Chapter 2

Classic Mechanism Design

Mechanism design is the sub-field of microeconomics and game theory that considers how to implement good system-wide solutions to problems that involve multiple self-interested agents, each with private information about their preferences. In recent years mechanism design has found many important applications; e.g., in electronic market design, in distributed scheduling problems, and in combinatorial resource allocation problems.

This chapter provides an introduction to the the game-theoretic approach to mechanism design, and presents important possibility and impossibility results in the literature. There is a well-understood sense of what can and cannot be achieved, at least with fully rational agents and without computational limitations. The next chapter discusses the emerging field of *computational* mechanism design, and also surveys the economic literature on limited communication and agent bounded-rationality in mechanism design. The challenge in computational mechanism design is to design mechanisms that are *both* tractable (for agents and the auctioneer) and retain useful game-theoretic properties. For a more general introduction to the mechanism design literature, MasColell *et al.* [MCWG95] provides a good reference. Varian [Var95] provides a gentle introduction to the role of mechanism design in systems of computational agents.

In a mechanism design problem one can imagine that each agent holds one of the "inputs" to a well-formulated but incompletely specified optimization problem, perhaps a constraint or an objective function coefficient, and that the system-wide goal is to solve the specific instantiation of the optimization problem specified by the inputs. Consider for example a network routing problem in which the system-wide goal is to allocate resources to minimize the total cost of delay over all agents, but each agent has private information about parameters such as message size and its unit-cost of delay. A typical approach in mechanism design is to provide incentives (for example with suitable payments) to promote truth-revelation from agents, such that an optimal solution can be computed to the distributed optimization problem.

Groves mechanisms [Gro73] have a central role in classic mechanism design, and promise to remain very important in computational mechanism design. Indeed, Groves mechanisms have a focal role in my dissertation, providing strong guidance for the design of mechanisms in the combinatorial allocation problem. Groves mechanisms solve problems in which the goal is to select an outcome, from a set of discrete outcomes, that maximizes the *total value* over all agents. The Groves mechanisms are *strategy-proof*, which means that truth-revelation of preferences over outcomes is a dominant strategy for each agent— optimal whatever the strategies and preferences of other agents. In addition to providing a robust solution concept, strategy-proofness removes game-theoretic complexity from each individual agent's decision problem; an agent can compute its optimal strategy without needing to model the other agents in the system. In fact (see Section 2.4), Groves mechanisms are the *only* strategy-proof and value-maximizing (or efficient) mechanisms amongst an important class of mechanisms.

But Groves mechanisms have quite bad computational properties. Agents must report complete information about their preferences to the mechanism, and the optimization problem— to maximize value —is solved centrally once all this information is reported. Groves mechanisms provide a completely centralized solution to a decentralized problem. In addition to difficult issues such as privacy of information, trust, etc. the approach fails computationally in combinatorial domains either when agents cannot compute their values for all possible outcomes, or when the mechanism cannot solve the centralized problem. Computational approaches attempt to retain the useful game-theoretic properties but relax the requirement of complete information revelation. As one introduces alternative distributed implementations it is important to consider effects on game-theoretic properties, for example the effect on strategy-proofness.

Here is an outline of the chapter. Section 2.1 presents a brief introduction to game theory, introducing the most important solution concepts. Section 2.2 introduces the theory of mechanism design, and defines desirable mechanism properties such as efficiency, strategy-proofness, individual-rationality, and budget-balance. Section 2.3 describes the revelation principle, which has proved a powerful concept in mechanism design theory, and introduces incentive-compatibility and direct-revelation mechanisms. Section 2.4 presents the efficient and strategy-proof Vickrey-Clarke-Groves mechanisms, including the Generalized Vickrey Auction for the combinatorial allocation problem. Sections 2.5 and 2.6 summarize the central impossibility and possibility results in mechanism design. Finally, Section 2.7 is provides a brief discussion of optimal auction design and the conflict between the goals of revenue-maximization and efficiency.

2.1 A Brief Introduction to Game Theory

Game theory [vNM47, Nas50] is a method to study a system of self-interested agents in conditions of strategic interaction. This section provides a brief tour of important game-theoretic solution concepts. Fudenberg & Tirole [FT91] and Osborne & Rubinstein [OR94] provide useful introductions to the subject. Places to start for auction theory include McAfee & McMillan [PMM87] and Wurman *et al.* [WWW00].

2.1.1 Basic Definitions

It is useful to introduce the idea of the *type* of an agent, which determines the preferences of an agent over different outcomes of a game. This will bring clarity when we discuss mechanism design in the next section. Let $\theta_i \in \Theta_i$ denote the type of agent i, from a set of possible types Θ_i. An agent's preferences over outcomes $o \in \mathcal{O}$, for a set $\mathcal{O}$ of outcomes, can then be expressed in terms of a utility function that is parameterized on the type. Let $u_i(o, \theta_i)$ denote the utility of agent i for outcome $o \in \mathcal{O}$ given type θ_i. Agent i prefers outcome o_1 over o_2 when $u_i(o_1, \theta_i) > u_i(o_2, \theta_i)$.

The fundamental concept of agent choice in game theory is expressed as a *strategy*. Without providing unnecessary structure, a strategy can loosely be defined as:

DEFINITION 2.1 [strategy] A strategy is a complete contingent plan, or decision rule, that defines the action an agent will select in every distinguishable state of the world.

Let $s_i(\theta_i) \in \Sigma_i$ denote the strategy of agent i given type θ_i, where Σ_i is the set of all possible strategies available to an agent. Sometimes the conditioning on an agent's type is left implicit, and I write s_i for the strategy selected by agent i given its type.

In addition to *pure*, or deterministic strategies, agent strategies can also be *mixed*, or stochastic. A mixed strategy, written $\sigma_i \in \Delta(\Sigma_i)$ defines a probability distribution over pure strategies.

Example. In a single-item ascending-price auction, the state of the world (p, x) defines the current ask price $p \geq 0$ and whether or not the agent is holding the item in the provisional allocation $x \in \{0, 1\}$. A strategy defines the bid $b(p, x, v)$ that an agent will place for every state, (p, x) and for every value $v \geq 0$ it might have for the item. A best-response strategy is as follows:

$$b_{\mathrm{BR}}(p, x, v) = \begin{cases} p & \text{, if } x = 0 \text{ and } p < v \\ \text{no bid} & \text{, otherwise} \end{cases}$$

One can imagine that a *game* defines the set of actions available to an agent (e.g. valid bids, legal moves, etc.) and a mapping from agent strategies to an outcome (e.g. the agent with highest bid at the end of the auction wins the item and pays that price, checkmate to win the game, etc.)

Again, avoiding unnecessary detail, given a game (e.g. an auction, chess, etc.) we can express an agent's utility as a function of the strategies of all the agents to capture the essential concept of strategic interdependence.

Definition 2.2 [utility in a game] Let $u_i(s_1, \ldots, s_I, \theta_i)$ denote the utility of agent i at the outcome of the game, given preferences θ_i and strategies profile $s = (s_1, \ldots, s_I)$ selected by each agent.

In other words, the utility, $u_i(\cdot)$, of agent i determines its preferences over its own strategy and the strategies of other agents, given its type θ_i, which determines its base preferences over different outcomes in the world, e.g. over different allocations and payments.

Example. In a single-item ascending-price auction, if agent 2 has value $v_2 = 10$ for the item and follows strategy $b_{\mathrm{BR},2}(p, x, v_2)$ defined above, and agent 1 has value v_1 and follows strategy $b_{\mathrm{BR},1}(p, x, v_1)$, then the utility to agent 1 is:

$$u_1\left(b_{\mathrm{BR},1}(p, x, v_1), b_{\mathrm{BR},2}(p, x, 10), 10\right) = \begin{cases} v_1 - (10 + \epsilon) & \text{, if } v_1 > 10 \\ 0 & \text{, otherwise} \end{cases}$$

where $\epsilon > 0$ is the minimal bid increment in the auction and agent i's utility given value v_i and price p is $u_i = v_i - p$, i.e. equal to its surplus.

The basic model of agent rationality in game theory is that of an *expected utility maximizer*. An agent will select a strategy that maximizes its expected utility, given its preferences θ_i over outcomes, beliefs about the strategies of other agents, and structure of the game.

2.1.2 Solution Concepts

Game theory provides a number of solution concepts to compute the outcome of a game with self-interested agents, given assumptions about agent preferences, rationality, and information available to agents about each other.

The most well-known concept is that of a Nash equilibrium [Nas50], which states that in equilibrium every agent will select a utility-maximizing strategy given the strategy of every other agent. It is useful to introduce notation $s = (s_1, \ldots, s_I)$ for the joint strategies of all agents, or *strategy profile*, and $s_{-i} = (s_1, \ldots, s_{i-1}, s_{i+1}, s_I)$ for the strategy of every agent except agent i. Similarly, let θ_{-i} denote the type of every agent except i.

Definition 2.3 [Nash equilibrium] A strategy profile $s = (s_1, \ldots, s_I)$ is in Nash equilibrium if every agent maximizes its expected utility, for every i,

$$u_i(s_i(\theta_i), s_{-i}(\theta_{-i}), \theta_i) \geq u_i(s_i'(\theta_i), s_{-i}(\theta_{-i}), \theta_i), \qquad \text{for all } s_i' \neq s_i$$

In words, every agent maximizes its utility with strategy s_i, given its preferences and the strategy of every other agents. This definition can be extended in a straightforward way to include mixed strategies.

Although the Nash solution concept is fundamental to game theory, it makes very strong assumptions about agents' information and beliefs about other agents, and also loses power in games with *multiple* equilibria. To play a Nash equilibrium in a one-shot game every agent must have perfect information (and know every other agent has the same information, etc., i.e. common knowledge) about the preferences of every other agent, agent rationality must also be common knowledge, and agents must all select the same Nash equilibrium.

A stronger solution concept is a *dominant strategy* equilibrium. In a dominant strategy equilibrium every agent has the same utility-maximizing strategy, for all strategies of other agents.

Definition 2.4 [Dominant-strategy] Strategy s_i is a dominant strategy if it (weakly) maximizes the agent's expected utility for all possible strategies of other agents,

$$u_i(s_i, s_{-i}, \theta_i) \geq u_i(s_i', s_{-i}, \theta_i), \qquad \text{for all } s_i' \neq s_i,\ s_{-i} \in \Sigma_{-i}$$

In other words, a strategy s_i is a dominant strategy for an agent with preferences θ_i if it maximizes expected utility, whatever the strategies of other agents.

Example. In a sealed-bid second-price (Vickrey auction), the item is sold to the highest bidder for the second-highest price. Given value v_i, bidding strategy

$$b_i(v_i) = v_i$$

is a dominant strategy for agent i because its utility is

$$u_i(b_i, b', v_i) = \begin{cases} v_i - b' & \text{, if } b_i > b' \\ 0 & \text{otherwise} \end{cases}$$

for bid b_i and highest bid from another agent b'. By case analysis, when $b' < v_i$ then any bid $b_i \geq b'$ is optimal, and when $b' \geq v_i$ then any bid $b_i < b'$ is optimal. Bid $b_i = v_i$ solves both cases.

Dominant-strategy equilibrium is a very robust solution concept, because it makes no assumptions about the information available to agents about each other, and does not require an agent to believe that other agents will behave rationally to select its own optimal strategy. In the context of mechanism design, dominant strategy implementations of social choice functions are much more desirable than Nash implementations (which in the context of the informational assumptions at the core of mechanism design are essentially useless).

A third solution concept is *Bayesian-Nash equilibrium.* In a Bayesian-Nash equilibrium every agent is assumed to share a common *prior* about the distribution of agent types, $F(\theta)$, such that for any particular game the agent profiles are distributed according to $F(\theta)$. In equilibrium every agent selects a strategy to maximize expected utility in equilibrium with expected-utility maximizing strategies of other agents.

Definition 2.5 [Bayesian-Nash] A strategy profile $s = (s_1(\cdot), \ldots, s_I(\cdot))$ is in Bayesian-Nash equilibrium if for every agent i and all preferences $\theta_i \in \Theta_i$

$$u_i(s_i(\theta_i), s_{-i}(\cdot), \theta_i) \geq u_i(s_i'(\theta_i), s_{-i}(\cdot), \theta_i), \qquad \text{for all } s_i'(\cdot) \neq s_i(\cdot)$$

where u_i is used here to denote the *expected* utility over distribution $F(\theta)$ of types.

Comparing Bayesian-Nash with Nash equilibrium, the key difference is that agent i's strategy $s_i(\theta_i)$ must be a best-response to the *distribution* over strategies of other agents, given distributional information about the preferences of other agents. Agent i does not necessarily play a best-response to the *actual* strategies of the other agents.

Bayesian-Nash makes more reasonable assumptions about agent information than Nash, but is a weaker solution concept than dominant strategy equilibrium. Remaining problems with Bayesian-Nash include the existence of multiple equilibria, information asymmetries, and rationality assumptions, including common-knowledge of rationality.

The solution concepts, of Nash, dominant-strategy, and Bayesian-Nash, hold in both *static* and *dynamic* games. In a static game every agent commits to its strategy simultaneously (think of a sealed-bid auction for a simple example). In a dynamic game actions are interleaved with observation and agents can learn information about the preferences of other agents during the course of the game (think of an iterative auction, or stages in a negotiation). Additional refinements to these solution concepts have been proposed to solve dynamic games, for example to enforce *sequential rationality* (backwards induction) and to remove *non-credible threats* off the equilibrium path. One such refinement is sub-game perfect Nash, another is perfect Bayesian-Nash (which applies to dynamic games of incomplete information), see [FT91] for more details.

Looking ahead to mechanism design, an ideal mechanism provides agents with a dominant strategy *and* also implements a solution to the multi-agent distributed optimization problem. We can state the following preference ordering across implementation concepts: dominant $\succ$ Bayesian-Nash $\succ$ Nash. In fact, a Nash solution concept in the context of a mechanism design problem is essentially useless unless agents are very well-informed about each others' preferences, in which case it is surprising that the mechanism infrastructure itself is not also well-informed.

2.2 Mechanism Design: Important Concepts

The mechanism design problem is to implement an optimal system-wide solution to a decentralized optimization problem with self-interested agents with private information about their preferences for different outcomes.

Recall the concept of an agent's *type*, $\theta_i \in \Theta_i$, which determines its preferences over different outcomes; i.e. $u_i(o, \theta_i)$ is the utility of agent i with type θ_i for outcome $o \in \mathcal{O}$.

The system-wide goal in mechanism design is defined with a *social choice function*, which selects the optimal outcome given agent types.

DEFINITION 2.6 [Social choice function] Social choice function $f : \Theta_1 \times \ldots \times \Theta_I \rightarrow \mathcal{O}$ chooses an outcome $f(\theta) \in \mathcal{O}$, given types $\theta = (\theta_1, \ldots, \theta_I)$.

In other words, given agent types $\theta = (\theta_1, \ldots, \theta_I)$, we would like to choose outcome $f(\theta)$. The mechanism design problem is to implement "rules of a game", for example defining possible strategies and the method used to select an outcome based on agent strategies, to implement the solution to the social choice function despite agent's self-interest.

DEFINITION 2.7 [mechanism] A mechanism $\mathcal{M} = (\Sigma_1, \ldots, \Sigma_I, g(\cdot))$ defines the set of strategies Σ_i available to each agent, and an *outcome rule* $g : \Sigma_1 \times \ldots \times \Sigma_I \rightarrow \mathcal{O}$, such that $g(s)$ is the outcome implemented by the mechanism for strategy profile $s = (s_1, \ldots, s_I)$.

In words, a mechanism defines the strategies available (e.g., bid at least the ask price, etc.) and the method used to select the final outcome based on agent strategies (e.g., the price increases until only one agent bids, then the item is sold to that agent for its bid price).

Game theory is used to analyze the outcome of a mechanism. Given mechanism $\mathcal{M}$ with outcome function $g(\cdot)$, we say that a mechanism *implements* social choice function $f(\theta)$ if the outcome computed with *equilibrium* agent strategies is a solution to the social choice function for all possible agent preferences.

DEFINITION 2.8 [mechanism implementation] Mechanism $\mathcal{M} = (\Sigma_1, \ldots, \Sigma_I, g(\cdot))$ *implements* social choice function $f(\theta)$ if $g(s_1^*(\theta_1), \ldots, s_I^*(\theta_I)) = f(\theta)$, for all $(\theta_1, \ldots, \theta_I) \in \Theta_1 \times \ldots \times \Theta_I$, where strategy profile $(s_1^*, \ldots, s_I^*)$ is an equilibrium solution to the game induced by $\mathcal{M}$.

The equilibrium concept is deliberately left undefined at this stage, but may be Nash, Bayesian-Nash, dominant- or some other concept; generally as strong a solution concept as possible.

To understand why the mechanism design problem is difficult, consider a very naive mechanism, and suppose that the system-wide goal is to implement social choice function

$f(\theta)$. The mechanism asks agents to *report their types*, and then simply implements the solution to the social choice function that corresponds with their reports, i.e. the outcome rule is equivalent to the social choice function, $g(\hat{\theta}) = f(\hat{\theta})$ given reported types $\hat{\theta} = (\hat{\theta}_1, \ldots, \hat{\theta}_I)$. But, there is no reason for agents to report their true types! In a Bayesian-Nash equilibrium each agent will choose to announce a type $\hat{\theta}_i$ to maximize its expected utility, and solve:

$$\max_{\theta'_i \in \Theta_i} \; \mathrm{E}_{\theta_{-i}} u_i(\theta'_i, s_{-i}(\theta_{-i}), \theta_i)$$

given distributional information about the types of other agents, and under the assumption that the other agents are also following expected-utility maximizing strategies. This announced type $\hat{\theta}_i$ need not equal the agent's true type.

Looking ahead, the mechanism design problem is to design a mechanism— a set of possible agent strategies and an outcome rule —to implement a social choice function with desirable properties, in as strong a solution concept as possible; i.e. dominant is preferred to Bayesian-Nash because it makes less assumptions about agents.

2.2.1 Properties of Social Choice Functions

Many properties of a mechanism are stated in terms of the properties of the social choice function that the mechanism implements. A good place to start is to outline a number of desirable properties for social choice functions.

A social choice function is *Pareto optimal* (or Pareto efficient) if it implements outcomes for which no alternative outcome is strongly preferred by at least one agent, and weakly preferred by all other agents.

Definition 2.9 [Pareto optimal] Social choice function $f(\theta)$ is Pareto optimal if for every $o' \neq f(\theta)$, and all types $\theta = (\theta_1, \ldots, \theta_I)$,

$$u_i(o', \theta_i) > u_i(o, \theta_i) \quad \Rightarrow \quad \exists j \in \mathcal{I} \;\; u_j(o', \theta_j) < u_j(o, \theta_j)$$

In other words, in a Pareto optimal solution no agent can every be made happier without making at least one other agent less happy.

A very common assumption in auction theory and mechanism design, and one which I will follow in my dissertation, is that agents are *risk neutral* and have *quasi-linear utility* functions.

DEFINITION 2.10 [Quasi-linear Preferences] A quasi-linear utility function for agent i with type θ_i is of the form:

$$u_i(o, \theta_i) = v_i(x, \theta_i) - p_i$$

where outcome o defines a choice $x \in \mathcal{K}$ from a discrete choice set and a *payment* p_i by the agent.

The type of an agent with quasi-linear preferences defines its *valuation function*, $v_i(x)$, i.e. its value for each choice $x \in \mathcal{K}$. In an allocation problem the alternatives $\mathcal{K}$ represent allocations, and the transfers represent payments to the auctioneer. Quasi-linear preferences make it straightforward to transfer utility across agents, via side-payments.

Example. In an auction for a single-item, the outcome defines the allocation, i.e. which agent gets the item, and the payments of each agent. Assuming that agent i has *value* $v_i = \$10$ for the item, then its utility for an outcome in which it is allocated the item is $u_i = v_i - p = 10 - p$, and the agent has positive utility for the outcome so long as $p < \$10$.

Risk neutrality follows because an expected utility maximizing agent will pay as much as the expected value of an item. For example in a situation in which it will receive the item with value \$10 with probability π, an agent would be happy to pay as much as $\$10\pi$ for the item.

With quasi-linear agent preferences we can separate the outcome of a social choice function into a choice $x(\theta) \in \mathcal{K}$ and a payment $p_i(\theta)$ made by each agent i:

$$f(\theta) = (x(\theta), p_1(\theta), \ldots, p_I(\theta))$$

for preferences $\theta = (\theta_1, \ldots, \theta_I)$.

The *outcome rule*, $g(s)$, in a mechanism with quasi-linear agent preferences, is decomposed into a *choice rule*, $k(s)$, that selects a choice from the choice set given strategy profile s, and a payment rule $t_i(s)$ that selects a payment for agent i based on strategy profile s.

DEFINITION 2.11 [quasi-linear mechanism] A quasi-linear mechanism $\mathcal{M} = (\Sigma_1, \ldots, \Sigma_I, k(\cdot), t_1(\cdot), \ldots, t_I(\cdot))$ defines: the set of strategies Σ_i available to each agent; a *choice rule* $k : \Sigma_1 \times \ldots \times \Sigma_I \rightarrow \mathcal{K}$, such that $k(s)$ is the choice implemented for strategy profile $s = (s_1, \ldots, s_I)$; and *transfer rules* $t_i : \Sigma_1 \times \ldots \times \Sigma_I \rightarrow \mathbb{R}$, one for each agent i, to compute the payment $t_i(s)$ made by agent i.

Properties of social choice functions implemented by a mechanism can now be stated *separately*, for both the choice selected and the payments.

A social choice function is *efficient* if:

DEFINITION 2.12 [allocative efficiency] Social choice function $f(\theta) = (x(\theta), p(\theta))$ is *allocatively-efficient* if for all preferences $\theta = (\theta_1, \ldots, \theta_I)$

$$\sum_{i=1}^{I} v_i(x(\theta), \theta_i) \geq \sum_i v_i(x', \theta_i), \qquad \text{for all } x' \in \mathcal{K} \tag{Eff}$$

It is common to state this as *allocative* efficiency, because the choice sets often define an allocation of items to agents. An efficient allocation maximizes the total value over all agents.

A social choice function is *budget-balanced* if:

DEFINITION 2.13 [budget-balance] Social choice function $f(\theta) = (x(\theta), p(\theta))$ is *budget-balanced* if for all preferences $\theta = (\theta_1, \ldots, \theta_I)$

$$\sum_{i=1}^{I} p_i(\theta) = 0 \tag{BB}$$

In other words, there are no net transfers out of the system or into the system. Taken together, allocative efficiency and budget-balance imply Pareto optimality.

A social-choice function is *weak* budget-balanced if:

DEFINITION 2.14 [weak budget-balance] Social choice function $f(\theta) = (x(\theta), p(\theta))$ is *weakly budget-balanced* if for all preferences $\theta = (\theta_1, \ldots, \theta_I)$

$$\sum_{i=1}^{I} p_i(\theta) \geq 0 \tag{WBB}$$

In other words, there can be a net payment made from agents to the mechanism, but no net payment from the mechanism to the agents.

2.2.2 Properties of Mechanisms

Finally, we can define desirable properties of mechanisms. In describing the properties of a mechanism one must state: the *solution concept*, e.g. Bayesian-Nash, dominant, etc.; and the *domain of agent preferences*, e.g. quasi-linear, monotonic, etc.

The definitions follow quite naturally from the concept of implementation (see definition 2.8) and properties of social choice functions. A mechanism has property P if it implements a social choice function with property P.

For example, consider the definition of a *Pareto optimal* mechanism:

DEFINITION 2.15 [Pareto optimal mechanism] Mechanism $\mathcal{M}$ is Pareto optimal if it *implements* a Pareto optimal social choice function $f(\theta)$.

Technically, this is *ex post* Pareto optimality; i.e. the outcome is Pareto optimal for the specific agent types. A weaker form of Pareto optimality is *ex ante*, in which there is no outcome that at least one agent strictly prefers and all other agents weakly prefer *in expectation*.

Similarly, a mechanism is efficient if it selects the choice $x(\theta) \in \mathcal{K}$ that maximizes total value:

DEFINITION 2.16 [efficient mechanism] Mechanism $\mathcal{M}$ is efficient if it *implements* an allocatively-efficient social choice function $f(\theta)$.

Corresponding definitions follow for budget-balance and weak budget-balance. In the case of budget-balance it is important to make a careful distinction between *ex ante* and *ex post* budget balance.

DEFINITION 2.17 [*ex ante* BB] Mechanism $\mathcal{M}$ is *ex ante* budget-balanced if the equilibrium net transfers to the mechanism are balanced *in expectation* for a distribution over agent preferences.

DEFINITION 2.18 [*ex post* BB] Mechanism $\mathcal{M}$ is *ex post* budget-balanced if the equilibrium net transfers to the mechanism are non-negative *for all* agent preferences, i.e. every time.

Another important property of a mechanism is *individual-rationality*, sometimes known as "voluntary participation" constraints, which allows for the idea that an agent is often not forced to participate in a mechanism but can decide whether or not to participate. Essentially, individual-rationality places constraints on the *level* of expected utility that an agent receives from participation.

Let $\overline{u}_i(\theta_i)$ denote the expected utility achieved by agent i outside of the mechanism, when its type is θ_i. The most natural definition of individual-rationality (IR) is *interim* IR, which states that the expected utility to an agent that *knows its own preferences but* *has only distributional information about the preferences of the other agents* is at least its expected outside utility.

DEFINITION 2.19 [individual rationality] A mechanism $\mathcal{M}$ is (*interim*) individual-rational if for all preferences θ_i it implements a social choice function $f(\theta)$ with

$$u_i(f(\theta_i, \theta_{-i})) \geq \overline{u}_i(\theta_i) \tag{IR}$$

where $u_i(f(\theta_i, \theta_{-i}))$ is the *expected utility* for agent i at the outcome, given distributional information about the preferences θ_{-i} of other agents, and $\overline{u}_i(\theta_i)$ is the expected utility for non-participation.

In other words, a mechanism is individual-rational if an agent can always achieve as much expected utility from participation as without participation, given prior beliefs about the preferences of other agents.

In a mechanism in which an agent can withdraw once it learns the outcome *ex post* IR is more appropriate, in which the agent's expected utility from participation must be at least its best outside utility *for all* possible types of agents in the system. In a mechanism in which an agent must choose to participate before it even knows its own preferences then *ex ante* IR is appropriate; *ex ante* IR states that the agent's expected utility in the mechanism, averaged over all possible preferences, must be at least its expected utility without participating, also averaged over all possible preferences.

One last important mechanism property, defined for *direct-revelation* mechanisms, is *incentive-compatibility.* The concept of incentive compatibility and direct-revelation mechanisms is very important in mechanism design, and discussed in the next section in the context of the *revelation principle.*

2.3 The Revelation Principle, Incentive-Compatibility, and Direct-Revelation

The *revelation principle* states that under quite weak conditions any mechanism can be transformed into an equivalent *incentive-compatible direct-revelation mechanism*, such that it implements the same social-choice function. This proves to be a powerful theoretic tool, leading to the central possibility and impossibility results of mechanism design.

A direct-revelation mechanism is a mechanism in which the only actions available to agents are to make direct claims about their preferences to the mechanism. An incentive-compatible mechanism is a direct-revelation mechanism in which agents report *truthful information* about their preferences in equilibrium. Incentive-compatibility captures the essence of designing a mechanism to overcome the self-interest of agents— in an incentive-compatible mechanism an agent will choose to report its private information truthfully, out of its own self-interest.

Example. The second-price sealed-bid (Vickrey) auction is an incentive-compatible (actually strategy-proof) direct-revelation mechanism for the single-item allocation problem.

Computationally, the revelation principle must be viewed with great suspicion. Direct-revelation mechanisms are often too expensive for agents because they place very high demands on information revelation. An iterative mechanism can sometimes implement the same outcome as a direct-revelation mechanism but with less information revelation and agent computation. The revelation principle restricts *what* we can do, but does not explain *how* to construct a mechanism to achieve a particular set of properties. This is discussed further in Chapter 3.

2.3.1 Incentive Compatibility and Strategy-Proofness

In a direct-revelation mechanism the only action available to an agent is to submit a claim about its preferences.

Definition 2.20 [direct-revelation mechanism] A direct-revelation mechanism $\mathcal{M} = (\Theta_1, \ldots, \Theta_I, g(\cdot))$ restricts the strategy set $\Sigma_i = \Theta_i$ for all i, and has outcome rule $g : \Theta_1 \times \ldots \times \Theta_I \rightarrow \mathcal{O}$ which selects an outcome $g(\hat{\theta})$ based on reported preferences $\hat{\theta} = (\hat{\theta}_1, \ldots, \hat{\theta}_I)$.

In other words, in a direct-revelation mechanism the strategy of agent i is to report type $\hat{\theta}_i = s_i(\theta_i)$, based on its actual preferences θ_i.

A *truth-revealing* strategy is to report true information about preferences, for all possible preferences:

Definition 2.21 [truth-revelation] A strategy $s_i \in \Sigma_i$ is truth-revealing if $s_i(\theta_i) = \theta_i$ for all $\theta_i \in \Theta_i$.

In an *incentive-compatible* (IC) mechanism the equilibrium strategy profile $s^* = (s_1^*,$

$\ldots, s_I^*)$ has every agent reporting its true preferences to the mechanism. We first define Bayesian-Nash incentive-compatibility:

DEFINITION 2.22 [Bayesian-Nash incentive compatible] A direct-revelation mechanism $\mathcal{M}$ is Bayesian-Nash incentive-compatible if truth-revelation is a Bayesian-Nash equilibrium of the game induced by the mechanism.

In other words, in an incentive-compatible mechanism every agent's expected utility maximizing strategy in equilibrium with every other agent is to report its true preferences.

A mechanism is *strategy-proof* (or dominant-strategy incentive-compatible) if truth-revelation is a *dominant-strategy* equilibrium:

DEFINITION 2.23 [strategy-proof] A direct-revelation mechanism $\mathcal{M}$ is strategy-proof if it truth-revelation is a dominant-strategy equilibrium.

Strategy-proofness is a very useful property, both game-theoretically and computationally. Dominant-strategy implementation is very robust to assumptions about agents, such as the information and rationality of agents. Computationally, an agent can compute its optimal strategy without modeling the preferences and strategies of other agents.

A simple equivalence exists between the outcome function $g(\hat{\theta})$ in a direct-revelation mechanism, which selects an outcome based on reported types $\hat{\theta}$ and the social choice function $f(\theta)$ *implemented* by the mechanism, i.e. computed in equilibrium.

PROPOSITION 2.1 (incentive-compatible implementation). *An incentive-compatible direct-revelation mechanism* $\mathcal{M}$ implements *social choice function* $f(\theta) = g(\theta)$, *where* $g(\theta)$ *is the* outcome rule *of the mechanism.*

In other words, in an incentive-compatible mechanism the outcome rule is precisely the social choice function implemented by the mechanism. In Section 2.4 we introduce the *Groves* mechanisms, which are strategy-proof efficient mechanisms for agents with quasi-linear preferences, i.e. the choice rule $k(\hat{\theta})$ computes the efficient allocation given reported types $\hat{\theta}$ and an agent's dominant strategy is truth-revelation.

2.3.2 The Revelation Principle

The *revelation principle* states that under quite weak conditions any mechanism can be transformed into an equivalent *incentive-compatible direct-revelation mechanism* that implements the same social-choice function. The revelation principle is an important tool for the theoretical analysis of what is possible, and of what is impossible, in mechanism design. The revelation principle was first formulated for dominant-strategy equilibria [Gib73], and later extended by Green & Laffont [GJJ77] and Myerson [Mye79, Mye81].

One interpretation of the revelation principle is that incentive-compatibility comes for free. This is not to say that truth-revelation is easy to achieve, but simply that if an indirect-revelation and/or non-truthful mechanism solves a distributed optimization problem, then we would also expect a direct-revelation truthful implementation.

The revelation principle for dominant strategy implementation states that any social choice function than is implementable in dominant strategy is also implementable in a strategy-proof mechanism. In other words it is possible to restrict attention to truth-revealing direct-revelation mechanisms.

THEOREM 2.1 (Revelation Principle). *Suppose there exists a mechanism (direct or otherwise) $\mathcal{M}$ that implements the social-choice function $f(\cdot)$ in dominant strategies. Then $f(\cdot)$ is* truthfully implementable *in dominant strategies, i.e. in a strategy-proof mechanism.*

PROOF. If $\mathcal{M} = (\Sigma_1, \ldots, \Sigma_I, g(\cdot))$ implements $f(\cdot)$ in dominant strategies, then there exists a profile of strategies $s^*(\cdot) = (s_1^*(\cdot), \ldots, s_I^*(\cdot))$ such that $g(s^*(\theta)) = f(\theta)$ for all θ, and for all i and all $\theta_i \in \Theta_i$,

$$u_i(g(s_i^*(\theta_i), s_{-i}), \theta_i) \geq u_i(g(\hat{s}_i, s_{-i}), \theta_i)$$

for all $\hat{s}_i \in \Sigma_i$ and all $s_{-i} \in \Sigma_{-i}$, by definition of dominant strategy implementation. Substituting $s_{-i}^*(\theta_{-i})$ for s_{-i} and $s_i^*(\hat{\theta}_i)$ for $\hat{s}_i$, we have:

$$u_i(g(s_i^*(\theta_i), s_{-i}^*(\theta_{-i})), \theta_i) \geq u_i(g(s_i^*(\hat{\theta}_i, s_{-i}^*(\theta_{-i})), \theta_i)$$

for all $\hat{\theta}_i \in \Theta_i$ and all $\theta_{-i} \in \Theta_{-i}$. Finally, since $g(s^*(\theta)) = f(\theta)$ for all θ, we have:

$$u_i(f(\theta_i, \theta_{-i}), \theta_i) \geq u_i(f(\hat{\theta}_i, \theta_{-i}), \theta_i)$$

for all $\hat{\theta}_i \in \Theta_i$ and all $\theta_{-i} \in \Theta_{-i}$. This is precisely the condition required for $f(\cdot)$ to be truthfully implementable in dominant strategies in a direct-revelation mechanism. The outcome rule in the strategy-proof mechanism, $g : \theta_1 \times \ldots \times \theta_I \rightarrow \mathcal{O}$, is simply equal to the social choice function $f(\cdot)$. ▮

The intuition behind the revelation principle is as follows. Suppose that it is possible to *simulate* the entire system— the bidding strategies of agents and the outcome rule — of an indirect mechanism, given complete and perfect information about the preferences of every agent. Now, if it is possible to claim credibly that the "simulator" will implement an agent's optimal strategy faithfully, given information about the preferences (or type) of the agent, then it is optimal for an agent to truthfully report its preferences to the new mechanism.

This dominant-strategy revelation principle is quite striking. In particular, it suggests that to identify which social choice functions are implementable in dominant strategies, we need only identify those functions $f(\cdot)$ for which truth-revelation is a dominant strategy for agents in a direct-revelation mechanism with outcome rule $g(\cdot) = f(\cdot)$.

A similar revelation principle can be stated in Bayesian-Nash equilibrium.

THEOREM 2.2 (Bayesian-Nash Revelation Principle). *Suppose there exists a mechanism (direct or otherwise) $\mathcal{M}$ that implements the social-choice function $f(\cdot)$ in Bayesian-Nash equilibrium. Then $f(\cdot)$ is truthfully implementable in a (Bayesian-Nash) incentive-compatible direct-revelation mechanism.*

In other words, if the goal is to implement a particular social choice function in Bayesian-Nash equilibrium, it is sufficient to consider only incentive-compatible direct-revelation mechanisms.

The proof closely follows that of the dominant-strategy revelation principle. One problem with the revelation principle for Bayesian-Nash implementation is that the distribution over agent types must be common knowledge to the direct-revelation mechanism, in addition to the agents.

2.3.3 Implications

With the revelation principle in hand we can prove *impossibility results* over the space of direct-revelation mechanisms, and construct *possibility results* over the space of direct-revelation mechanisms.

However, the revelation principle ignores computational considerations and should *not* be taken as a statement that it is sufficient to consider only direct-revelation mechanisms in practical mechanism design. The revelation principle states what can be achieved, what cannot be achieved, but without stating the *computational structure* to achieve a particular set of properties. In particular, in my dissertation I argue that iterative and indirect mechanisms are important in many combinatorial applications, and can provide tractable solutions to problems in which single-shot direct-revelation mechanisms fail.

Rather, the revelation principle provides a rich structure to the mechanism design problem, focusing goals and delineating what *is* and *is not* possible. For example, if a particular direct-revelation mechanism $\mathcal{M}$ is the only mechanism with a particular combination of properties, *then any mechanism, including iterative and indirect mechanisms*, must implement the same outcome (e.g. allocation and payments) as mechanism $\mathcal{M}$ for the same agent preferences.

For example:

- Suppose that the only direct mechanisms with useful properties P1, P2 and P3 are in the class of mechanisms $\mathcal{M}'$. It follows that any mechanism m with properties P1, P2 and P3 must be "outcome equivalent" to a direct mechanism in $\mathcal{M}'$, in the sense that m must implement the same outcome as a mechanism in this class for all possible agent types.

- Suppose that *no* direct mechanism has properties P1, P2 and P3. It follows that there can be no mechanism (direct or otherwise) with properties P1, P2 and P3.

The next section introduces an important family of mechanisms with dominant-strategy solutions.

2.4 Vickrey-Clarke-Groves Mechanisms

In seminal papers, Vickrey [Vic61], Clarke [Cla71] and Groves [Gro73], proposed the Vickrey-Clarke-Groves family of mechanisms, often simply called the Groves mechanisms, for problems in which agents have quasi-linear preferences. The Groves mechanisms are allocatively-efficient and strategy-proof direct-revelation mechanisms.

In special cases there is a Groves mechanism that is also individual-rational and satisfies *weak budget-balance*, such that the mechanism does not require an outside subsidy to operate. This is the case, for example, in the Vickrey-Clarke-Groves mechanism for a combinatorial auction.

In fact, the Groves family of mechanisms characterize the *only* mechanisms that are allocatively-efficient and strategy-proof [GJJ77] amongst direct-revelation mechanisms.

THEOREM 2.3 (Groves Uniqueness). *The Groves mechanisms are the only allocatively-efficient and strategy-proof mechanisms for agents with quasi-linear preferences and general valuation functions, amongst all direct-revelation mechanisms.*

The revelation principle extends this uniqueness to general mechanisms, direct or otherwise. Given the premise that iterative mechanisms often have preferable computational properties in comparison to sealed-bid mechanisms, this uniquenss suggests a focus on *iterative Groves mechanisms* because:

any iterative mechanism that achieves allocative efficiency in dominant-strategy implementation must implement a Groves outcome.

In fact, we will see in Chapter 7 that an iterative mechanism that implements the Vickrey outcome can have slightly weaker properties than those of a single-shot Vickrey scheme.

Krishna & Perry [KP98] and Williams [Wil99] have recently proved the uniqueness of Groves mechanisms among efficient and Bayesian-Nash mechanisms.

2.4.1 The Groves Mechanism

Consider a set of possible alternatives, $\mathcal{K}$, and agents with quasi-linear utility functions, such that

$$u_i(k, p_i, \theta_i) = v_i(k, \theta_i) - p_i$$

where $v_i(k, \theta_i)$ is the agent's *value* for alternative k, and p_i is a payment by the agent to the mechanism. Recall that the *type* $\theta_i \in \Theta_i$ is a convenient way to express the valuation function of an agent.

In a direct-revelation mechanism for quasi-linear preferences we write the outcome rule $g(\hat{\theta})$ in terms of a *choice rule*, $k : \Theta_1 \times \ldots \times \Theta_I \to \mathcal{K}$, and a *payment rule*, $t_i : \Theta_1 \times \ldots \times \Theta_I \to \mathbb{R}$, for each agent i.

In a Groves mechanism agent i reports type $\hat{\theta}_i = s_i(\theta_i)$, which may not be its true type. Given reported types $\hat{\theta} = (\hat{\theta}_1, \ldots, \hat{\theta}_I)$, the choice rule in a Groves mechanism computes:

$$k^*(\hat{\theta}) = \arg\max_{k \in \mathcal{K}} \sum_i v_i(k, \hat{\theta}_i) \tag{1}$$

Choice k^* is the selection that maximizes the total reported value over all agents.

The payment rule in a Groves mechanism is defined as:

$$t_i(\hat{\theta}) = h_i(\hat{\theta}_{-i}) - \sum_{j \neq i} v_j(k^*, \hat{\theta}_j) \tag{2.1}$$

where $h_i : \Theta_{-i} \to \mathbb{R}$ is an arbitrary function on the reported types of every agent except i. This freedom in selecting $h_i(\cdot)$ leads to the description of a "family" of mechanisms. Different choices make different tradeoffs across budget-balance and individual-rationality.

2.4.2 Analysis

Groves mechanisms are efficient and strategy-proof:

THEOREM 2.4 (Groves mechanisms). *Groves mechanisms are allocatively-efficient and strategy-proof for agents with quasi-linear preferences.*

PROOF.

We prove that Groves mechanisms are strategy-proof, such that truth-revelation is a dominant strategy for each agent, from which allocative efficiency follows immediately because the choice rule $k^*(\theta)$ computes the efficient allocation (1).

The utility to agent i from strategy $\hat{\theta}_i$ is:

$$\begin{aligned} u_i(\hat{\theta}_i) &= v_i(k^*(\hat{\theta}), \theta_i) - t_i(\hat{\theta}) \\ &= v_i(k^*(\hat{\theta}), \theta_i) + \sum_{j \neq i} v_j(k^*(\hat{\theta}), \hat{\theta}_j) - h_i(\hat{\theta}_{-i}) \end{aligned}$$

Ignoring the final term, because $h_i(\hat{\theta}_{-i})$ is independent of an agent i's reported type, we prove that truth-revelation $\hat{\theta}_i = \theta_i$ solves:

$$
\begin{aligned}
& \max_{\hat{\theta}_i \in \Theta_i} \left[v_i(k^*(\hat{\theta}_i, \hat{\theta}_{-i}), \theta_i) + \sum_{j \neq i} v_j(k^*(\hat{\theta}_i, \hat{\theta}_{-i}), \hat{\theta}_j) \right] \\
= & \max_{\hat{\theta}_i \in \Theta_i} \left[v_i(x, \theta_i) + \sum_{j \neq i} v_j(x, \hat{\theta}_j) \right] \qquad (2)
\end{aligned}
$$

where $x = k^*(\hat{\theta}_i, \hat{\theta}_{-i})$ is the outcome selected by the mechanism. The only effect of the agent's announced type $\hat{\theta}_i$ is on x, and the agent can maximize (2) by announcing $\hat{\theta}_i = \theta_i$ because then the mechanism computes $k^*(\hat{\theta}_i, \hat{\theta}_{-i})$ to explicitly solve:

$$
\max_{k \in \mathcal{K}} v_i(k, \theta_i) + \sum_{j \neq i} v_j(k, \hat{\theta}_j)
$$

Truth-revelation is the *dominant strategy* of agent i, whatever the reported types $\hat{\theta}_{-i}$ of the other agents. ▮

The effect of payment $t_i(\hat{\theta}) = (\cdot) - \sum_{j \neq i} v_j(k^*, \hat{\theta}_j)$ is to "internalize the externality" placed on the other agents in the system by the reported preferences of agent i. This aligns the agents' incentives with the system-wide goal of an efficient allocation, an agent *wants* the mechanism to select the best system-wide solution given the reports of other agents and its own *true* preferences.

The first term in the payment rule, $h_i(\hat{\theta}_{-i})$, can be used to achieve (weak) budget-balance and/or individual rationality. It is not possible to simply total up the payments made to each agent in the Groves scheme and divide equally across agents, because the total payments depend on the outcome, and therefore the reported type of each agent. This would break the strategy-proofness of the mechanism.

2.4.3 The Vickrey Auction

The special case of Clarke mechanism for the allocation of a single item is the familiar second-price sealed-bid auction, or Vickrey [Vic61] auction.

In this case, with bids b_1 and b_2 to indicate the first- and second- highest bids, the item is sold to the item with the highest bid (agent 1), for a price computed as:

$$
b_1 - (b_1 - b_2) = b_2
$$

i.e. the *second-highest* bid.

One can get some intuition for the strategy-proofness of the Groves mechanisms in this special case. Truth-revelation is a dominant strategy in the Vickrey auction because an agent's bid determines the *range* of prices that it will accept, but not the actual price it pays. The price that an agent pays is completely independent of its bid price, and even if an agent knows the second-highest bid it can still bid its true value because it only pays just enough to out-bid the other agent. In addition, notice that *weak* budget-balance holds, because the second-highest bid price is non-negative, and *individual-rationality* holds because the second-highest bid price is no greater than the highest bid price, which is equal to the winner agent's value in equilibrium.

2.4.4 The Pivotal Mechanism

The Pivotal, or Clarke, mechanism [Cla71] is a Groves mechanism in which the payment rule, $h_i(\hat{\theta}_{-i})$, is carefully set to achieve individual-rationality, while also maximizing the payments made by the agents to the mechanism. The Pivotal mechanism also achieves *weak* budget-balance whenever that is possible in an efficient and strategy-proof mechanism [KP98].

The Clarke mechanism [Cla71] computes the additional transfer term as:

$$h_i(\hat{\theta}_{-i}) = \sum_{j \neq i} v_j(k^*_{-i}(\hat{\theta}_{-i}), \hat{\theta}_j) \tag{2.2}$$

where $k^*_{-i}(\hat{\theta}_{-i})$ is the *optimal collective choice* for with agent i taken out of the system:

$$k^*_{-i}(\hat{\theta}_{-i}) = \arg\max_{k \in \mathcal{K}} \sum_{j \neq i} v_j(k, \hat{\theta}_j)$$

This is a valid additional transfer term because the reported value of the second-best allocation without agent i is *independent* of the report from agent i. The strategy-proofness and efficiency of the Groves mechanisms are left unchanged.

The Clarke mechanism is a useful special-case because it is also *individual rational* in quite general settings, which means that agents will choose to participate in the mechanism (see Section 2.2.2).

To keep things simple, let us assume that agent i's expected utility from not participating in the mechanism is $\overline{u}_i(\theta_i) = 0$. The Clarke mechanism is individual rational when the following two (sufficient) conditions hold on agent preferences:

DEFINITION 2.24 [choice set monotonicity] The feasible choice set available to the mechanism $\mathcal{K}$ (weakly) increases as additional agents are introduced into the system.

DEFINITION 2.25 [no negative externalities] Agent i has non-negative value, i.e. $v_i(k^*_{-i}, \theta_i) \geq 0$, for any optimal solution choice, $k^*_{-i}(\theta_{-i})$ without agent i, for all i and all θ_i.

In other words, with *choice set monotonicity* an agent cannot "block" a selection, and with *no negative externalities*, then any choice not involving an agent has a neutral (or positive) effect on that agent.

For example, the conditions of choice-set monotonicity and no negative externalities hold in the following settings:

- In a *private goods* market environment: introducing a new agent cannot make existing trades infeasible (in fact it can only increase the range of possible trades); and with only private goods no agent has a negative value for the trades executed between other agents (relative to no trades).

- In a *public project* choice problem: introducing a new agent cannot change the range of public projects that can be implemented; and no agent has negative value for any public project (relative to the project not going ahead).

PROPOSITION 2.2 (Clarke mechanism). *The Pivotal (or Clarke) mechanism is (*ex post*) individual-rational, efficient, and strategy-proof when choice-set monotonicity and no negative externalities hold and with quasi-linear agent preferences.*

PROOF. To show individual-rationality (actually *ex post* individual-rationality), we show that the utility to agent i in the equilibrium outcome of the mechanism is always non-negative. We can assume truth-revelation in equilibrium. The utility to agent i with type θ_i is:

$$\begin{aligned} u_i(\theta_i, \theta_{-i}) &= v_i(k^*(\theta), \theta_i) - \left(\sum_{j \neq i} v_j(k^*_{-i}(\theta_{-i}), \theta_j) - \sum_{j \neq i} v_j(k^*(\theta), \theta_j) \right) \\ &= \sum_i v_i(k^*(\theta), \theta_i) - \sum_{j \neq i} v_j(k^*_{-i}(\theta_{-i}), \theta_j) \qquad (3) \end{aligned}$$

Expression (3) is non-negative because the value of the best solution without agent i, $\sum_{j\neq i} v_j(k^*_{-i}(\theta_{-i}), \theta_j)$, cannot be greater than the value of the best solution with agent i, $\sum_i v_i(k^*(\theta), \theta_i)$. This follows because any choice with agents $j \neq i$ is also feasible with all agents (*monotonicity*), and has just as much total value (*no negative externalities*). ∎

The Clarke mechanism also achieves *weak* budget-balance in special-cases. A sufficient condition is the *no single-agent effect*:

Definition 2.26 [no single-agent effect] For any collective choice k' that is optimal in some scenario with all agents, i.e. $k' = \max_{k\in\mathcal{K}} \sum_i v_i(k, \theta_i)$, for some $\theta \in \Theta$, then for all i there must exist another choice k_{-i} that is feasible without i and has as much value to the remaining agents $j \neq i$.

In words, the *no single-agent effect* condition states that any one agent can be removed from an optimal system-wide solution without having a negative effect on the best choice available to the remaining agents. This condition holds in the following settings:

- In an auction with only buyers (i.e. the auctioneer holds all the items for sale), so long as all buyers have "free disposal", such that they have at least as much value for more items than less items.

- In a public project choice, because the set of choices available is static, however many agents are in the system.

Proposition 2.3 (Clarke weak budget-balance). *The Pivotal (or Clarke) mechanism is (ex post) individual-rational, weak budget-balanced, efficient and strategy-proof when choice-set monotonicity, no negative externalities, and no single-agent effect hold, and with quasi-linear agent preferences.*

Proof. Again, we can assume truth-revelation in equilibrium, and prove that the total transfers are non-negative, such that the mechanism does not require a subsidy, i.e.

$$\sum_i t_i(\theta) \geq 0$$

for all $\theta \in \Theta$. Substituting the expression for agent transfers, we have:

$$\sum_i \left(\sum_{j\neq i} v_j(k^*_{-i}(\theta_{-i}), \theta_j) - \sum_{j\neq i} v_j(k^*(\theta), \theta_j) \right) \geq 0$$

This is satisfied in Clarke because the transfer is non-negative for *every* agent i, i.e.:

$$\sum_{j \neq i} v_j(k^*_{-i}(\theta_{-i}), \theta_j) \geq \sum_{j \neq i} v_j(k^*(\theta), \theta_j), \qquad \forall i$$

This condition holds by a simple feasibility argument with the no single-agent effect, because any solution to the system with all agents remains feasible and has positive value without any one agent. ▮

As soon as there are buyers and sellers in a market we very quickly lose even weak budget-balance with Groves-Clarke mechanisms. The budget-balance problem in a combinatorial exchange is addressed in Parkes, Kalagnanam & Eso [PKE01], where we propose a number of methods to trade-off strategy-proofness and allocative efficiency for budget-balance.

2.4.5 The Generalized Vickrey Auction

The Generalized Vickrey Auction is an application of the Pivotal mechanism to the combinatorial allocation problem. The combinatorial allocation problem (CAP) was introduced in Section 1.2. There are a set $\mathcal{G}$ of items to allocate to $\mathcal{I}$ agents. The set of choices $\mathcal{K} = \{(S_1, \ldots, S_I) \; : \; S_i \cap S_j = \emptyset, S_i \subseteq \mathcal{G}\}$ where S_i is an allocation of a bundle of items to agent i. Given preferences (or type) θ_i, each agent i has a quasi-linear utility function, $u_i(S, p_i, \theta_i) = v_i(S, \theta_i) - p_i$, for bundle S and payment p_i. For notational simplicity we will drop the "type" notation in this section, and simply write $v_i(S, \theta_i) = v_i(S)$.

The efficient allocation computes an allocation to maximize the total value:

$$\begin{aligned} S^* = \arg \max_{S=(S_1,\ldots,S_I)} \sum_i v_i(S_i) \\ \text{s.t.} \quad S_i \cap S_j = \emptyset, \qquad \text{for all } i, j \end{aligned}$$

The Pivotal mechanism applied to this problem is a sealed-bid combinatorial auction, often called the *Generalized Vickrey Auction* (GVA). The special case for a single item is the Vickrey auction. In the GVA each agent bids a value for all possible sets of items, and the mechanism computes an allocation and payments.

The GVA has the following useful properties:

Theorem 2.5 (Generalized Vickrey Auction). *The GVA is efficient, strategy-proof, individual-rational, and weak budget-balanced for agents with quasi-linear preferences in the combinatorial allocation problem.*

Description

Each agent $i \in \mathcal{I}$ submits a (possibly untruthful) valuation function, $\hat{v}_i(S)$, to the auctioneer. The outcome rule in the Pivotal mechanism computes $k^*(\hat{\theta})$, the allocation that maximizes reported value over all agents. In the GVA this is equivalent to the auctioneer solving a "winner-determination" problem, solving CAP with the reported values and computing allocation $\mathbf{S}^* = (S_1^*, \ldots, S_I^*)$ to maximize reported value. Let V^* denote the total value of this allocation. Allocation $\mathbf{S}^*$ is the allocation implemented by the auctioneer.

The payment rule in the Pivotal mechanism also requires that the auctioneer solves a smaller CAP, with each agent i taken out in turn, to compute $k^*_{-i}(\theta_{-i})$, the best allocation without agent i. Let $(S_{-i})^*$ denote this second-best allocation, and $(V_{-i})^*$ denote its value.

Finally, from the Groves-Clarke payment rule $t_i(\hat{\theta})$, see (2.1) and (2.2), the auctioneer computes agent i's payment as:

$$p_{\text{vick}}(i) = (V_{-i})^* - \sum_{j \neq i} \hat{v}_j(S_j^*)$$

In words, an agent pays the marginal negative effect that its participation has on the (reported) value of the other agents. Equivalently, the Vickrey payment can be formulated as a discount $\Delta_{\text{vick}}(i)$ from its bid price, $\hat{v}_i(S_i^*)$, i.e. $p_{\text{vick}}(i) = \hat{v}_i(S_i^*) - \Delta_{\text{vick}}(i)$, for *Vickrey discount*:

$$\Delta_{\text{vick}}(i) = V^* - (V_{-i})^*$$

Analysis

Efficiency and strategy-proofness follow immediately from the properties of the Groves mechanism. Weak budget-balance also holds; it is simple to show that each agent pays a non-negative amount to the auctioneer by a simple feasibility argument. Individual-rationality also holds, and agents pay no more than their value for the bundle they receive;

Name	Preferences	Solution concept	Impossible	Environment
GibSat	general	dominant	Non-dictatorial (incl. Pareto Optimal)	general
Hurwicz	quasi-linear	dominant	Eff& BB	simple-exchange
MyerSat	quasi-linear	Bayesian-Nash	Eff& BB & IR	simple-exchange
GrLaff	quasi-linear	coalition-proof	Eff	simple-exchange

Table 2.1: Mechanism design: Impossibility results. *Eff* is *ex post* allocative efficiency, *BB* is *ex post* (and strong) budget-balance, and *IR* is *interim* individual rationality.

it is simple to show that discounts are always non-negative, again by a simple feasibility argument. Alternatively, one can verify that conditions choice-set monotonicity, no negative externalities, and no single-agent effect hold for the CAP.

2.5 Impossibility Results

The revelation principle allows the derivation of a number of impossibility theorems that outline the combinations of properties that *no* mechanism can achieve (with fully rational agents) in particular types of environments. The basic approach to show impossibility is to assume direct-revelation and incentive-compatibility, express the desired properties of an outcome rule as a set of mathematical conditions (including conditions for incentive-compatibility), and then show a conflict across the conditions.

Table 2.1 describes the main impossibility results. Results are delineated by *conditions on agent preferences*, the *equilibrium solution concept*, and the assumptions about the *environment*. The "Impossible" column lists the combinations of desirable mechanism properties that cannot be achieved in each case.

As discussed in Section 2.2.2, *ex post* refers to conditions tested at the outcome of the mechanism. *Interim* individual-rationality means that an agent that knows its own preferences but only has distributional information about the preferences of other agents will choose to participate in the mechanism.

A few words about the interpretation of impossibility results are probably useful. Impossibility for restricted preferences in an exchange is more severe than for general preferences and general environments, because general conditions include these as special cases.

In addition, impossibility for weak solution concepts such as Bayesian-Nash is more restrictive than impossibility for strong solution concepts like dominant strategy implementation.

We also need a few more definitions:

Definition 2.27 [dictatorial] A social-choice function is *dictatorial* if one (or more) agents always receives one of its most-preferred alternatives.

Definition 2.28 [general preferences] Preferences θ_i are *general* when they provide a complete and transitive preference ordering $\succ$ on outcomes. An ordering is *complete* if for all $o_1, o_2 \in \mathcal{O}$, we have $o_1 \succ o_2$ or $o_2 \succ o_1$ (or both). An ordering is *transitive* if for all $o_1, o_2, o_3 \in \mathcal{O}$, if $o_1 \succ o_2$ and $o_2 \succ o_3$ then $o_1 \succ o_3$.

Definition 2.29 [coalition-proof] A mechanism $\mathcal{M}$ is *coalition-proof* if truth revelation is a dominant strategy for any coalition of agents, where a coalition is able to make side-payments and re-distribute items after the mechanism terminates.

Definition 2.30 [general environment] A *general* environment is one in which there is a discrete set of possible outcomes $\mathcal{O}$ and agents have general preferences.

Definition 2.31 [simple exchange] A *simple exchange* environment is one in which there are buyers and sellers, selling single units of the same good.

The Gibbard [Gib73] and Satterthwaite [Sat75] impossibility theorem shows that for a *sufficiently rich class* of agent preferences it is impossible to implement a satisfactory social choice function in dominant strategies. A related impossibility result, due to Green and Laffont [GJJ77] and Hurwicz [Hur75], demonstrates the impossibility of efficiency and budget-balance with dominant strategy implementation, even in quasi-linear environments.

More recently, the Myerson-Satterthwaite impossibility theorem [Mye83] extends this impossibility to include Bayesian-Nash implementation, if *interim* individual-rationality is also required. Williams [Wil99] and Krishna & Perry [KP98] provide alternative derivations of this general impossibility theorem, using properties about the Groves family of mechanisms.

Green & Laffont [GL79] demonstrate that no allocatively-efficient and strategy-proof mechanism can also be safe from manipulation by coalitions, even in quasi-linear environments.

The following sections describe the results in more details.

2.5.1 Gibbard-Satterthwaite Impossibility Theorem

A negative result due to Gibbard [Gib73] and Satterthwaite [Sat75] states that it is impossible, in a sufficiently rich environment, to implement a non-dictatorial social-choice function in dominant strategy equilibrium.

THEOREM 2.6 (Gibbard-Satterthwaite Impossibility Theorem). *If agents have general preferences, and there are at least two agents, and at least three different optimal outcomes over the set of all agent preferences, then a social-choice function is dominant-strategy implementable if and only if it is dictatorial.*

Clearly all dictatorial social-choice functions must be strategy-proof. This is simple to show because the outcome that is selected is the most preferred, or maximal outcome, for the reported preferences of one (or more) of the agents— so an agent should report its true preferences. For a proof in the other direction, that any strategy-proof social-choice function must be dictatorial, see MasColell *et al.* [MCWG95].

Impossibility results such as Gibbard-Satterthwaite must be interpreted with great care. In particular the results do not necessarily continue to hold in *restricted environments.* For example, although no dictatorial social choice function can be Pareto optimal or efficient, this impossibility result does not apply directly to markets. The market environment naturally imposes additional structure on preferences. In particular, the Gibbard-Satterthwaite impossibility theorem may not hold if one of the following conditions are relaxed:

— additional constraints on agent preferences (e.g. quasi-linear)

— weaker implementation concepts, e.g. Bayesian-Nash implementation

In fact a market environment has been shown to make implementation easier. Section 2.6 introduces a number of *non-dictatorial* and *strategy-proof* mechanisms in restricted environments; e.g. McAfee [McA92] for quasi-linear preferences in a double-auction, and Barberà & Jackson [BJ95] for quasi-concave preferences in a classic exchange economy.

2.5.2 Hurwicz Impossibility Theorem

The Hurwicz impossibility theorem [Hur75] is applicable to even *simple* exchange economies, and for agents with quasi-linear preferences. It states that it is impossible to implement an efficient and budget-balanced social choice function in dominant-strategy in market settings, even without requiring individual-rationality and even with additional restrictions on agent valuation functions.[1]

Hurwicz [Hur72] first showed a conflict between efficiency and strategy-proofness in a simple two agent model. The general impossibility result follows from Green & Laffont [GJJ77] and Hurwicz [Hur75], and more recently Hurwicz and Walker [HW90]. Green & Laffont and Hurwicz established that no member of the Groves family of mechanisms has budget-balance, and that the Groves family is the unique set of strategy-proof implementation rules in a simple exchange economy. I find it useful to refer to this result as the Hurwicz impossibility theorem.

THEOREM 2.7 (Hurwicz Impossibility Theorem). *It is impossible to implement an efficient, budget-balanced, and strategy-proof mechanism in a simple exchange economy with quasi-linear preferences.*

This result is quite negative, and suggests that if allocative efficiency and budget-balance are required in a simple exchange economy, then looking for dominant strategy solutions is not useful (via the revelation principle). Fortunately, strong budget-balance if often not necessary, and we can achieve strategy-proofness, efficiency and weak budget-balance via the Vickrey-Clarke-Groves mechanisms in a number of interesting domains.

2.5.3 Myerson-Satterthwaite Impossibility Theorem

The Myerson-Satterthwaite impossibility theorem [Mye83] strengthens the Hurwicz impossibility result to include Bayesian-Nash implementation, if *interim* individual-rationality is also required.

THEOREM 2.8 (Myerson-Satterthwaite). *It is impossible to achieve allocative efficiency, budget-balance and* (interim) *individual-rationality in a Bayesian-Nash incentive-compatible mechanism, even with quasi-linear utility functions.*

[1]Schummer [Sch97] has recently shown that even for the case of two agents with linear preferences it is not possible to achieve strategy-proofness and efficiency.

Name	Pref	Solution	Possible	Environment	
Groves	quasi-linear	dominant	Eff & (IR *or* WBB)	exchange	VCG
dAGVA	quasi-linear	Bayesian-Nash	Eff & BB	exchange	[dG79, Arr79]
Clarke	quasi-linear	dominant	Eff & IR	exchange	[Cla71]
GVA	quasi-linear	dominant	Eff, IR & WBB	comb auction	VCG
MDP	classic	iterative local-Nash	Pareto	exchange	[DdlVP71, Mal72 Rob79]
BJ95	classic	dominant	BB & non-dictatorial	exchange	[BJ95]
Quadratic	classic	Nash	Pareto & IR	exchange	[GL77b]

Table 2.2: Mechanism design: Possibility results. *Eff* is *ex post* allocative efficiency, *BB* is *ex post* strong budget-balance, *WBB* is *ex post* weak budget-balance, *IR* is *interim* individual-rationality, *Pareto* is *ex post* Pareto-optimality.

Myerson & Satterthwaite [Mye83] demonstrate this impossibility in a two-agent one-good example, for the case that trade is possible but not certain (e.g. the buyer and seller have overlapping valuation ranges). Williams [Wil99] and Krishna & Perry [KP98] provide alternative derivations of this general impossibility result, using properties of the Groves family of mechanisms.

An immediate consequence of this result is that we can only hope to achieve at most *two* of Eff, IR and BB in an market with quasi-linear agent preferences, even if we look for Bayesian-Nash implementation. The interested reader can consult Laffont & Maskin [LM82] for a technical discussion of various approaches to achieve any two of these three properties.

In the next section we introduce the dAGVA mechanism [Arr79, dG79], that is able to achieve efficiency and budget-balance, but loses individual-rationality. The dAGVA mechanism is an "expected Groves mechanism."

2.6 Possibility Results

The central positive result is the family of Vickrey-Clarke-Groves (VCG) mechanisms, which are allocatively-efficient (but not budget-balanced) strategy-proof mechanisms in quasi-linear domains. VCG mechanisms clearly demonstrate that it is possible to implement non-dictatorial social choice functions in more restricted domains of preferences. However, as expected from the impossibility results of Green & Laffont [GJJ77] and Hurwicz [Hur75], they are not efficient *and strong* budget-balanced.

Table 2.2 summarizes the most important possibility results. A quick check confirms that these possibility results are all consistent with the impossibility results of Table 2.1! By the revelation principle we effectively get "incentive-compatibility" for free in direct-revelation mechanisms, and these are all incentive-compatible except the iterative MDP procedure.

The possibility results are delineated by *agent preferences*, the *equilibrium solution concept* and the *environment* or problem domain.

We need a few additional definitions to explain the characterization.

DEFINITION 2.32 [classic preferences] Classic preferences are strictly quasi-concave, continuous and increasing utility functions.

DEFINITION 2.33 [exchange environment] Exchange simply refers to a bilateral trading situation, with agents that have general valuation functions (including bundle values).

Contrary to impossibility results, for possibility results a strong implementation concept is more useful than a weak implementation, e.g. dominant is preferred to Bayesian-Nash, and a general environment such as an exchange is preferred to a more restricted environment such as a combinatorial auction (which can be viewed as a one-sided exchange).

The Groves, Clarke (Pivotal), and GVA mechanisms have already been described in Section 2.4. Checking back with the impossibility results: Groves mechanisms are consistent with the Gibbard-Satterthwaite impossibility theorem because agent preferences are not general but quasi-linear;[2] and Groves mechanisms are consistent with the Hurwicz/Myerson-Satterthwaite impossibility theorems because strong budget-balance does not hold. Groves mechanisms are *not* strong budget-balanced. This failure of strong budget-balance can be acceptable in some domains; e.g., in one-sided auctions (combinatorial or otherwise) with a single seller and multiple buyers it may be acceptable to achieve *weak* budget-balance and transfer net payments to the seller.

2.6.1 Efficiency and Strong Budget-Balance: dAGVA

An interesting extension of the Groves mechanism, the *dAGVA* (or "expected Groves") mechanism, due to Arrow [Arr79] and d'Aspremont & Gérard-Varet [dG79], demonstrates that it is possible to achieve efficiency and budget-balance in a Bayesian-Nash equilibrium,

[2]MasColell also notes that there are no dictatorial outcomes in this environment.

even though this is impossible in dominant-strategy equilibrium (Hurwicz). However, the dAGVA mechanism is *not* individual-rational, which we should expect by the Myerson-Satterthwaite impossibility theorem.

THEOREM 2.9 (dAGVA mechanism). *The dAGVA mechanism is* ex ante *individual-rational, Bayesian-Nash incentive-compatible, efficient and (strong) budget-balanced with quasi-linear agent preferences.*

The dAGVA mechanism is a direct-revelation mechanism in which each agent announces a type $\hat{\theta}_i \in \Theta_i$, that need not be its true type. The mechanism is an "expected-form" Groves mechanism [Rob87, KP98].

The allocation rule is the same as for the Groves mechanism:

$$k^*(\hat{\theta}) = \max_{k \in \mathcal{K}} \sum_i v_i(k, \hat{\theta}_i)$$

The structure of the payment rule is also quite similar to that in the Groves mechanism:

$$t_i(\hat{\theta}) = h_i(\hat{\theta}_{-i}) - E_{\theta_{-i}} \left[\sum_{j \neq i} v_j(k^*(\hat{\theta}_i, \theta_{-i}), \theta_j) \right]$$

where as before $h(\cdot)$ is an arbitrary function on agents' types. The second term is the expected total value for agents $j \neq i$ when agent i announces type $\hat{\theta}_i$ and agents $j \neq i$ tell the truth. It is a function of only agent i's announcement, not of the actual strategies of agents $j \neq i$, making it a little different from the formulation of agent transfers in the Groves mechanism. In effect, agent i receives a transfer due to this term equal to the *expected* externality of a change in its own reported type on the other agents in the system.

The Bayesian-Nash incentive-compatibility with this transfer follows from a similar line of reasoning as the strategy-proofness of the Groves mechanisms. A proof is in the appendix to this chapter.

The interesting thing about the dAGVA mechanism is that it is possible to choose the $h_i(\cdot)$ functions to satisfy budget-balance, such that $\sum_i t_i(\theta) = 0$ for all $\theta \in \Theta_i$. Define the "expected social welfare (or value)" of agents $j \neq i$ when agent i announces its type θ_i as

$$\mathrm{SW}_{-i}(\hat{\theta}_i) = E_{\theta_{-i}} \left[\sum_{j \neq i} v_j(k^*(\hat{\theta}_i, \theta_{-i}), \theta_j) \right]$$

and note that this does not depend on announced types of agents $j \neq i$. The additional term in the payment rule is defined, for agent i, as:

$$h_i(\hat{\theta}_{-i}) = \left(\frac{1}{I-1}\right) \sum_{j \neq i} \mathrm{SW}_{-j}(\hat{\theta}_j)$$

which is the "averaged" expected social welfare to every other agent given the announced types of agents $j \neq i$. This gives budget-balance because each agent also pays an equal $1/(I-1)$ share of the total payments made to the other agents, none of which depend on its own announced type. See the appendix of this chapter for a proof.

The incentive properties and properties of full optimality, i.e. efficiency and budget-balance, make the dAGVA procedure very attractive. However, the dAGVA mechanism has a number of problems:

(1) it may not satisfy the individual rationality constraint (even *ex ante*)

(2) Bayesian-Nash implementation is much weaker than dominant-strategy implementation

(3) it places high demands on agent information-revelation

Roberts [Rob87] provides a very interesting discussion of the conditions required for an iterative method to implement the dAGVA mechanism with less information from agents. In fact, he claims that it is *impossible* to find a successful iterative procedure because an agent's announcement in earlier periods must also affect its payments in subsequent periods, breaking incentive-compatibility.

2.6.2 Dominant-strategy Budget-Balance with Inefficient Allocations

A number of mechanisms have been proposed to achieve budget-balance (perhaps weak budget-balance) in dominant strategy mechanisms, for some loss in allocative efficiency. McAfee [McA92] presents a mechanism for a double auction (with multiple buyers and sellers) that is strategy-proof and satisfies weak budget-balance, but for some loss in allocative efficiency.

Barberà & Jackson [BJ95] characterize the set of strategy-proof social-choice functions that can be implemented with budget-balance in an exchange economy with classic preferences. Comparing back with the Gibbard-Satterthwaite impossibility theorem [Gib73, Sat75], it is possible to implement non-dictatorial social choice functions in this restricted set of preferences, even though preferences are not quasi-linear. In fact Barberà

& Jackson show that it is necessary and sufficient to implement "fixed-proportion trading rules", in which (loosely speaking) agents trade in pre-specified proportions. Given Hurwicz's impossibility theorem, it is not surprising that the trading rules are not fully allocatively-efficient.

2.6.3 Alternative implementation Concepts

One method to extend the range of social-choice functions that can be implemented is to consider alternative equilibrium solution concepts. In the context of direct-revelation mechanisms (i.e. static games of incomplete information) we have already observed that Bayesian-Nash implementation can help (e.g. in the dAGVA mechanism). One difficulty with Bayesian-Nash implementation is that it requires more information and rationality assumptions of agents. Similarly, we might expect that moving to a Nash implementation concept can help again.

Groves & Ledyard [GL77] inspired much of the literature on Nash implementation. The *Quadratic mechanism* is Pareto efficient in the exchange environment with classic preferences, in that all Nash equilibria are Pareto efficient. In this sense, it is demonstrated that it is possible to implement budget-balanced and efficient outcomes with Nash implementation, while (Myerson-Satterthwaite) this is not possible with Bayesian-Nash.

However, the Nash implementation concept is quite problematic. An agent's Nash strategy depends on the strategies of other agents, and on complete information about the (private) types of each agent. Clearly, it is quite unreasonable to expect agents to select Nash strategies in a one-shot direct-revelation mechanism. The solution concepts only make sense if placed within an iterative procedure, where agents can adjust towards Nash strategies across rounds [Gro79].

Moore & Rupullo [MR88] consider subgame-perfect Nash implementation in dynamic games, and show that this expands the set of social-choice functions that can be implemented in strategy-proof mechanisms. Of course, introducing a new solution concept requires a new justification of the merits of the subgame-perfect refinement to Nash equilibrium in a dynamic game. A fascinating recent idea, due to Kalai & Ledyard [KL98] considers "repeated implementation", in which the authors consider the implementable social-choice functions in a repeated game, with strong results about the effect on implementation.

The mechanism design literature is almost exclusively focused on direct-revelation mechanisms and ignores the costs of information revelation and centralized computation. One exception is the *MDP* planning procedure, proposed by Drèze & de la Vallée Poussin [DdlVP71] and Malinvaud [Mal72]. The MDP mechanism is an iterative procedure, in which in each round each agent announces "gradient" information about its preferences for different outcomes. The center adjusts the outcome towards a Pareto optimal solution in an exchange environment with classic agent preferences. If the agents report truthful information the MDP procedure is Pareto optimal (i.e. fully efficient).

Drèze & de la Vallée Poussin [DdlVP71] also consider the incentives to agents for reporting truthful information in each round, and showed that truthful revelation is a local maximin strategy (i.e. maximizes the utility of an agent given that other agents follow a worst-case strategy). Truth revelation is also a Nash equilibrium at termination.

In addition, Roberts [Rob79] proved that if agents play a local Nash equilibrium at each stage in the procedure, to maximize the immediate increase in utility of the project, then the mechanism will still converge to a Pareto optimum even though the agents do not report truthful information. Roberts retains a myopic assumption, and studied only a local game in which agents did not also consider the effect of information on future rounds.

Champsaur & Laroque [CL82] departed from this assumption of myopic behavior, and assumed that every agent considers the Nash equilibrium over a period of T periods. The agents forecast the strategies of other agents over T periods, and play a Nash equilibrium. The MDP procedure is still Pareto optimal, but the main difference is that the center has much less control over the final outcome (it is less useful as a policy tool). The outcome for large T approaches the competitive equilibrium.

Modeling agents with a Nash equilibrium, even in the local game, still makes the (very) unreasonable assumption that agents have complete information about each others' preferences, for example to compute the equilibrium strategies. Roberts [Rob79] discusses iterative procedures in which truthful revelation locally dominant at each stage. Of course, one must expect some loss of efficiency if strategy-proofness is the goal.

*i*Bundle [Par99, PU00a] is an efficient ascending-price auction for the combinatorial allocation problem, with myopic best-response agent strategies. The auction is weak budget-balanced, and individual-rational. Although myopic best-response is not a rational sequential strategy for an agent, it is certainly a more reasonable implementation concept than a local Nash strategy, requiring only price information and information about an agent's own preferences. As discussed in Chapter 7, an extended auction, *i*Bundle Extend&Adjust, provably computes VCG payments in many problems. Computing the outcome of a Groves mechanism with myopic best-response strategies makes myopic best-response a Bayesian-Nash equilibrium of the iterative auction.

2.7 Optimal Auction Design

In a seminal paper, Myerson [Mye81] adopted a constructive approach to mechanism design for *private-values* auction, in which an agent's value is independent of that of other agents. Myerson considers an objective of *revenue maximization*, instead of allocative-efficiency, and formulates the mechanism design problem as an optimization problem. The objective is to design an outcome function for a direct-revelation mechanism that maximizes the expected revenue subject to constraints on: *feasibility* (no item can be allocated more than once); *individual-rationality* (the expected utility for participation is non-negative); and *incentive-compatibility.*

Focusing on direct-revelation mechanisms (following the revelation principle), Myerson derives conditions on the allocation rule $k : \Theta \rightarrow \mathcal{K}$ and the payment rules $t_i : \Theta \rightarrow \mathbb{R}$ for an auction to be optimal. Without solving for explicit functional forms $k(\cdot)$ and $t_i(\cdot)$ Myerson is able to derive the *revenue equivalence theorem*, which essentially states that any auction that implements a particular allocation rule $k(\cdot)$ must have the same expected payments.

In general the goals of revenue-maximization and efficiency are in conflict. Myerson constructs an optimal (revenue-maximizing) auction in the simple single-item case, and demonstrates that a seller with distributional information about the values of agents can maximize its expected revenue with an inefficient allocation-rule. The seller announces a non-zero reservation price, which increases its revenue in some cases but also introduces a slight risk that the seller will miss a profitable trade (making it inefficient).

Krishna & Perry [KP98] develop a generalized revenue-equivalence principle:

THEOREM 2.10 (generalized revenue-equivalence). *In quasi-linear environments, all Bayesian-Nash incentive-compatible mechanisms with the same choice rule $k(\cdot)$ are expected revenue equivalent up to an additive constant.*

This is essentially a statement that all mechanisms that implement a particular allocation rule are equivalent in their transfer rules. We have already seen a similar result, i.e. that the Groves mechanisms are unique among efficient & strategy-proof mechanisms [GL87].

Krishna & Perry also show that the GVA maximizes revenue over all efficient and individual-rational mechanisms, even amongst mechanisms with Bayesian-Nash implementation:

THEOREM 2.11 (Revenue-optimality of GVA). *The GVA mechanism maximizes the expected revenue amongst all efficient, (Bayesian-Nash) incentive-compatible, and individual-rational mechanisms.*

It is interesting that the dominant-strategy GVA mechanism maximizes revenue over all Bayesian-Nash incentive-compatible and efficient mechanisms.

Ausubel & Cramton [AC98] make a simpler argument for efficient mechanisms in the presence of after-markets. Intuitively, in the presence of an after-market that will allow agents to achieve an efficient allocation outside of the auction, the auctioneer maximizes profits by providing agents with an allocation that they find most desirable and extracting their surplus. A similar argument can be made in the presence of alternate markets. If the auctioneer does not compute efficient allocations then agents will go elsewhere.

Appendix: Proof of dAGVA properties

The intuition behind the Bayesian-Nash incentive-compatibility of the dAGVA mechanism follows a similar line of reasoning to that for the strategy-proofness of Groves. Suppose that the other agents announce their true types, the expected utility to agent i for announcing its true type (given correct information about the distribution over the types of other agents) is:

$$E_{\theta_{-i}}\left[v_i(k^*(\theta_i,\theta_{-i}),\theta_i)\right] + E_{\theta_{-i}}\left[\sum_{j\neq i} v_j(k^*(\theta_i,\theta_{-i}),\theta_j)\right]$$
$$= E_{\theta_{-i}}\left[\sum_{j=1}^{I} v_j(k^*(\theta),\theta_j)\right]$$

and this is greater than

$$E_{\theta_{-i}}\left[\sum_{j=1}^{I} v_j(k^*(\hat{\theta}_i,\theta_{-i}),\theta_j)\right]$$

for all $\hat{\theta}_i \in \Theta_i$ because by reporting its true type the agent explicitly instructs the mechanism to compute an allocation that maximizes the inner-term of the expectation for all possible realizations of the types θ_{-i} of the other agents.

Finally, we show that the dAGVA mechanism is budget-balanced:

$$\begin{aligned}
\sum_i t_i(\theta) &= \left(\frac{1}{I-1}\right)\sum_i \sum_{j\neq i} \mathrm{SW}_{-j}(\theta_j) - \sum_i \mathrm{SW}_{-i}(\theta_i) \\
&= \left(\frac{1}{I-1}\right)\sum_i (I-1)\mathrm{SW}_{-i}(\theta_i) - \sum_i \mathrm{SW}_{-i}(\theta_i) \\
&= 0
\end{aligned}$$

Intuitively, each agent i receives a payment equal to $\mathrm{SW}_{-i}(\theta_i)$ for its announced type, which is the expected social welfare effect on the other agents. To balance the budget each agent also pays an equal $1/(I-1)$ share of the total payments made to the other agents, none of which depend on its own announced type.