct this, it may lower his actual worst-off type. Simultaneously, this adjustment could inadvertently further increase the other agent's actual worst-off type. As a result, there is no guarantee that the search for a saddle point will converge from any given starting point. This paper identifies two monotonicities in the structure of the problem that greatly simplify the search for the saddle point. First, there is a general alignment between the designer's objective and the agents' preferences: whether from an efficiency or a revenue objective, the designer benefits from implementing better outcomes for agents of higher types. Second, as a result of the gross substitute assumption, one agent's outcome weakly improves when another agent's type is lower. However,

these two monotonicities do not completely address the convergence issue. Specifically, if the search for the saddle point results in a situation where one agent's candidate worst-off type is lower than their actual worst-off type, any attempt to adjust this might result in another agent's candidate worst-off type being lower than their actual worst-off type, creating a cycle. To make use of these two monotonicities effectively, the appropriate direction to search for the saddle point is from above. Therefore, Algorithm 1 begins with each agent's candidate worst-off type at the upper end of their type space. The specifics of the algorithm ensures that the iterated candidate worst-off types remain weakly above the actual worst-off types. Thus Algorithm 1 also has a flavor of descending clock auction that clears an asset market in expectation.

The framework delivers several new insights. First, cooperative production environments—where agents may both supply resources and consume the outcome—can generate higher designer surplus than standard exchange models. Second, even when the designer optimally commits to uniformly worse outside options, the resulting mechanism may yield higher worst-case payoffs for some agents. Third, the optimal mechanism exhibits quantity premium: the average price per unit increases with quantity. Fourth, the model can be reinterpreted as the design of a general asset market, in which the roles of suppliers and consumers are determined endogenously by the mechanism. Finally, the same algorithm used to compute the optimal mechanism also supports an empirical application: it enables estimation of the designer's welfare weight by matching model-implied allocations to observed outcomes.

Our paper relates to the literature of optimal contract under countervailing incentives. Lewis and Sappington (1989a) studies the countervailing incentive that could arise from regulating a monopolist in the setting of Baron and Myerson (1982) and shows an example when the regulator might benefit from creating countervailing incentives. Lewis and Sappington (1989b) shows how an inflexible rule, endowing the agent with a critical factor of production, creates countervailing incentives and is a feature of the optimal contract. Jullien (2000) uses a optimal control approach and studies countervailing incentives arising from type-dependent outside options in a principal-agent problem. Our paper differs from them by studying the countervailing incentives in a multi-agent setting. Our paper also relates to the literature of multi-product monopoly tracing back to Adams and Yellen (1976). The closest to our multi-product monopoly application is Loertscher and Muir (2024) who characterize the optimal mechanism for a monopolist selling horizontally differentiated products on a Hotelling line, where the buyer faces countervailing incentives. Our method to solve the problem is most similar to Nöldeke and Samuelson (2005) who provides a method to solve the binding monotonicity constraint without using optimal

control. We extend the method to a multi-agent setting, provide conditions when the method works, and supply an algorithm to find the optimal mechanism.

The paper proceeds as follows: In Section 2, we present the model, detailing the agents' characteristics, type-dependent outside and inside options, and the mechanism design problem. In Section 3, we discuss the theoretical underpinnings of the our approach. Section 4 introduces the optimal mechanism and our constructive algorithm for computing the optimal mechanism. In Section 5, we apply our framework to several practical scenarios. Proofs are in Appendix B.

2 Model

There is a designer, and a set of n agents $i \in I$. The designer offers a mechanism $m = (x, t)$ where x specifies the outcome and $t = (t_i)_{i \in I}$ describes the transfer from the agents to the designer. There is an exogenously given feasibility set $\mathcal{X}$ for the outcomes. Each agent $i \in I$ has a single-dimensional private type θ_i that determines the agent's payoff from the outcome x and transfer t_i in the following fashion:

$$u_i(m, \theta_i) = v_i(x, \theta_i) - t_i,$$

where for the main text of the paper, we assume $v_i(x, \theta_i)$ is linear in θ_i so we can write $v_i(x, \theta_i) = d_i(x)\theta_i$ for some $d_i : \mathcal{X} \to \mathbb{R}$. The designer knows $d_i(\cdot)$ and hence an ordinal preference for $x \in \mathcal{X}$ for each $i \in I$, but not θ_i and hence the cardinal preference of $i \in I$ over $x \in \mathcal{X}$.

Non-linear utility When $v_i(x, \theta_i)$ is non-linear in θ_i, the analysis in the paper still works if $v_i(x, \theta_i)$ is multiplicatively separable in x and θ_i, that is, for some $d_i : \mathcal{X} \to \mathbb{R}$ and $w_i : \Theta_i \to \mathbb{R}$, $v_i(x, \theta_i) = d_i(x)w_i(\theta_i)$. This is because $w_i(\theta_i)$ is essentially the *type* of concern. When $v_i(x, \theta_i)$ is not multiplicatively separable in x and θ_i, the intuition behind the analysis still applies. We discuss that case in detail in **??**.

Example: cooperative production *Consider a designer pools resources from a group agents to produce some output. Each agent has some resources (e.g. labour, expertise, land, capital etc) that can contribute to the production. There are L type of resources. Let $r_i = (r_i^l)_{l \in L}$ be agent i's contribution of each of the L resources and $r = (r_i)_{i \in I}$. Let $\hat{r}_i$ be agent i's resource constraint. Given the contribution r, the designer can pool the resources together can produce some outputs. The outputs are differentiated and there are G types of them. Let $q_i = (q_i^g)_{g \in G}$ be agent i's*

allocation of each of the G resources. Let $p : r \mapsto p(r)$ with $p(r) \subset \mathbb{R}^G$ be the production function. Then an outcome is $x = (r, q)$. The feasible outcome space is $\mathcal{X} = \{(r, q) : r_i \leq \hat{r}_i \quad \forall i, l, \quad \sum_{i \in I} q_i \leq p(r)\}$. Each agent's type θ_i is their private value of leisure time. The function $d_i(x)$ converts an outcome into leisure time. Instead of contributing labour, an agent can go on a vacation; instead of contributing land, an agent can sell the land and go on a vacation; when allocated an apartment of her dream, an agent works less and go on a vacation. [2]

Example: Externality *Consider a designer allocating a homogeneous good to a group of agents who care about each other's allocation, that is, when there are allocative externalities. Let $\hat{q}$ be the total supply of the good and q_i the quantity allocated to agent i. Then an outcome is $x = (q_i)_{i \in I}$ and $\mathcal{X} = \{q : \sum_{i \in I} q_i \leq \hat{q}\}$. An agent's type θ_i captures the agent's value for her* own *allocation. The function $d_i(x)$ converts the allocative externalities experienced by agent i in terms of her own allocation. For example, $d_i(x) = q_i + \sum_j e_j q_j$ where e_j captures the degree of externality that i experiences from j's allocation. If $e_j < 0$, i experiences negative externalities from j's allocation. $d_i(\cdot)$ does not need to be linear in q_i. For example, agents may experience externalities from market concentration*[3]*: $d_i(x) = q_i + \alpha \sum_j (\frac{q_j}{\hat{q}})^2$.*

Example: Differentiated goods *Consider a designer allocating a set G of differentiated goods. Let q_i^g be the quantity of good g allocated to agent i. An outcome is $x = (q_i^g)_{i \in I, g \in G}$ and $\mathcal{X} = \{q : \sum_{i \in I} q_i \leq \hat{q}\}$. Each agent's type θ_i is the agent's value for good 1. The function $d_i(q_i)$ converts all goods $g \in G$ in terms of good 1 consumption. For example, $d_i(q_1, q_2) = q_1 - q_2$ implies that a type that prefers good 1 ($\theta > 0$) does not like good 2. This captures one form of inside options that lead to countervailing incentives in truthful reporting and the worst-off type to lie in the interior of the type space.*[4]

Gross substitutes The following assumption on agents' preference is the key condition needed for Theorem 1. It establishes a substitutability among the agents. Towards stating the assumption, we define the social welfare given a type realization $\theta \in \Theta$ and a particular $x \in \mathcal{X}$ as

$$sw(x, \theta) = \sum_{i \in I} d_i(x) \theta_i.$$

[2] When agents have linear utilities in procurement and allocation and there is only one good and no externality, $u_i(m, \theta_i)$ is simply $(q_i - r_i)\theta_i - t_i$. So this becomes the standard partnership model.

[3] The Herfindahl–Hirschman index

[4] Other possible functional forms include $d_i(q_1, q_2) = \min\{q_1, q_2\}$ (complements), $d_i(q_1, q_2) = q_1^b q_2^{(1-b)}$ (Cobb-Douglas).

Let $X(\theta) = \arg\max_{x \in \mathcal{X}} sw(x, \theta)$ be the set of efficient outcomes.

Assumption 1 (Gross substitutes). *If $\theta, \theta' \in \mathbb{R}^n_+$ satisfy that $\theta_i = \theta'_i$ for all $i \neq j$ and $\theta_j > \theta'_j$, for all $x' \in X(\theta')$, there exists $x \in X(\theta)$ such that $d_i(x) \leq d_i(x')$ for all $i \neq j$.*

This assumption ensures that when agent j's type decreases, the efficient outcome does not reduce the welfare received by any other agent. This assumption is trivially satisfied in any settings where an exogenous amount of goods are allocated to agents who experience no externalities. In the cooperative production example without externalities, Assumption 1 is satisfied as long as the production set has free disposal. In the Externality example, Assumption 1 is satisfied as long as it is satisfied if own allocations matter more than others', formally: $|\frac{\partial d_i(q)}{\partial q_i}| \geq |\frac{\partial d_i(q)}{\partial q_j}|$ for any $i \neq j$. Beyond ruling out complementarity across agents, this assumption also imposes a richness condition on the outcome set $\mathcal{X}$: intuitively, if relative to x, x' move two agents in the same direction (i.e., increasing or decreasing their welfare), the set $\mathcal{X}$ must include some intermediate x'' that moves them in opposite directions. For example, Assumption 1 will be violated if $X = \{0, 1\}$ is the decision space of a public project. We provide a formal link between this assumption and a single-crossing condition on the social welfare function in Lemma C.1 in Appendix C.

The designer can issue a credible threat to each agent i for not participating that results agent i receiving a type-dependent payoff $\hat{u}_i(\theta_i)$, which determines each agent's outside option. We assume $\hat{u}_i''(\cdot) \leq 0$. In the cooperative production example (without externalities), $\hat{u}_i(\theta_i) = 0$: agents can choose to not participate and the designer can do nothing about it.[5] In the externality example, the designer can commit to to a punishment by choosing an outcome that minimizes agent i's payoff, that is, $\hat{u}_i(\theta_i) = \theta_i \min_{q \in X} d_i(q)$.

Each agent's type θ_i is independently drawn from the distribution F_i with compact support Θ_i. The distributions are regular and common knowledge. We also allow F_i to be degenerate for some i such that the type of agent i is public information.

2.1 Direct mechanism and constraints

A direct mechanism, m, maps a type report $\theta \in \Theta$ to a distribution of outcomes x, and payments t. Let $\pi(\theta) \in \Delta(\mathcal{X})$ be the distribution of outcomes $\mathcal{X}$ for a type report $\theta \in$

[5] This notion of outside option differs from the standard partnership model. (See Cramton et al. (1987); Loertscher and Wasser (2019)) In our model, the partnership contribution r is part of the outcome. The mechanism may acquire all or part of any agent's resource.

Θ. Let $d_i(\theta_i; m) = \mathbb{E}_{\pi,\theta_{-i}}[d_i(x(\theta_i, \theta_{-i}))]$ and $T_i(\theta_i) = \mathbb{E}_{\theta_{-i}}[t_i(\theta)]$. The interim expected outcome for an agent i with type θ_i is

$$\mathbb{E}_{\pi,\theta_{-i}}[\theta_i d_i(\theta_i; m) - t_i(\theta)] = \theta_i d_i(\theta_i; m) - T_i(\theta_i).$$

Feasibility A direct mechanism m is feasible iff for all $\theta \in \Theta$, $x \in \mathcal{X}$ for all $\pi(x; \theta) > 0$. Let $\mathcal{M}$ be the set of feasible direct mechanisms.

Bayesian incentive compatibility (IC) A direct mechanism m is IC iff for all $\theta_i, \theta_i' \in \Theta_i$,

$$\theta_i d_i(\theta_i; m) - T_i(\theta_i) \geq \theta_i d_i(\theta_i'; m) - T_i(\theta_i')$$

Individual Rational (IR) A direct mechanism m is IR iff for all $\theta_i \in \Theta_i$,

$$\theta_i d_i(\theta_i; m) - T_i(\theta_i) \geq \hat{u}_i(\theta_i)$$

Information $F, u, \hat{u}$ are common knowledge.

Designer's objective Let $\alpha \in [0, 1)$ be the designer's weight on welfare. Then the designer's objective is

$$\max_{m \in \mathcal{M}} \alpha \mathbb{E}_{\pi,\theta}\left[\sum_{i \in I} \theta_i d_i(\theta_i; m)\right] + (1 - \alpha)\mathbb{E}_{\pi,\theta}\left[\sum_{i \in I} t_i(\theta)\right]$$

subject to IC and IR.

3 Preliminary

This section summarizes the general approach commonly used in the literature to solve problems of this class. While the results themselves are not novel, the goal is to present a unified framework that future work can readily apply—saving time and avoiding the need to re-derive standard results. The section includes the technical details required to derive and prove the optimal mechanism. Readers primarily interested in applications may skip ahead to section 4, which is sufficient for understanding and computing the optimal mechanism.

Given any direct mechanism m, an agent i's payoff from truthfully reporting type θ_i is

$$U_i(\theta_i; m) = \theta_i d_i(\theta_i; m) - T_i(\theta_i).$$

Standard arguments in mechanism design give us the following results regarding the IC constraints:

Lemma 1 (IC)**.** *A direct mechanism m is IC iff for all $i \in I$, $\theta_i, \theta_i' \in \Theta_i$,*

$$U_i(\theta_i; m) = U_i(\theta_i'; m) + \int_{\theta_i'}^{\theta_i} d_i(y; m) dy$$

and $d_i(\theta_i; m)$ is non-decreasing in θ_i for all $\theta_i \in \Theta_i$.

and regarding the IR constraints:

Lemma 2 (Worst-off types)**.** *If a direct mechanism m satisfies that for all $i \in I$, $d_i(\cdot; m)$ is non-decreasing, then there exists a worst-off type $\hat{\theta}_i = \arg\min_{\theta_i \in [\underline{\theta}_i, \overline{\theta}_i]} U_i(\theta_i; m) - \hat{u}_i(\theta_i)$ for each $i \in I$. Moreover,*

1. *$\hat{\theta}_i = \underline{\theta}_i$ if $d_i(\theta_i; m) \geq \hat{u}_i'(\theta_i)$ for all $\theta_i \in \Theta_i$,*
2. *$\hat{\theta}_i = \overline{\theta}_i$ if $d_i(\theta_i; m) \leq \hat{u}_i'(\theta_i)$ for all $\theta_i \in \Theta_i$,*
3. *otherwise, $\hat{\theta}_i = \inf\{\theta_i \in \Theta_i : d_i(\theta_i; m) \geq \hat{u}_i'(\theta_i)\}$.*

For each $i \in I$, let

$$\hat{h}_i(\theta, \hat{\theta}_i) = \begin{cases} \frac{F_i(\theta_i)-1}{f_i(\theta_i)}, & \theta_i \geq \hat{\theta}_i \\ \frac{F_i(\theta_i)}{f_i(\theta_i)}, & \theta_i < \hat{\theta}_i \end{cases}$$

So the weighted virtual value function is $\phi_i^B(\theta_i) = \alpha\theta_i + (1-\alpha)\hat{h}_i(\theta, \underline{\theta}_i)$ and the weighted virtual cost function is $\phi_i^S(\theta_i) = \alpha\theta_i + (1-\alpha)\hat{h}_i(\theta, \overline{\theta}_i)$. The regularity condition assumes that both $\phi_i^B(\cdot)$ and $\phi_i^S(\cdot)$ are non-decreasing. Denote the conditions (1-3) in 2 as (IR') and let $\mathcal{M}^{\uparrow}$ denote the set of direct mechanisms m such that $d_i(\cdot; m)$ is non-decreasing for all $i \in I$. By Lemma 1, the designer's objective can be written as

$$\max_{m \in \mathcal{M}^{\uparrow}} \sum_{i \in I} \mathbb{E}\left[\alpha \cdot \theta_i d_i(\theta_i; m) + (1-\alpha) \cdot \hat{h}_i(\theta_i, \hat{\theta}_i) d_i(\theta_i; m)\right] - \hat{u}_i(\hat{\theta}_i)$$

subject to m and $(\hat{\theta}_i)_{i \in I}$ satisfies (IR').

For some $z_i \in [\phi_i^B(\underline{\theta}_i), \phi_i^S(\overline{\theta}_i)]$, let $\theta_i^B(z_i) = (\phi_i^B)^{-1}(z_i)$ and $\theta_i^S(z_i) = (\phi_i^S)^{-1}(z_i)$ and define

$$h_i(\theta, z_i) = \begin{cases} \frac{F_i(\theta_i)-1}{f_i(\theta_i)}, & \theta_i > \hat{\theta}_i \\ z_i - \theta_i, & \theta_i \in [\theta_i^S(z_i), \theta_i^B(z_i)], \\ \frac{F_i(\theta_i)}{f_i(\theta_i)}, & \theta_i < \hat{\theta}_i \end{cases}$$

Let $G_i(z_i) = \mathbb{E}[\theta_i + h_i(\theta_i, z_i)]$. $G_i(z_i)$ is strictly increasing in z_i. The following lemma is useful to link the ironing parameter z_i to the worst-off types $\hat{\theta}_i$, and for the later concavification/ironing procedure. [6].

Lemma 3 (Ironing parameter)**.** *For each $i \in I$ and $\hat{\theta}_i \in \Theta_i$, there exists a unique $z_i \in [\phi_i^B(\underline{\theta}_i), \phi_i^S(\overline{\theta}_i)]$, given by $z_i = G_i^{-1}(\hat{\theta}_i)$, such that*

$$\mathbb{E}[\hat{h}_i(\theta_i, \hat{\theta}_i)] = \mathbb{E}[h_i(\theta_i, z_i)].$$

Moreover, $G_i^{-1}(\cdot)$ is strictly increasing.

For the rest of the paper, we drop the notation of $G_i(\cdot)$, and with slight abuses of notation, we use $\hat{\theta}_i(z_i)$ to denote the worst-off type corresponding to the ironing parameter z_i and $z_i(\hat{\theta}_i)$ to denote the ironing parameter corresponding to the worst-off type $\hat{\theta}_i$. For any $m \in \mathcal{M}$ and $\theta \in \Theta$, let $\phi_i(\theta_i, z_i) = \alpha\theta_i + (1-\alpha)h_i(\theta_i, z_i)$. Let

$$\begin{aligned} v(m, \theta) &= \sum_{i \in I} d_i(\theta_i; m) \cdot (\alpha\theta_i + (1-\alpha)h_i(\theta_i, z_i)) \\ &= \sum_{i \in I} d_i(\theta_i; m)\phi_i(\theta_i, z_i). \end{aligned}$$

The next lemma provide a candidate solution, $m^* \in \arg\max_{m \in \mathcal{M}} v(m, \theta)$. Lemma 4 is the key intuition for why the saddle point approach works under this setup. The proof can be found in Appendix B.1

Lemma 4 (Monotonicity)**.** $m^* \in \mathcal{M}^{\uparrow}$.

Lemma 5 provides a further characterization of the candidate solution.

Lemma 5 (Pooling)**.** *There exists $m^* \in \arg\max_{m \in \mathcal{M}} \sum_{i \in I} d_i(\theta_i; m)\phi_i(\theta_i, z_i)$ such that $d_i(\theta_i; m)$ is constant for all $\theta_i \in [\theta_i^S(z_i), \theta_i^B(z_i)]$ and for all $i \in I$*

[6]These two lemmas follow from the intermediate value theorem and recognizing that is essentially $\hat{\theta}_i = \mathbb{E}[\theta_i + h_i(\theta_i, z_i)]$, a continuous and monotonically increasing function in z_i. $\mathbb{E}[\theta_i + h_i(\theta_i, \underline{z}_i)] = \underline{\theta}_i$ while $\mathbb{E}[\theta_i + h_i(\theta_i, \overline{z}_i)] = \overline{\theta}_i$, where $\underline{z}_i = \phi_i^B(\underline{\theta}_i)$ and $\overline{z}_i = \phi_i^S(\overline{\theta}_i)$

For the rest of the paper, we use $m^z = m^z$ to denote the maximizer with the pooling region and highlight the dependence on the ironing parameters $z = (z_i)_{i \in I}$.

The next lemma is the concavification/ironing result[7]:

Lemma 6 (Concavification)**.**

$$m^z \in \arg \max_{m \in \mathcal{M}^\uparrow} \max_{m \in \mathcal{M}^\uparrow} \sum_{i \in I} \mathbb{E}\left[\alpha \cdot \theta_i d_i(\theta_i; m) + (1-\alpha) \cdot \hat{h}_i(\theta_i, \hat{\theta}_i(z_i)) d_i(\theta_i; m)\right]$$

Finally, Lemma 7 summarizes the saddle-point approach.

Lemma 7 (Saddle point)**.**

$$m^z \in \arg \max_{m \in \mathcal{M}^\uparrow} \max_{m \in \mathcal{M}^\uparrow} \sum_{i \in I} \mathbb{E}\left[\alpha \cdot \theta_i d_i(\theta_i; m) + (1-\alpha) \cdot \hat{h}_i(\theta_i, \hat{\theta}_i(z_i)) d_i(\theta_i; m)\right] - u_i(\hat{\theta}_i(z_i))$$

if for all $i \in I$, $\hat{\theta}_i(z_i)$ and m^z satisfies (IR').

We provide a proof in subsection B.4.

4 Optimal mechanism

We first describe the optimal mechanism, then clarify how to compute the key parameters in the optimal mechanism with Algorithm 1.

4.1 The saddle-point/optimal mechanism

On a high level, the optimal mechanism implements a tie-breaking rule among the set of point-wise efficient outcomes according to agents' virtual types. The payment from each agent $i \in I$ is dominant strategy incentive compatible (DSIC) payment for the distribution of the efficient outcomes.

Virtual type For each agent $i \in I$, the mechanism assigns a virtual type to the agent

$$\phi_i(\theta_i, \boldsymbol{z_i}) = \begin{cases} \alpha\theta_i + (1-\alpha)\frac{F(\theta_i)}{F(\theta_i)}, & \theta_i < \theta_i^S \\ z_i, & \theta_i \in [\theta_i^S, \theta_i^B] \\ \alpha\theta_i + (1-\alpha)\frac{F(\theta_i)-1}{F(\theta_i)}, & \theta_i > \theta_i^B \end{cases}$$

[7]Previous paper provides excellent proof for this lemma, see e.g. Dworczak and Muir (2024). We provide a proof in subsection B.3

where θ_i^S, θ_i^B satisfies that $\alpha\theta_i^S + (1-\alpha)\frac{F(\theta_i^S)}{F(\theta_i^S)} = z_i$ and $\alpha\theta_i^B + (1-\alpha)\frac{F(\theta_i^B)-1}{F(\theta_i^B)} = z_i$. Note that the virtual type for each agent i depends on an ironing parameter z_i. Algorithm 1 details the algorithm to find the optimal $(z_i)_{i\in I}$.

Point-wise maximizers Given the type reports $\theta \in \Theta$ and the mapped virtual types $(\phi_i(\theta_i, z_i))_{i\in I}$, the mechanism computes the set of point-wise maximizers

$$M(\theta; z) = \arg\max_x \sum_{i\in I} d_i(x)\phi_i(\theta_i, z_i)$$

Tie-breaking rule Whenever the set of point-wise maximizers are not a single-ton, that is, $|M(\theta; z)| > 1$, the mechanism uses a tie-breaking rule, $a(\theta; z) \in \Delta(M(\theta; z))$ such that for each $x \in M(\theta; z)$, each $a^x(\theta; z)$ assigns a probability that $x \in M(\theta; z)$ is implemented when θ is reported. Let

$$\pi^{z,a}(\theta) = \begin{cases} a^x(\theta; z), & x \in M(\theta; z) \\ 0, & x \notin M(\theta; z) \end{cases}$$

So $\pi^{z,a}(\theta) \in \Delta(\mathcal{X})$ is a distribution of the outcomes $\mathcal{X}$.

DSIC payment For each $i \in I$, $d_i(\theta; \pi^{z,a}) = \mathbb{E}_{\pi^{z,a}(\theta)} d_i(x)$. Then, the DSIC payment rule is such that for all $\theta \in \Theta$ and $i \in I$,

$$t_i(\theta_i, \theta_{-i}; \pi^{z,a}) = t_i(\hat{\theta}_i(z_i), \theta_{-i}; \pi^{z,a}) + \theta_i d_i(\theta_i, \theta_{-i}; \pi^{z,a}) - \hat{\theta}_i(z_i) d_i(\hat{\theta}_i(z_i), \theta_{-i}; \pi^{z,a}) - \int_{\hat{\theta}_i(z_i)}^{\theta_i} d_i(x, \theta_{-i}; \pi^{z,a}) d$$

and $t_i(\hat{\theta}_i(z_i), \theta_{-i}; \pi^{z,a}) = \hat{\theta}_i(z_i) d_i(\hat{\theta}_i(z_i), \theta_{-i}; \pi^{z,a}) - \hat{u}_i(\hat{\theta}_i(z_i))$.

4.2 Compute the ironing parameters and tie-breaking rule

This section presents the second key contribution of the paper. While the first is to identify a broad class of mechanism design problems that admit a saddle point characterization, the second is to show that the optimal mechanism can be computed via a simple algorithm.

The goal of Algorithm 1 is to find a set of ironing parameters z and tie-breaking rule a such that $\pi^{z,a}$ and $(\hat{\theta}_i(z_i))_{i\in I}$ satisfy the individual rationality constraint (IR'). We describe the key idea here, omitting some knife-edge cases. First, the individual rationality constraint, roughly speaking, means that types whose virtual type is z_i (the *pooled* type)

receive an expected payoff that is equivalent to their outside option. So Algorithm 1 begins with the highest possible ironing parameter z_i for each agent i, such that the pooled typed receive an expected payoff higher their outside option. Then the algorithm lowers each z_i and hence lower the expected payoff to the pooled types, until all pooled types receive an expected payoff that is equivalent to their outside option.

Towards that end, we introduce a few useful notations. First, note that for all $i \in I$ and θ_i such that $\phi_i(\theta_i, z_i) = z_i$, $\mathbb{E}_{\theta_{-i}}[d_i(\theta_i, \theta_i; \pi^{z,a})]$ is the same. So we use the short-hand notation $d_i(z_i; z, a)$ to denote that quantity for each $i \in I$. Second, we define a 'worst' tie-breaking rule for each agent i.Given z, we define the worst tie-breaking rule for agent i as $a_i^z \in \arg\min_{a:a(\theta)\in\Delta(M(\theta;z))} d_i(z_i; z, a)$. Last, for each $i \in I$, we define $\tilde{z}_i = \inf\{z_i \in \mathbb{R} : d_i(z_i; z, a_i^z) \geq u_i'(\hat{\theta}_i(z_i))\}$ as the lowest ironing parameter z_i for i such that the worst-off type aligns with $\hat{\theta}_i(z_i)$ under the worst tie-breaking rule for agent i. We let $\tilde{z}_i(a) = \inf\{z_i \in \mathbb{R} : d_i(z_i; z, a) \geq u_i'(\hat{\theta}_i(z_i))\}$ as the lowest ironing parameter z_i for i such that the worst-off type aligns with $\hat{\theta}_i(z_i)$ under any tie-breaking rule a.

Algorithm 1

Initialize: For each $i \in I$, set $z_i = \phi_i^S(\overline{\theta}_i)$.
while (IR') is not satisfied for all a **do**
  while $\{i : z_i > \max\{\phi_i^B(\underline{\theta}_i), \tilde{z}_i(z_{-i})\}\} \neq \emptyset$ and (IR') is not satisfied for all a **do**
    Select any $i \in I$ such that $z_i > \max\{\phi_i^B(\underline{\theta}_i), \tilde{z}_i(z_{-i})\}$.
    Update $z_i \leftarrow \max\{\phi_i^B(\underline{\theta}_i), \tilde{z}_i(z_{-i})\}$.
    Update
$$a \in \arg\min_a \sum_{i:z_i \notin \{\phi_i^S(\overline{\theta}_i), \phi_i^B(\underline{\theta}_i)\}} |z_i - \tilde{z}_i(z_{-i})(a)|.$$
  end while
  if $\{i : z_i > \max\{\phi_i^B(\underline{\theta}_i), \tilde{z}_i(z_{-i})\}\} = \emptyset$ and (IR') is not satisfied for all a **then**
    For all $z_i = \tilde{z}_i(z_{-i})$, update $z_i \leftarrow z_i^-$
  end if
end while
return z and a

Let (z, a) be the ironing parameters and tie-breaking rule output by Algorithm 1, then

Theorem 1. *$(\pi^{a,z}, (t_i(\cdot, \pi^{a,z}))_{i\in I})$ is an optimal mechanism.*

The proof can be found in subsection B.5.

Example. We illustrate how Algorithm 1 works in a simple setting. Consider a polluting factory (agent 1) and a nearby resident (agent 2). Each outcome $x \in X$ is a two-dimensional vector $x = (x_1, x_2)$, where x_1 denotes the amount of pollution permitted to

the factory, and x_2 is the number of trees the factory is required to plant. The agents' utility functions are:

$$v_1(x, \theta_1) = (10x_1 - x_2)\theta_1, \quad v_2(x, \theta_2) = (-x_1 + 10x_2)\theta_1.$$

Pollution provides more benefit to the factory than harm to the resident, while tree planting benefits the resident more than it costs the factory. The feasible set is bounded: $x_1 \leq 1$, $x_2 \leq 1$: the factory has a maximum polluting capacity and there is limit to how many trees the factory can plant.

The types θ_1 and θ_2 are independently drawn from $U[0, 1]$, and the designer (e.g., a regulator or mediator) cares only about profit, so $\alpha = 0$. Both agents have zero outside options: $\hat{u}_i(\theta_i) = 0$. Table 1 shows the evolution of the ironing parameters (z_1, z_2) and the corresponding virtual surplus contributions $d_i(z_i \mid z)$ over iterations.

Iteration	z_1	z_2	$d_1(z_1 \mid z)$	$d_2(z_2 \mid z)$
1	2	2	9.1	9.1
2	$\frac{2}{105}$	2	0	> 0
3	$\frac{2}{105}$	$\frac{11}{105}$	0	0

Table 1: Iteration values for ironing parameters and virtual surplus.Note that this is not the unique solution. $z_1 = 2/21$ and $z_2 = 2/105$ is another set of ironing parameters that achieves the same objective value.

5 Applications

The usefulness of Algorithm 1 not only lies in computing the optimal mechanism given any mechanism design problem that falls into Assumption 1, but also in enabling one to do some interesting comparative statics. This section demonstrates that.

5.1 The value of cooperative production

Most businesses operate as platforms: they pool resources from one group of agents, transform those resources into output, and sell the output for profit—often to a different group of agents. What distinguishes cooperative production is the dual role of participants: agents may simultaneously contribute inputs and consume the final products. A salient example is land redevelopment. Such projects often involve multiple stakeholders—landowners, developers, and public entities—who pool complementary resources such as land, capital, expertise, and regulatory support to transform underutilized land into valuable real estate. In a cooperative production model, each stakeholder contributes to the process and also shares in the benefits. For instance, landowners may provide their property in exchange for a share of the developed units, which they can retain or sell. Developers offer construction and planning expertise, and government entities may provide infrastructure and regulatory facilitation. In contrast, a non-cooperative redevelopment project separates contributors from beneficiaries: a central planner acquires land, capital, and expertise from one group of agents and sells the redeveloped land to another. This section compares the value generated by cooperative production with that of standard platform models.

Let I' denote the set of agents in a cooperative production setting. To construct I', take the set of pure suppliers in I, denoted by $\hat{I}$, and define for each $i \in \hat{I}$ a corresponding agent i', who is identical to i except that they also derive value from some allocation, i.e., $d_{i'}((r_{i'}, r_{-i'}), q) > d_i((r_{i'}, r_{-i'}), 0)$ for some $(r, q) \in \mathcal{X}$. We refer to such agents as consuming suppliers and denote the set by $\hat{I}'$, and define $I' = (I \setminus \hat{I}) \cup \hat{I}'$.

We assume the designer has no credible threats: $\hat{u}_i(\theta_i) = 0$ for all $i \in I$ and $\theta_i \in \Theta_i$. To simplify exposition, assume there are no externalities, so each agent's valuation depends only on their own allocation and procurement: for all $i \in I$, and any $(r, q), (r', q') \in \mathcal{X}$, $d_i((r_i, r_{-i}), (q_i, q_{-i})) = d_i((r_i, r'_{-i}), (q_i, q'_{-i}))$. With this, we write $d_i(r_i, q_i)$ for clarity.

Let $\pi^{z,a}$ denote the optimal mechanism as characterized in Theorem 1 for agent set I, and $\pi^{z',a'}$ the corresponding mechanism for agent set I'. Define the value of the mechanism under each design as follows:

$$\mathcal{V}(I') = \alpha\, \mathbb{E}_{\pi^{z',a'},\theta}\left[\sum_{i\in I'} \theta_i d_i(\theta_i; m)\right] + (1-\alpha)\, \mathbb{E}_{\pi^{z',a'},\theta}\left[\sum_{i\in I'} t_i(\theta)\right],$$
$$\mathcal{V}(I) = \alpha\, \mathbb{E}_{\pi^{z,a},\theta}\left[\sum_{i\in I} \theta_i d_i(\theta_i; m)\right] + (1-\alpha)\, \mathbb{E}_{\pi^{z,a},\theta}\left[\sum_{i\in I} t_i(\theta)\right].$$

Proposition 1 compares the values via a revealed preference argument, and illustrates how Theorem 1 applies.

Proposition 1. *A cooperative production achieves weakly higher value than a standard platform.* $\mathcal{V}(I') \geq \mathcal{V}(I)$.

Proof. First, the optimal mechanism when the set of agents are I, $(\pi^{a,z}, (t_i(\cdot, \pi^{a,z}))_{i\in I'})$, is feasible, IR and IC for the set of agents I'. Feasibility is obvious since the amount of resources are the same. The rest is equivalent to showing that $\pi^{a,z}$ and $(\hat{\theta}_i(z_i))_{i\in I'})$ satisfy (IR'). This can be seen by interpreting $\pi^{a,z}$ as a mechanism that commits to $q_i = 0$ for all $i \in \hat{I}'$, since for each $i' \in \hat{I}'$, the utility from receiving nothing is at least as good as in the original mechanism where $d_i(r_i, q_i) = d_i(r_i, 0)$ by construction. Then the value of using $(\pi^{a,z}, (t_i(\cdot, \pi^{a,z}))_{i\in I'})$ is

$$\alpha\mathbb{E}_{\pi^{z',a'},\theta}\left[\sum_{i\in I} \theta_i d_i(\theta_i; m)\right] + (1-\alpha)\mathbb{E}_{\pi^{z',a'},\theta}\left[\sum_{i\in I} t_i(\theta)\right] = \mathcal{V}(I)$$

By optimality of $(\pi^{a',z'}, (t_i(\cdot, \pi^{a',z'}))_{i\in I'})$, $\mathcal{V}(I') \geq \mathcal{V}(I)$. □

Following exactly the same argument as that of Propositions 1, converting some pure suppliers in any set of agents I' to consuming suppliers to form a set of agents I'' always improves the value.

Proposition 2. *Converting some pure suppliers to consuming suppliers always improves the value:* $\mathcal{V}(I'') \geq \mathcal{V}(I')$.

Propositions 1 and 2 highlight a key advantage of cooperative production: when suppliers can also be compensated through the output, they can be incentivized to report their types truthfully at lower cost to the designer. A pure supplier has the incentive to exaggerate the cost of supplying. But if the supplier also values the output (e.g., a landowner who will later receive a redeveloped apartment), they simultaneously have an incentive to understate their valuation of the final good. The countervailing incentives reduces the information rents required for truth-telling. In the land redevelopment

example, a landowner may exaggerate the disutility from giving up their property, but their valuation of the new apartment is likely correlated with this disutility—possibly even perfectly so, as in our model. The designer can use this structure to acquire land at a lower cost. By leveraging the consumption value of the output to suppliers, cooperative production expands the designer's screening space and can increase surplus.

5.2 The value of outside options/credible threat

It is also informative to consider how the optimal mechanism responds to changes in agents' outside options. We distinguish between two kinds of changes: level shifts, where an agent's outside option increases uniformly across all types; and shape changes, where the outside option becomes more or less sensitive to the agent's type (i.e., the derivative changes). Both types of variation affect the designer's ability to screen agents and extract surplus.

Let $\hat{\hat{u}}_i(\cdot)$ and $\hat{u}_i(\cdot)$ denote two different exogenous outside option functions for agent i. Let $(\hat{z}, \hat{\mathcal{V}})$ and $(z, \mathcal{V})$ be the ironing parameters and corresponding optimal objective values computed by Algorithm 1 under $\hat{\hat{u}}_i(\cdot)$ and $\hat{u}_i(\cdot)$, respectively. The first result is immediate:

Proposition 3. *If $\hat{\hat{u}}_i(\theta_i) \geq \hat{u}_i(\theta_i)$ for all $\theta_i \in \Theta_i$, then the objective value weakly decreases: $\hat{\mathcal{V}} \leq \mathcal{V}$.*

Next, we study how changes in the slope of the outside option affect the mechanism. For a fixed saddle-point mechanism with ironing parameters z, define the worst-case utility for agent i as $\underline{u}_i(z) := \hat{u}_i(\hat{\theta}_i(z_i))$, where $\hat{\theta}_i(z_i)$ is the type whose participation constraint binds. We find that when an agent's outside option becomes more sensitive to type—i.e., the derivative increases—this improves the worst-case payoffs of all agents (weakly), including the agent whose outside option changed.

Proposition 4. *Suppose $\hat{\hat{u}}_i'(\theta_i) \geq \hat{u}_i'(\theta_i)$ for all $\theta_i \geq \hat{\theta}_i(z_i)$. Then:*

1. *For all $j \neq i$, $\underline{u}_j(\hat{z}) \geq \underline{u}_j(z)$.*
2. *If in addition $\hat{\hat{u}}_i(\theta_i) \geq \hat{u}_i(\theta_i)$ for all $\theta_i \geq \hat{\theta}_i(z_i)$, then $\underline{u}_i(\hat{z}) \geq \underline{u}_i(z)$.*

The key to this result is the following monotonicity lemma, which shows how type sensitivity in one agent's outside option affects the ironing parameters for all agents:

Lemma 8. *If $\hat{\hat{u}}_i'(\theta_i) \geq \hat{u}_i'(\theta_i)$ for all $\theta_i \geq \hat{\theta}_i(z_i)$, then $\hat{z}_j \geq z_j$ for all $j \in I$.*

The proof demonstrates how Algorithm 1 is useful.

Proof. We first prove Lemma 8. Applying Algorithm 1, we use z_t and $\hat{z}_t$ to denote the candidate ironing parameters while the algorithm is at its tth iteration when the outside option for agent i is $\hat{u}_i(\cdot)$ and $\hat{\hat{u}}_i(\cdot)$ respectively. Changes in z_t and $\hat{z}_t$ are the same until the first time where $z_{i,t}$ is lowered. Let that be the Tth iteration. Since $\hat{\hat{u}}_i'(\theta_i) \geq \hat{u}_i'(\theta_i)$ for all $\theta_i \geq \hat{\theta}_i(z_i)$, $\hat{z}_{i,T} \geq z_{i,T}$. By Lemma B.1 (See Appendix B.5), for all $j \in I$ and $t > T$, $\hat{z}_{j,T} \geq z_{j,T}$. This proves Lemma 8.

By Lemma 3, $\hat{\theta}_j(\hat{z}_j) \geq \hat{\theta}_j(z_j)$ for all $j \in I$. Since we assumed that the outside options are all increasing in types, so $\underline{u}_j(\hat{z}) = \hat{u}_j(\hat{\theta}_j(\hat{z}_j)) \geq \underline{u}_j(z) = \hat{u}_j(\hat{\theta}_j(z_j))$ for all $j \neq i$ and $\underline{u}_i(\hat{z}) = \hat{\hat{u}}_i(\hat{\theta}_i(\hat{z}_i)) \geq \underline{u}_i(z) = \hat{u}_i(\hat{\theta}_i(z_i))$ if in addition $\hat{\hat{u}}_i(\theta_i) \geq \hat{u}_i(\theta_i)$ for all $\theta_i \geq \hat{\theta}_i(z_i)$. □

When the worst-off type is in the interior, the incentive constraint is binding in the downward (upward) direction among the types above (below) the worst-off type. When the outside option is more sensitive in type among the higher types than among the lower types, then satisfying the downward incentive constraint becomes more costly for the designer than satisfying the upward one. Hence the optimal mechanism shifts up agent i's worst-off type. As a secondary effect of when agent i's higher worst-off type and hence higher virtual type, the mechanism now delivers more surplus to agent i crowding out the surplus to other agents. To maintain their individual rationality, the worst-off types of these other agents must also increase. This is the key mechanism behind the monotonicity result in Lemma 8.

Optimal threat. Proposition 3 implies that if the designer can credibly commit to an outside option that is uniformly worse for the agents, then doing so weakly increases the designer's objective. For example, in environments with externalities, the designer may commit to an allocation that generates the most negative externalities for non-participants—thereby discouraging agents from opting out of the mechanism.

However, Proposition 4 reveals that selecting the uniformly worst outside option does not necessarily result in all agents receiving their *worst* worst-case payoffs. When the outside option becomes more sensitive to type—particularly for one agent—this may shift their worst-off type upward. The corresponding increase in virtual type can crowd out others, tightening the allocation constraint, and paradoxically improving the worst-case payoffs of the remaining agents. Thus, the designer's choice of threat can create strategic spillovers across agents, and maximizing leverage over one agent may inadvertently benefit others.

When a uniformly worst outside option does not exist—i.e., when threats are not ranked the same way for all types—the designer's problem becomes combinatorial. The optimal threat then depends not only on the shape of each agent's outside option, but

also on how it interacts with the location of their worst-off type, which itself depends on the outside options of other agents.

5.3 'Quantity' premium

The optimal mechanism can be implemented directly: agents report their types, and an outcome $x \in M(\theta)$ is selected according to a tie-breaking rule $a(\theta)$. An alternative implementation is via a menu of bundles (T_i, d_i), where T_i is the payment and d_i is the promised expected benefit level. That is, the designer offers agent i an outcome distribution π such that $d_i = \mathbb{E}_\pi[d_i(x)]$. An agent i with type θ_i's expected payoff after choosing the bundle (T_i, d_i) is $\theta_i d_i - T_i$. So the optimal menu to each agent i will be $\mathcal{N}_i := \{(T_i(\theta_i), d_i(\theta_i) : \theta_i \in \Theta_i\}$ where

$$T_i(\theta_i) = \mathbb{E}_{\theta_{-i}}[t_i(\theta_i, \theta_{-i}; \pi^{z,a})]$$
$$d_i(\theta_i) = \mathbb{E}_{\theta_{-i}}[d_i(\theta_i, \theta_{-i}; \pi^{z,a})]$$

A natural object of interest is the price per unit of benefit, defined as $\frac{T_i}{d_i}$ for each bundle $(T_i, d_i) \in \mathcal{N}_i$. The following result shows that the optimal mechanism features quantity premium: the marginal price of benefit increases with the level of benefit.Proposition 5 states this result. We defer the proof to Appendix B.6.

Proposition 5. *The optimal menu exhibits quantity premium. That is, for all $i \in I$ and $(T_i, d_i) \in \mathcal{N}_i$, $\frac{T_i}{d_i}$ increases in d_i.*

5.4 Designing a general asset market

Another interpretation of the model is the design of a general asset market. The connection to cooperative production is natural: let $\hat{r}_i$ denote agent i's initial endowment of assets, r_i their sold assets, and q_i their procured assets through the mechanism. While this includes the standard case where agents exchange their endowments, our model goes further—allowing the allocation space q to differ from the procurement space r, so that q can represent how procured assets are ultimately used (e.g., transformed, bundled, or redistributed).

More generally, for an arbitrary outcome space $\mathcal{X}$ and an outcome $x \in \mathcal{X}$, one can interpret the sign of $d_i(x)$ as revealing an agent's role in the mechanism: agents with $d_i(x) < 0$ act as suppliers of the outcome, and those with $d_i(x) > 0$ act as consumers. For instance, in the public decision to build a polluting factory, downstream residents

experience $d_i(x) < 0$ and are effectively supplying the negative externality, while the factory itself, for whom $d_j(x) > 0$, is the consumer of that outcome.

Theorem 1 and Algorithm 1 then provide guidance for designing such a market. First, the designer collects information about the distributions F, initial resource constraints $\hat{r}$, agents' marginal utilities $(d_i)_{i\in I}$, and outside options $(\hat{u}_i)_{i\in I}$. Second, Algorithm 1 can be interpreted as clearing the market in expectation: descending from the highest feasible ironing parameter for each agent, the algorithm identifies the point at which each agent's 'expected supply', $\hat{u}_i'(\hat{\theta}_i(z_i))$, equals their 'expected demand', $d_i(z_i)$. In this sense, Algorithm 1 resembles a descending clock auction run in parallel across agents, where the individualized clock for each agent ticks down to their worst-off type—effectively serving as a personalized "price" that governs participation. Finally, the asset market is cleared either directly via the mechanism $(\pi^{z,a}, t_i(\cdot))$, or through the corresponding optimal menus $(\mathcal{N}_i)_{i\in I}$.

5.5 Estimating the designer's weight on welfare

Beyond computing the optimal mechanism for a given welfare weight, an important empirical exercise is to recover the designer's implicit welfare weight from observed outcomes. In settings where historical data on outcomes and agent characteristics are available, one can estimate the designer's preferences by finding the welfare weight that best rationalizes the observed decisions.

We outline a general procedure:

1. Estimate each agent i's utility function, assumed to take the form $v_i(x, \theta_i) = d_i(x)\theta_i$. Then, estimate the type distribution F and outside option functions $(\hat{u}_i)_{i\in I}$.

2. For periods or instances where historical outcomes are observed, estimate each agent i's realized type (e.g., using production data or market signals).

3. For a grid of welfare weights $\alpha \in [0, 1]$, use Theorem 1 and Algorithm 1 to compute the optimal outcome under the estimated type profile.

4. Identify the welfare weight that minimizes the distance between the observed outcome and the predicted optimal outcome (e.g., in ℓ^2 or another suitable norm).

As an illustration, consider a water mitigation plan. Here, r represents the quantity of water rights procured from each agent (e.g., farmers), and q represents how those rights are reallocated. For agent i, $d_i(r, q)$ captures the net mitigation benefit (e.g., water retained for crops or sold), while θ_i reflects the agent's marginal value of water, which can

be estimated using crop prices and planting choices from that year. A farmer's outside option is the profit they would earn under no mitigation agreement. By estimating these primitives and comparing actual outcomes to model-implied outcomes, one can back out the designer's weight on welfare that most closely aligns with observed allocations.

References

ADAMS, W. J. AND J. L. YELLEN (1976): "Commodity bundling and the burden of monopoly," *The quarterly journal of economics*, 90, 475–498.

BARON, D. P. AND R. B. MYERSON (1982): "Regulating a monopolist with unknown costs," *Econometrica: Journal of the Econometric Society*, 911–930.

CRAMTON, P., R. GIBBONS, AND P. KLEMPERER (1987): "Dissolving a partnership efficiently," *Econometrica: Journal of the Econometric Society*, 615–632.

DWORCZAK, P. AND E. V. MUIR (2024): "A Mechanism-Design Approach to Property Rights," Working Paper 4637366, Social Science Research Network, revise and resubmit at Econometrica.

JULLIEN, B. (2000): "Participation Constraints in Adverse Selection Models," *Journal of Economic Theory*, 93, 1–47.

LEWIS, T. R. AND D. E. SAPPINGTON (1989a): "Countervailing incentives in agency problems," *Journal of economic theory*, 49, 294–313.

——— (1989b): "Inflexible rules in incentive problems," *The American Economic Review*, 69–84.

LOERTSCHER, S. AND E. V. MUIR (2024): "Optimal Hotelling Auctions," Tech. Rep. 7083-24, MIT Sloan School of Management, Cambridge, MA, working Paper.

LOERTSCHER, S. AND C. WASSER (2019): "Optimal structure and dissolution of partnerships," *Theoretical Economics*, 14, 1063–1114.

MILGROM, P. AND C. SHANNON (1994): "Monotone comparative statics," *Econometrica*, 62, 157–180.

NÖLDEKE, G. AND L. SAMUELSON (2005): "Optimal bunching without optimal control," *Available at SSRN 785804*.

SORENSEN, A. (2000): "Conflict, consensus or consent: implications of Japanese land readjustment practice for developing countries," *Habitat international*, 24, 51–73.

A Computing the optimal mechanism

B Proofs

B.1 Proof of Lemma 4

Proof. Let $\mathcal{D} := \{(d_i(x))_{i\in I} : x \in \mathcal{X}\}$. For any $\pi(d) \in \Delta(\mathcal{D})$, there exists a direct mechanism m such that for all $\theta \in \Theta$, the distribution of $(d_i(x))_{i\in I}$ follows $\pi(d)$. Call m the induced direct mechanism by $\pi(d)$. So

$$\max_{m\in\mathcal{M}} v(m,\theta) = \max_{\pi(d)\in\Delta(\mathcal{D})} \int \sum_{i\in I} d_i\phi_i(\theta_i,z_i)\pi(d).$$

Let $D^*(\theta) = \arg\max \sum_{i\in I} d_i\phi_i(\theta_i,z_i)$ and let $d^*(\theta) \in D^*(\theta)$. Then

$$\sum_{i\in I} d_i^*(\theta)\phi_i(\theta_i,z_i) = \max_{\pi(d)\in\Delta(\mathcal{D})} \int \sum_{i\in I} d_i\phi_i(\theta_i,z_i)\pi(d).$$

For what remains, we show any $d_i^*(\theta) \in D^*(\theta)$ is non-decreasing in θ_i for all $\theta_{-i} \in \Theta_{-i}$. Then for any distribution F_{-i}, $d_i(\theta_i; m^*)$ is non-decreasing in θ_i for any m^* that is an induced direct mechanism by any $\pi(d) \in \Delta(D^*(\theta))$.

Towards showing that $d_i^*(\theta) \in D^*(\theta)$ is non-decreasing in θ_i for all $\theta_{-i} \in \Theta_{-i}$, consider the following function:

$$w(d,\theta_i) = \sum_{i\in I} d_i\phi_i(\theta_i,z_i)$$

where $w : \mathcal{D}\times\Theta_i \to \mathbb{R}$. We give $\mathcal{D}$ the lexicographic order such that for any $d, d' \in \mathcal{D}$, $d >_i d'$ if either $d_i > d_i'$ or $d_i = d_i'$ and there exists $k \neq i$ such that $d_j = d_j'$ for all $j < k$, and $d_k > d_k'$. Given this order, $\mathcal{D}$ is a chain. Moreover, $w(d,\theta_i)$ satisfies the single-crossing property in (d,θ_i): for any $d > d'$ and $\theta_i > \theta_i'$, $w(d,\theta_i') > w(d',\theta_i')$ implies $w(d,\theta_i) > w(d',\theta_i)$ because

$$\begin{aligned} w(d,\theta_i) - w(d',\theta_i) &= (d_i - d_i')\phi_i(\theta_i,z_i) + \sum_{j\neq i\in I} (d_j - d_j')\phi_j(\theta_j,z_j) \\ &= w(d,\theta_i') - w(d',\theta_i') + (d_i - d_i')\left(\phi_i(\theta_i,z_i) - \phi_i(\theta_i',z_i)\right) \end{aligned}$$

and $(d_i - d_i')\left(\phi_i(\theta_i,z_i) - \phi_i(\theta_i',z_i)\right) \geq 0$. Then $d_i^*(\theta) \in D^*(\theta)$ is non-decreasing in θ_i for all $\theta_{-i} \in \Theta_{-i}$ follows from Theorem 4 in Milgrom and Shannon (1994).

□

B.2 Proof of Lemma 5

Proof. For any $m^0 \in \arg\max_{m\in\mathcal{M}} \sum_{i\in I} d_i(\theta_i; m)\phi_i(\theta_i, z_i)$, we amend m^0 sequentially by $i = 1, 2, \ldots, n$ so that $m^i(\theta_i, \theta_{-i}) = m^{i-1}(\theta_i^S(z_i), \theta_{-i})$ for all $\theta_i \in [\theta_i^S(z_i), \theta_i^B(z_i)]$. Then $v(m^n, \theta) = v(m^0, \theta)$. So $m^n \in \arg\max_{x\in\mathcal{X}} \sum_{i\in I} d_i(\theta_i; m)\phi_i(\theta_i, z_i)$. □

B.3 Proof of Lemma 6

For any $m \in \mathcal{M}^{\uparrow}$, let

$$\hat{V}(m) = \sum_{i\in I} \mathbb{E}\left[\alpha \cdot d_i(\theta_i; m)\theta_i + (1-\alpha)\cdot \hat{h}_i(\theta_i, \hat{\theta}_i) d_i(\theta_i; m)\right]$$
$$V(m) = \sum_{i\in I} \mathbb{E}\left[\alpha \cdot d_i(\theta_i; m)\theta_i + (1-\alpha)\cdot h_i(\theta_i, z_i) d_i(\theta_i; m)\right] - \hat{u}_i(\hat{\theta}_i(z_i))$$

$$\hat{V}(m) - V(m) = \sum_{i\in I} \int_{\theta_i^S(z_i)}^{\theta_i^B(z_i)} d_i(\theta_i; m)\hat{h}_i(\theta_i, \hat{\theta}_i) dF_i(\theta_i) - \sum_{i\in I} \int_{\theta_i^S(z_i)}^{\theta_i^B(z_i)} d_i(\theta_i; m) h_i(\theta_i, z_i) dF_i(\theta_i)$$

And for each i, let $\hat{H}_i(\tau, \hat{\theta}_i) = \int_0^\tau \hat{h}_i(\theta_i, \hat{\theta}_i) dF_i(\theta_i)$ and $H_i(\tau, z_i) = \int_0^\tau h_i(\theta_i, z_i) dF_i(\theta_i)$. By construction $\hat{H}_i(\tau, \hat{\theta}_i) \geq H_i(\tau, z_i)$ for all τ. Then

$$\begin{aligned}
&\sum_{i\in I} \int_{\theta_i^S(z_i)}^{\theta_i^B(z_i)} d_i(\theta_i; m)\hat{h}_i(\theta_i, \hat{\theta}_i) dF_i(\theta_i) - \sum_{i\in I} \int_{\theta_i^S(z_i)}^{\theta_i^B(z_i)} d_i(\theta_i; m) h_i(\theta_i, z_i) dF_i(\theta_i) \\
&= d_i(\theta_i; m)\left(\hat{H}_i(\theta_i, \hat{\theta}_i) - H_i(\theta_i, z_i)\right)\Big|_{\theta_i^S(z_i)}^{\theta_i^B(z_i)} - \int_{\theta_i^S(z_i)}^{\theta_i^B(z_i)} \left(\hat{H}_i(\theta_i, \hat{\theta}_i) - H_i(\theta_i, z_i)\right) d d_i(\theta_i; m) \\
&= -\int_{\theta_i^S(z_i)}^{\theta_i^B(z_i)} \left(\hat{H}_i(\theta_i, \hat{\theta}_i) - H_i(\theta_i, z_i)\right) d d_i(\theta_i; m)
\end{aligned}$$

since $\hat{H}_i(\theta_i, \hat{\theta}_i) - H_i(\theta_i, z_i) = 0$ at $\theta_i \in \{\theta_i^S(z_i), \theta_i^B(z_i)\}$. For any $m \in \mathcal{M}^{\uparrow}$,

$$-\int_{\theta_i^S(z_i)}^{\theta_i^B(z_i)} \left(\hat{H}_i(\theta_i, \hat{\theta}_i) - H_i(\theta_i, z_i)\right) d d_i(\theta_i; m) \leq 0.$$

So $\hat{V}(m) \leq V(m)$ for any non-decreasing $d_i(\cdot; m)$. By Lemma 5

$$\int_{\theta_i^S(z_i)}^{\theta_i^B(z_i)} \left(\hat{H}_i(\theta_i, \hat{\theta}_i) - H_i(\theta_i, z_i) \right) dd_i(\theta_i; m^z) = 0,$$

establishing the optimality.

B.4 Proof of Lemma 7

Let $m \in \mathcal{M}^{\uparrow}$, denote $\tau_i(m)$ the type that satisfies (IR') and let t be such that for all $i \in I$, $U_i(\tau_i(m); m) = \hat{u}_i(\tau_i(m))$. The value of objective function given m can be written as

$$\sum_{i \in I} \mathbb{E}\left[\alpha \cdot \theta_i d_i(\theta_i; m) + (1-\alpha) \cdot \hat{h}_i(\theta_i, \tau_i(m)) d_i(\theta_i; m) \right] - \sum_{i \in I} \hat{u}_i(\tau_i(m)).$$

For all $i \in I$, let $z_i^m = z_i(\tau_i(m))$. By Lemma 6,

$$\begin{aligned}
&\sum_{i \in I} \mathbb{E}\left[\alpha \cdot \theta_i d_i(\theta_i; m^{z^m}) \theta_i d_i(\theta_i; m^{z^m}) + (1-\alpha) \cdot \hat{h}_i(\theta_i, \tau_i(m)) d_i(\theta_i; m^{z^m}) \right] - \sum_{i \in I} \hat{u}_i(\tau_i(m)) \\
\geq &\sum_{i \in I} \mathbb{E}\left[\alpha \cdot \theta_i d_i(\theta_i; m) + (1-\alpha) \cdot \hat{h}_i(\theta_i, \tau_i(m)) d_i(\theta_i; m) \right].
\end{aligned}$$

If all $i \in I$, $(\tau_i(m))_{i \in I}$ and z^{z^m} satisfies (IR'), then there exists t^{z^m} such that $U_i(\tau_i(m); m^{z^m}) = \hat{u}_i(\tau_i(m))$ for all $i \in I$. Then

$$\begin{aligned}
&\sum_{i \in I} \mathbb{E}\left[\alpha \cdot \theta_i d_i(\theta_i; m^{z^m}) + (1-\alpha) \cdot \hat{h}_i(\theta_i, \tau_i(m)) d_i(\theta_i; m^{z^m}) \right] - \sum_{i \in I} \hat{u}_i(\tau_i(m)) - \sum_{i \in I} U_i(\tau_i(m); m^{z^m}) \\
\geq &\sum_{i \in I} \mathbb{E}\left[\alpha \cdot \theta_i d_i(\theta_i; m) + (1-\alpha) \cdot \hat{h}_i(\theta_i, \tau_i(m)) d_i(\theta_i; m) \right] - \sum_{i \in I} U_i(\tau_i(m); m).
\end{aligned}$$

B.5 Proof of Theorem 1

We first provide a road map for the things that we aim to show. Let t be the t-th iteration in the inner while loop.

Claim 1: At $t = 0$, if (IR') is not satisfied for all a, then $\{i : z_i > \max\{\phi_i^B(\underline{\theta}_i), \tilde{z}_i(z_{-i})\}\} \neq \emptyset$.

- so that the outer while loop will start with the inner while loop.

Claim 2: The inner while loop will eventually end.

Claim 3: If $\{i : z_i > \max\{\phi_i^B(\underline{\theta}_i), \tilde{z}_i(z_{-i})\}\} = \emptyset$ and (IR') is not satisfied for all a when the while loop ends, we show that the if loop either returns z and a that satisfy (IR') or returns z such that $\{i : z_i > \max\{\phi_i^B(\underline{\theta}_i), \tilde{z}_i(z_{-i})\}\} \neq \emptyset$.

Claim 4: The if loop is iterated a finite number of times before (IR') is satisfied for some a.

- so that the outer while loop eventually ends.

Claim 5: If (IR') is satisfied for some a, the outer while loop ends, we are done with the proof.

Lemmas B.1 and B.2 are useful for the rest of the proof.

Lemma B.1. *$d_i(z_i; z, a)$ is non-decreasing in z_i and non-increasing in z_j for all $j \neq i$ for all a.*

Proof. The first part follows from Lemma 4. For the second part, for some $d \in \mathcal{D}$, let

$$w(d, \phi_j) = d_i z_i + d_j \phi_j + \sum_{l \neq i,j} d_l \phi_l.$$

Let $\phi_j' > \phi_j$. By Assumption 1, any $d' \in \arg\max_{d \in \mathcal{D}} w(d, \phi_j')$ and $d \in \arg\max_{d \in \mathcal{D}} w(d, \phi_j)$ satisfies that $d_i' \leq d_i$. Notice that for any $z_j' > z_j$, $\phi_j(\theta_j, z_j') \geq_{FOSD} \phi_j(\theta_j, z_j)$. Hence, $d_i(z_i; z, a)$ is non-increasing in z_j for all a. □

Lemma B.2. *$d_i(z_i; z, a_i^z)$ is non-increasing in z_j for all $j \neq i$.*

Proof. Let z' be such that $z_i' = z_i$ but $z_j' \geq z_j$ for all $j \geq i$. By definition of $a_i^{z'}$, $d_i(z_i; z', a_i^z) \geq d_i(z_i; z', a_i^{z'})$. By Lemma B.1, $d_i(z_i; z, a_i^z) \geq d_i(z_i; z', a_i^z)$. Hence $d_i(z_i; z, a_i^z) \geq d_i(z_i; z', a_i^{z'})$. □

To prove Claim 1 At $t = 0$, $z_i > \phi_i^B(\underline{\theta}_i)$ for all $i \in I$. If (IR') is satisfied for some a, then it must be that either 1) $z_i = \tilde{z}_i(a)$ or 2) $z_i < \tilde{z}_i(a)$ and $d_i(z_i; z, a) < u_i'(\hat{\theta}_i(z_i))$. If (IR') is not satisfied for all a, there exists $i \in I$, $z_i > \tilde{z}_i(a)$. Hence, if (IR') is not satisfied for all a, then $\{i : z_i > \max\{\phi_i^B(\underline{\theta}_i), \tilde{z}_i(z_{-i})\}\} \neq \emptyset$.

To prove Claim 2 Let $\underline{I}(t)$ be the set of agents whose $z_i = \phi_i^B(\underline{\theta}_i)$. So $\overline{I}(t) = I / \underline{I}(t)$ is the set of agents whose z_i might be updated in some t'-th iteration for some $t' > t$. We use $a(t)$ and $z(t)$ to refer to the tie-breaking rule and ironing parameters set in iteration t. We first show two lemmas that characterize $(z_i, d_i(z_i; z, a)_{i \in I}$ over the run of the algorithm.

Lemma B.3. *For all $t \geq 1$ and for all $i \in \underline{I}(t)$, $d_i(z_i; z(t), a(t)) \geq \hat{u}_i'(\hat{\theta}_i(z_i))$.*

Proof. For any $i \in \underline{I}(t)/\underline{I}(t-1)$, $d_i(z_i; z(t), a(t)) \geq \hat{u}_i'(\hat{\theta}_i(z_i))$ holds by construction. For any $i \in \underline{I}(t-1)$, let $t_i < t$ be such that $i \in \underline{I}(t_i)/\underline{I}(t_i - 1)$. Then $d_i(z_i; z(t_i), a_i^{z(t_i)}) \geq \hat{u}_i'(\hat{\theta}_i(z_i))$. Note that $z(t)$ is non-increasing in t. So by Lemma B.2, $d_i(z_i; z(t), a_i^{z(t)}) \geq d_i(z_i; z(t_i), a_i^{z(t_i)})$. By the definition of $a_i^{z(t)}$, $d_i(z_i; z(t), a(t)) \geq d_i(z_i; z(t), a_i^{z(t)})$. Hence $d_i(z_i; z(t), a(t)) \geq \hat{u}_i'(\hat{\theta}_i(z_i))$. □

Lemma B.4. *For all $t \geq 1$ and for all $i \in \overline{I}(t)$, there are only two cases for $(z_i, d_i(z_i; z(t), a(t))_{i \in I}$:*

1. *$d_i(z_i; z(t), a(t)) \leq \hat{u}_i'(\hat{\theta}_i(z_i))$ and $z_i = \phi_i^S(\overline{\theta}_i)$,*
2. *$d_i(z_i; z(t), a(t)) \geq \hat{u}_i'(\hat{\theta}_i(z_i))$.*

Proof. For any $i \in \overline{I}(t)$ and $z_i = \phi_i^S(\overline{\theta}_i)$, it is either $d_i(z_i; z, a) \leq \hat{u}_i'(\hat{\theta}_i(z_i))$ and $d_i(z_i; z, a(t)) \geq \hat{u}_i'(\hat{\theta}_i(z_i))$ (a tautology). We need to show that for all t there does not exist $i \in \overline{I}(t)$ and $z_i \neq \phi_i^S(\overline{\theta}_i)$ such that $d_i(z_i; z(t), a) < \hat{u}_i'(\hat{\theta}_i(z_i))$. For any $i \in \overline{I}(t)$ and $z_i \neq \phi_i^S(\overline{\theta}_i)$, $z_i = \tilde{z}_i$ and that happens in some iteration $t_i < t$. By definition of $\tilde{z}_i$, $d_i(z_i; z(t_i), a_i^{z(t_i)}) \geq \hat{u}_i'(\hat{\theta}_i(z_i))$. By Lemma B.2, $d_i(z_i; z(t), a_i^{z(t)}) \geq d_i(z_i; z(t_i), a_i^{z(t_i)})$ since $t_i < t$. By the definition of $a_i^{z(t)}$, $d_i(z_i; z(t), a(t)) \geq d_i(z_i; z(t), a_i^{z(t)})$. □

Now we show that if (IR') is not satisfied for all a for all t, then $\{i : z_i > \max\{\phi_i^B(\underline{\theta}_i), \tilde{z}_i(z_{-i})\}\} = \emptyset$ for some $t < \infty$, so that the while loop eventually ends. First, $\{i : z_i > \max\{\phi_i^B(\underline{\theta}_i), \tilde{z}_i(z_{-i})\}\} \cap \underline{I}(t) = \emptyset$ for all t. So we want to show $\{i : z_i > \max\{\phi_i^B(\underline{\theta}_i), \tilde{z}_i(z_{-i})\}\} \cap \overline{I}(t) =$ for some $t < \infty$. For any $i \in \{i : z_i > \max\{\phi_i^B(\underline{\theta}_i), \tilde{z}_i(z_{-i})\}\} \cap \overline{I}(t)$, it must be that $d_i(z_i; z(t), a_i^{z(t)} \geq \hat{u}_i'(\hat{\theta}_i(z_i))$. As long as this holds, there will be a $t' > t$ when $z_i(t') = \max\{\phi_i^B(\underline{\theta}_i), \tilde{z}_i(z_{-i})\}$. If $\phi_i^B(\underline{\theta}_i) = \max\{\phi_i^B(\underline{\theta}_i), \tilde{z}_i(z_{-i})$ at t', $i \in \underline{I}(t')$ and $\{i : z_i > \max\{\phi_i^B(\underline{\theta}_i), \tilde{z}_i(z_{-i})\}$ reduces in size. For the rest of the proof, we use $\tilde{z}_i(t) = \tilde{z}_i(z_{-i}(t))$ to simplify notation and note the dependence of $\tilde{z}_i$ on the iteration t. If $\tilde{z}_i(t') = \max\{\phi_i^B(\underline{\theta}_i), \tilde{z}_i(t')$ at t', either 1) $\tilde{z}_i(t)$ remains constant throughout the iteration, hence will not be in the set $\{i : z_i > \max\{\phi_i^B(\underline{\theta}_i), \tilde{z}_i(z_{-i})\}$, or 2) $\tilde{z}_i(z_{-i})$ reduces (because some other z_j decreases in some iteration $t'' > t'$), then there will be a $t''' > t''$ where $z_i(t') = \max\{\phi_i^B(\underline{\theta}_i), \tilde{z}_i(t')\}$. This updating stops until $\tilde{z}_i(z_{-i}) = \phi_i^B(\underline{\theta}_i)$. Then $\{i : z_i > \max\{\phi_i^B(\underline{\theta}_i), \tilde{z}_i(z_{-i})\}$ reduces in size. Hence, the set $\{i : z_i > \max\{\phi_i^B(\underline{\theta}_i), \tilde{z}_i(z_{-i})\}$ reduces in size as t increases. By its finiteness, $\{i : z_i > \max\{\phi_i^B(\underline{\theta}_i), \tilde{z}_i(z_{-i})\}\} = \emptyset$ for some $t < \infty$.

To prove Claim 3 Let $\{i : z_i > \max\{\phi_i^B(\underline{\theta}_i), \tilde{z}_i(z_{-i})\}\} = \emptyset$ at iteration T. By Lemma B.4, for all $i \in \overline{I}(T)$, either

1. $d_i(z_i; z(T), a(T)) \leq \hat{u}_i'(\hat{\theta}_i(z_i))$ and $z_i = \phi_i^S(\overline{\theta}_i)$, or

2. $z_i = \tilde{z}_i(z_{-i})$.

Let z' be such that

$$z_i' = \begin{cases} (z_i(T))^-, & z_i(T) = \tilde{z}_i(z_{-i}(T)) \\ z_i, & \text{otherwise} \end{cases}$$

Hence z' is the updated ironing parameters in the if loop. We only need to show that if $\{i : z_i' > \max\{\phi_i^B(\underline{\theta}_i), \tilde{z}_i(z'_{-i})\}\} = \emptyset$, there exists a such that (IR') is satisfied. Combining with Lemma B.3 and Lemma B.4, we only need to show that there exists a tie-breaking rule a such that $z_i' = \tilde{z}_i(a)$ for all i such that $z_i(T) = \tilde{z}_i(z_{-i}(T))$. We show that is true for $a = a(T)$.

Consider the following set of maximizers that is defined over the agents' virtual type space:

$$X(\phi) = \arg\max_{x \in \mathcal{X}} \sum_{i \in I} d_i(x)\phi_i$$

For any $i \in I$, define the following tie-breaking rule that is defined over the agents' virtual type space, $\hat{a} : \mathbb{R}^n \to \Delta(\mathcal{X})$:

$$\hat{a}_i^x(\phi) = \begin{cases} 1, & x = \arg\min_{x \in X(\phi)} d_i(x) \\ 0, & \text{otherwise.} \end{cases}$$

Then for all z and θ, $a_i^z(\theta) = \hat{a}_i(z_i, (\phi_j(\theta_j, z_j))_{i \in I})$ by the definition of a_i^z. Now we show that

Lemma B.5. *For all* (z_i, ϕ_{-i}), $X(z_i^-, \phi_{-i}) = \arg\min_{x \in X(z_i, \phi_{-i})} d_i(x)$.

Proof. If $|X(z_i, \phi_{-i})| = 1$, let $x(z_i, \phi_{-i}) = X(z_i, \phi_{-i})$. It has to be that $x(z_i^-, \phi_{-i}) = x(z_i, \phi_{-i})$. If not, either $z_i d_i(x(z_i^-, \phi_{-i})) + \sum_{j \neq i} d_j(x(z_i^-, \phi_{-i}))\phi_j > z_i d_i(x(z_i, \phi_{-i})) + \sum_{j \neq i} d_j(x(z_i, \phi_{-i}))\phi_j$ or $z_i^- d_i(x(z_i, \phi_{-i})) + \sum_{j \neq i} d_j(x(z_i, \phi_{-i}))\phi_j > z_i^- d_i(x(z_i^-, \phi_{-i})) + \sum_{j \neq i} d_j(x(z_i^-, \phi_{-i}))\phi_j$, contradicting the optimality.

If $|X(z_i, \phi_{-i})| > 1$, for any $x, y \in X(z_i, \phi_{-i})$ with $d_i(x) > d_i(y)$, $z_i^- d_i(x) + \sum_{j \neq i} d_j(x)\phi_j < z_i^- d_i(y) + \sum_{j \neq i} d_j(y)\phi_j$. Hence $X(z_i^-, \phi_{-i}) = \arg\min_{x \in X(z_i, \phi_{-i})} d_i(x)$. □

Lemma B.5 shows that the payoff to agent i with type z_i and its worst-off tie-breaking rule is equivalent to the payoff to agent i with type z_i^-. Thus $d_i(z_i; z, a_i^z) = d_i(z_i^-; z', a(T))$

for all i such that $z_i(T) = \tilde{z}_i(z_{-i}(T))$. In addition, $\hat{\theta}_i(z_i) = \hat{\theta}_i(z_i^-)$ since $\hat{\theta}_i(\cdot)$ is continuous. Hence, for $a = a(T)$, $z'_i = \tilde{z}_i(a)$ for all i such that $z_i(T) = \tilde{z}_i(z_{-i}(T))$.

To prove Claim 4 If there is no first iteration at the if loop, we are done. If there is a first iteration at the if loop after iteration T in the inner while loop, and the Algorithm does not terminate, then there exists some $i' \in \overline{I}(T)$ such that $i' \in \{i : z'_i > \max\{\phi_i^B(\underline{\theta}_i), \tilde{z}_i(z'_{-i})\}\}$. We have shown that for all i such that $z_i(T) = \tilde{z}_i(z_{-i}(T))$, $z'_i = \tilde{z}_i(z'_{-i})$ in the proof for **Claim 3**. Hence, for all $i' \in \{i : z'_i > \max\{\phi_i^B(\underline{\theta}_i), \tilde{z}_i(z'_{-i})\}\}$, $z'_i = \phi_{i'}^S(\overline{\theta}_{i'})$. Note that z_i is non-increasing for all i in the Algorithm. So the set $\{i : z_i(t) = \phi_i^S(\overline{\theta}_i)\}$ reduces in size over iterations. Assume that the Algorithm does not terminate before the $\{i : z_i(t) = \phi_i^S(\overline{\theta}_i)\} = \emptyset$. The proof for **Claim 3** shows that the first iteration in the if loop after $\{i : z_i(t) = \phi_i^S(\overline{\theta}_i)\} = \emptyset$ will return z such that (IR') is satisfied for some a.

B.6 Proof of Proposition 5

For any $\theta_i \in \Theta_i$,

$$\frac{T_i(\theta_i)}{d_i(\theta_i)} = \hat{\theta}_i(z_i) + \int_{\hat{\theta}_i(z_i)}^{\theta_i} 1 - \frac{d_i(x)}{d_i(\theta_i)} dx - \frac{\hat{u}_i(\hat{\theta}_i(z_i))}{d_i(\theta_i)}.$$

$1 - \frac{d_i(x)}{d_i(\theta_i)}$ increases in θ_i since $d_i(\theta_i)$ increases in θ_i by Lemma 4. Thus the integral $\int_{\hat{\theta}_i(z_i)}^{\theta_i} 1 - \frac{d_i(x)}{d_i(\theta_i)} dx$ increases in θ_i. $\frac{\hat{u}_i(\hat{\theta}_i(z_i))}{d_i(\theta_i)}$ decreases in θ_i since $d_i(\theta_i)$ increases in θ_i. So $\frac{T_i(\theta_i)}{d_i(\theta_i)}$ increases in θ_i. Then Proposition 5 follows from that $d_i(\theta_i)$ increases in θ_i.

C Supplementary material

The following lemma links Assumption 1 to the single crossing property. Towards that end, let $\mathcal{D} := \{(d_i(x))_{i \in I} : x \in \mathcal{X}\}$. For each $i \in I$, define the order $\geq_i$ on $\mathcal{D}$ as the lexicographic order such that for any $d, d' \in \mathcal{D}$, $d >_i d'$ if either $d_i > d'_i$ or $d_i = d'_i$ and there exists $k \neq i$ such that $d_j = d'_j$ for all $j < k$, and $d_k > d'_k$. $(\mathcal{D}, \geq_i)$ is a chain for all i.

Lemma C.1. *Assumption 1 holds if for all $j \in I$, $w(d, \theta_j)$ satisfies single-crossing in $(d, -\theta_j)$ on $(\mathcal{D}, \geq_i)$ for all i.*

Proof. By Theorem 4 in Milgrom and Shannon (1994). □