Les Cahiers du GERAD ISSN: 0711–2440

The Return Function: A Tool for Computing Bayesian-Nash Equilibria in Mechanism Design

L. Nguyên Hoang, F. Soumis,
G. Zaccour

G–2012–70

November 2012

The Return Function: A Tool for Computing Bayesian-Nash Equilibria in Mechanism Design

Lê Nguyên Hoang
François Soumis

GERAD & École Polytechnique de Montréal
Montréal (Québec) Canada, H3C 3A7
len.hoang.lnh@gmail.com
francois.soumis@gerad.ca

Georges Zaccour

GERAD & HEC Montréal
Montréal (Québec) Canada, H3T 2A7
georges.zaccour@gerad.ca

November 2012

Les Cahiers du GERAD

G–2012–70

Abstract

In many situations, such as art auctions, privatization of public assets and allocation of television airwaves to wireless carriers, the value of the object on sale (product, service or asset) is not known beforehand, and a market has to be designed to determine its price. The market (or mechanism) designer has to set the rules of the game in a context where typically none of the parties has complete information about the preferences of the others. Finding a solution amounts at determining some Bayesian-Nash equilibria to that game, given that the designer has an interest in the outcome. In this article, we introduce the idea of return function, and use it to compute Bayesian-Nash equilibria in mechanism design. In a nutshell, given a player's choice of action, the other players' strategies and the mechanism chosen by the market designer, the return function of that given player is the density of the means of the probability distribution function of the outcome. Further, we define and consider optimality concepts for general forms of the principal's objective function. We also introduce the ideality gap function to assess the difference between the optimality of a mechanism with perfect information and its implementability constraint. Finally, we give a method for computing Bayesian-Nash equilibria and optimizing mechanisms, which is based on the return function.

Key Words: Mechanism Design; Return Function; Bayesian-Nash Equilibrium: Cake-Cutting Problem.

Résumé

De nombreux problèmes tels que la régulation des marchés, les enchères ou les choix politiques consistent à définir un système d'interaction entre différents agents. Ceci correspond à un problème de conception de mécanisme, dont la difficulté réside dans le fait que les agents agissent de façon rationnelle et égoïste. Ainsi, il n'est pas raisonnable de s'attendre à ce qu'ils soient honnêtes. Afin d'anticiper leurs actions, nous proposons une nouvelle méthode fondée sur la fonction de retour, qui aux actions des joueurs associe la densité de probabilité du résultat. Cette fonction est estimée empiriquement, et permet ensuite de développer une optimisation de mécanisme.

1 Introduction

Mechanism (or market) design has proven to be a successful approach for efficiently determining the value of a product or a service when there is no natural price that can be posted or negotiated for that product, as the case for, e.g., a painting by Picasso, energy prices in a deregulated electricity market, or the exploitation rights for a hydrocarbon-rich basin. These examples, and many others, share the following features: (i) There is a finite number of strategic agents (players, bidders or claimers) interested in acquiring the object (product, service or resource). (ii) Each agent has a *private* value for the object under consideration, and does not know how much the other agents value the same object. For instance, the cost of producing a kilowatt is not the same for all electricity companies participating in a given market, and each company knows its own cost, but only has incomplete knowledge of its competitors' costs. Similarly, a Picasso painting does not have the same value for all art collectors. (iii) The rules of the game are not given in advance, but are designed by an agent, called a mechanism designer, principal or regulator, who has an interest in the outcome. For instance, a public commission may auction off television airwaves to wireless carriers to create faster and more reliable networks, or to maximize its own revenues. A parent may ask kids at a party to express their preferences for different flavors, to fairly allocate a heterogeneous birthday cake.

One important issue that must be dealt with when designing a mechanism is that the agents may not behave truthfully. For instance, in the cake-cutting problem, some participants may hide their preferences in the hopes of obtaining a better piece than the one they would get by telling the truth. In such context, the relevant question is whether is it possible to design a mechanism in such a way that the participants in the game would find it in their best interest to truthfully report their private information. This question attracted the attention of economists, game theorists and operations research analysts.

The first focus was on designing auctions. In a seminal paper, Vickrey (1961) proposed second-highest-price auction, where the highest bidder wins the contract but pays the second-highest proposed sealed bid. The bidder with the highest valuation is more motivated to reveal his true valuation of the object in this type of auction than in other traditional auctions such as the classical English auction, where the highest bidder pays his actual bid. Another cornerstone in building the theory of mechanism design is the revelation principle, introduced by Gibbard (1973) for dominant strategies, and later generalized to any strategy by Dasgupta, Hammond, and Maskin (1979), Holmstrom (1977) and and Myerson (1979). This principle states that for any Bayesian-Nash equilibrium of a game of incomplete information, there exists a payoff-equivalent revelation mechanism that has an equilibrium where the players truthfully report their types. The revelation principle greatly simplifies the task of finding a mechanism-design solution, because the designer only needs to look at the set of equilibria characterized by incentive compatibility. This means that if the designer wants to reach some outcome or implement some property, then he can restrict his search to mechanisms in which players are willing to reveal their private information to the mechanism designer that has that outcome or property. If no such direct and truthful mechanism exists, then the conclusion is that no mechanism can be implemented to achieve this outcome or property. By narrowing the search, the problem of finding a mechanism becomes much easier.

During the last two decades or so, important developments in market design have taken place for three main reasons: "(i) The creation by government agencies, private firms or industrial associations of a number of markets to privatize public assets, restructure deregulated industries, or enhance inter-firm relations; (ii) a renewed focus on strategic analysis and game theory that together with the emergence of experimental economics contributed to the establishment of market design as a serious research field in economics; (iii) and, most importantly, the explosive development of electronic business, e-business tools that can embed the most complex market rules and facilitate their deployment." (Bourbeau et al. (2005)). Recent operations-research literature include work on assignment problems: see, e.g., Su and Zenios (2006) for kidney-transplant trade-off; Abdulkadirolu and Sönmez (2003) or Pathak (2011) for school choice; in supply chains, see, e.g., Jain and Raghavan (2009), Chen and Cheng (2012), Mes et al. (2011); and in revenue management, see, e.g., Vulcano et al. (2002), Manelli and Vincent (2007), Devenur and Hayes (2009).

In this article, we propose a new tool, which we call the return function, to compute Bayesian-Nash equilibria in the context of mechanism design. In a nutshell, given a player's choice of action, the other players' strategies and the mechanism chosen by the market designer, the return function of that given player is the density of the means of the probability distribution function of the outcome. Given this return function, the expected utility of the outcome of a player is then defined as a function of this return function and the player's type. Our formulation is fairly general and accounts for outcomes that cannot be defined deterministically. Further, we define and consider optimality concepts for general forms of the principal's objective function. We also introduce the ideality gap function to assess the difference between the optimality of a mechanism with perfect information and its implementability constraint. Finally, we give a method for computing Bayesian-Nash equilibria and optimizing mechanisms, which is based on the return function.

The rest of the paper is organized as follows: In Section 2, we introduce the return function and recall the definition of an equilibrium. In Section 3, we define some mechanism designs and establish some of their properties. Section 4 is devoted to showing how the return function can be used to compute Bayesian-Nash equilibria. In Section 5, we provide an illustration in the context of a cake-cutting problem. Section 6 concludes.

2 Model and Equilibrium

Let $N = \{1, \ldots, n\}$ be the set of players and A the set of actions of player $j \in N$. Denote by a_j an action of j, by $a = (a_1, \ldots, a_n) \in A^n$, the vector of all players' actions, and by $a_{-j} = (a_1, \ldots, a_{j-1}, a_{j+1,\ldots}, a_n) \in A^{n-1}$, the vector of the actions of all players other than j. Similar notations will be used throughout the paper for vectors of objects that refer to all players, or to $n-1$ players. Also, if E and F are two sets, then the set of functions from E to F will be denoted $\mathcal{F}(E, F)$. For clarity of exposition, we are assuming here that the set of actions is the same for all players. Admittedly, this is not the most general formulation, but in principle, there is no conceptual difficulty in extending the analysis to the case where the players have different action sets.

Given the players' choice of $a \in A^n$, the mechanism designer selects an outcome x from the set of admissible outcomes X, according to a certain mechanism $\mathcal{M}$, that is, $\mathcal{M}(a) = x \in X$. To illustrate, x could be, e.g., the workers' schedule for a given week, the shares of a cake allocated to the different claimers, or the quantity of energy to be supplied the next day by the bidding electricity companies. The mechanism $\mathcal{M}$ could be, e.g., one that minimizes the total system cost or maximizes customer satisfaction. By admissible outcomes, we mean that X contains only those allocations that satisfy all problem constraints, such as satisfying workers' contracts and companies service requirements, full allocation of the cake, etc.

In practice, this outcome may not be deterministically specified, because of some inherent random events. For instance, the next day's electricity demand depends on temperature, which cannot be predicted with certainty. To reflect this, we let mechanism $\mathcal{M}$ to be function in $\Delta(X)$, where $\Delta(X)$ is the set of probability density functions (PDFs), over the set of outcomes X. More specifically,

Definition 1 *A mechanism $\mathcal{M}$ is a function that, to actions $a \in A^n$, associates a PDF $\mathcal{M}(a) \in \Delta(X)$ of the outcome. The set of admissible mechanisms is $\mathbb{M} = \mathcal{F}(A^n, \Delta(X))$.*

We make the following remarks:

Remark 1 *If the mechanism is deterministic, then $\mathcal{M}(a)$ would have a Dirac distribution.*

Remark 2 *If the set of admissible PDFs must be restricted, for any reason, to Δ_X, with $\Delta_X \subset \Delta(X)$, then the set of admissible mechanisms will consequently be restricted to $\mathbb{M} = \mathcal{F}(A^n, \Delta_X)$.*

Remark 3 *If the set of admissible actions A is chosen by the mechanism designer, then the set of admissible mechanisms is the union of sets, that is, $\mathbb{M} = \bigcup_A \mathcal{F}(A^n, \Delta_X)$.*

The context covered by this last remark is natural in some applications, e.g., personnel scheduling, where the set of possible work shifts from which workers can choose is set by management (mechanism designer).

Each player $j \in N$ is defined by his type θ_j. Now, for each player j, a utility function matches his type and the outcome with a real number, as follows:

$$u_j(\theta_j, x) \geq u_j(\theta_j, x') \qquad \text{for} \quad x \succeq x',$$

where the symbol $\succeq$ means preferred to. Let $\theta = (\theta_1, \ldots, \theta_n) \in \Theta^n$, where Θ is the set of types, assumed to be the same for all players. As we are dealing with mechanisms $\mathcal{M}(a) \in \Delta(X)$, we extend the domain of definition of $u_j(\theta_j, .)$ to the set $\Delta(X)$ of PDFs over outcomes by considering that, for all $\theta_j \in \Theta$ and all $\chi \in \Delta(X)$,

$$u_j(\theta_j, \chi) = \mathbb{E}_{x \rightsquigarrow \chi}[u_j(\theta_j, x)],$$

where $x \rightsquigarrow \chi$ means that the random variable x has the probability density function χ.

Remark 4 *The assumption that the set of types is the same for all players can easily be relaxed by defining a set Θ_j for each $j \in N$. As we will be considering PDFs over types, this would then be equivalent to saying that each player has a set of types $\Theta = \bigcup_{j \in N} \Theta_j$ with a nil distribution over $\Theta - \Theta_j$.*

As usual, we assume that each player knows his type, and has incomplete knowledge of the other players' types. We also suppose that the principal only has incomplete knowledge of the players' types. Denote by $f \in \Delta(\Theta^n)$ the PDFs over types of all players, and by $f_{-j} \in \Delta(\Theta^{-j})$ the PDFs over types of all other players but j. The PDF f_{-j} represents beliefs held by player j about the other players' types. Note that we assume, for generality, that f_{-j} does not depend on the type of player $j \in N$.

Denote by σ_j a strategy of player j, and by Σ the set of strategies of this player. As for the set of actions, we assume, without any loss of generality, that the set of strategies is the same for all players. A strategy is a mapping that associates an action a_j to a type, i.e., $\sigma_j(\theta_j) = a_j$. Let $\sigma = (\sigma_1, \ldots, \sigma_n) \in \Sigma^n$.

2.1 Return Function

For each player j, we define a return function φ_j that, to an action $a_j \in A$, a vector of other players' strategies $\sigma_{-j} \in \Sigma^{-j}$, a PDF $f_{-j} \in \Delta(\Theta^{-j})$ and a mechanism $\mathcal{M} \in \mathbb{M}$, associates a PDF over the set of outcomes, i.e.,

$$\varphi_j \in \mathcal{F}(A \times \Sigma^{-j} \times \mathbb{M} \times \Delta(\Theta^{-j}), \Delta(X)),$$

that is,

$$\varphi_j(a_j, \sigma_{-j}, \mathcal{M}, f_{-j}) = \mathbb{E}_{\theta_{-j} \rightsquigarrow f_{-j}}[\mathcal{M}(a_j, \sigma_{-j}(\theta_{-j}))]. \tag{1}$$

Put differently, the return function $\varphi_j(a_j, \sigma_{-j}, \mathcal{M}, f_{-j})$ is the density of the means of the PDFs of the outcome when: player j plays action a_j, the other players use strategies σ_{-j}, the vector of types of the other players have the PDF f_{-j} and the principal chooses the mechanism $\mathcal{M}$. Given φ_j, the expected utility of the outcome of player j with type θ_j is equal to $u_j(\theta_j, \varphi_j(a_j, \sigma_{-j}, \mathcal{M}, f_{-j}))$.

Remark 5 *When all players have the same PDF for θ_{-j}, it is natural to assume that $\varphi_j(\cdot) = \varphi(\cdot), \forall j \in N$. This would occur in, e.g., games with a large number of players, where excluding a player will not fundamentally affect the probabilistic representation of the game for the other players. Clearly, in that case the mechanism needs to be symmetric, that is, for any permutation $\tau : N \to N$, we have the following property:*

$$\forall v \in V^n, \forall a \in A^n, \forall j \in N, \theta_j = \theta_{\tau(j)} \Rightarrow u_j(\theta_j, \mathcal{M}(a_j)) = u_{\tau(j)}(\theta_{\tau(j)}, \mathcal{M}(a_{\tau(j)})). \tag{2}$$

We will provide later on an illustrative example with symmetric players. We end this section by recalling the definitions of best-reply strategies and Bayesian-Nash equilibria.

Definition 2 *The set of best-reply strategies σ_j^{BR} for player j to strategies σ_{-j} of the other players with a PDF f_{-j}, and a given mechanism $\mathcal{M} \in \mathbb{M}$ selected by the principal is given by*

$$\Sigma_j^{BR}(\sigma_{-j}, \mathcal{M}, f_{-j}) = \arg\max_{\sigma_j \in \Sigma} \mathbb{E}_{\theta_j \rightsquigarrow f_j}\left[u_j(\theta_j, \varphi_j(\sigma_j(\theta_j), \sigma_{-j}, \mathcal{M}, f_{-j}))\right]. \tag{3}$$

The set $\Sigma^{BR}(\sigma, \mathcal{M}, f)$ of vectors σ^{BR} of best-reply strategies for all players is given by

$$\Sigma^{BR}(\sigma, \mathcal{M}, f) = \Big\{\sigma^{BR} \in \Sigma^n \mid \forall j \in N, \sigma_j^{BR} \in \Sigma_j^{BR}(\sigma_{-j}, \mathcal{M}, f_{-j})\Big\}. \tag{4}$$

Definition 3 *The set $\Sigma^{BN}(\mathcal{M}, f)$ of Bayesian-Nash equilibria is the set of strategies that are best replies against themselves, i.e.,*

$$\Sigma^{BN}(\mathcal{M}, f) = \Big\{\sigma^{BN} \in \Sigma^n \mid \sigma^{BN} \in \Sigma^{BR}(\sigma^{BN}, \mathcal{M}, f)\Big\}. \tag{5}$$

3 Mechanism Design

Up to now, everything has been done assuming a given mechanism, $\mathcal{M} \in \mathbb{M}$. In this section, we deal with the choice of a mechanism that yields some desirable outcome to the designer. To do this, we will define the notion of a *direct mechanism* and then introduce an objective function for the mechanism designer. We will show some theoretical results and describe a practical method to optimize mechanisms. The revelation principle will be a key component in the forthcoming construction.

3.1 Direct Mechanism

If the principal knew the different players' types, then he could straightforwardly use them to design a direct mechanism and select an admissible outcome. More specifically:

Definition 4 *A direct mechanism $\mathcal{D} \in \mathbb{D} = \mathcal{F}(\Theta^n, \Delta_X)$ is a function that, to a vector of types $\theta \in \Theta^n$, associates an admissible PDF over outcomes $\mathcal{D}(\theta) \in \Delta_X$.*

Note that, whereas an admissible mechanism $\mathcal{M}$ maps the set of actions A^n into Δ_X, a direct mechanism $\mathcal{D}$ maps the set of types into Δ_X. The two mechanisms would be the same if the set of players' actions coincided with the set of types. In the example provided later on, we suppose such a coincidence. The introduction of this direct mechanism will allow us to analyze the optimality of mechanisms when the players behave truthfully.

A strategy profile σ^{truth} is truthful if it satisfies $\sigma^{truth}(\theta) = \theta$, $\forall \theta \in \Theta^n$.

Definition 5 *A direct mechanism $\mathcal{D}$ is Bayesian Incentive Compatible (BIC) if the strategy profile σ^{truth} is a Bayesian-Nash equilibrium, that is, $\sigma^{truth} \in \Sigma^{BN}(\mathcal{D}, f)$.*

A Bayesian Incentive Compatible (BIC) direct mechanism $\mathcal{D}$ is denoted $\mathcal{D}_{BIC}$. The set of $\mathcal{D}_{BIC}$ mechanisms depends on the PDF f over types, and is denoted $\mathbb{D}_{BIC}(f)$.

Definition 6 *A direct mechanism is Dominant Strategy Incentive Compatible (DSIC) if σ_j^{truth} is a dominant strategy for all $j \in N$.*

The set of DSIC direct mechanisms is denoted $\mathbb{D}_{DSIC}$. The difference between the above two definitions is straightforward. In the first case, players have an incentive to be truthful, provided the other players are truthful. In the second case, players have an incentive to be truthful no matter what the others are.

The following theorem characterizes the relationship between the DSIC and BIC mechanisms:

Theorem 1 *A direct mechanism $\mathcal{D}$ is a DSIC direct mechanism if and only if it is a BIC direct mechanism for any PDF f, i.e.,*

$$\mathbb{D}_{DSIC} = \bigcap_{f \in \Delta(\Theta^n)} \mathbb{D}_{BIC}(f). \tag{6}$$

Proof. Clearly, a DSIC direct mechanism is always BIC. Hence, $\mathbb{D}_{DSIC} \subset \bigcap_{f \in \Delta(\Theta^n)} \mathbb{D}_{BIC}(f)$.

Now, consider a direct mechanism $\mathcal{D} \in \bigcap_{f\in\Delta(\Theta^n)} \mathbb{D}_{BIC}(f)$. We need to prove that $\mathcal{D}$ is DSIC. For this, we must show that for $\sigma_{-j}^{truth}(\theta_{-j})$, the best response of player j is $\sigma_j^{truth}(\theta_j)$, for all $\theta \in \Theta^n$. This is equivalent to saying that $\mathcal{D} \in \mathbb{D}_{BIC}(\delta(\theta))$, where $\delta(\theta)$ is the density of the Dirac distribution with support $\prod_{j\in N}\{\theta_j\}$. Since $\delta(\theta) \in \Delta(\Theta^n)$, we have $\bigcap \mathbb{D}_{BIC}(f) \subset \mathbb{D}_{BIC}(\delta(\theta))$, which implies that $\mathcal{D} \in \mathbb{D}_{BIC}(\delta(\theta))$. This proves the opposite inclusion. □

As recalled in the introduction, the revelation principle states that for any Bayesian-Nash equilibrium of a game of incomplete information, there exists a payoff-equivalent revelation mechanism that has an equilibrium in which the players truthfully report their types. Following a similar idea, we construct a revelation-direct mechanism as follows: given a mechanism $\mathcal{M} \in \mathbb{M}$ and a Bayesian-Nash equilibrium $\sigma^{BN} \in \Sigma^{BN}(\mathcal{M}, f)$, and given the players' types θ, we compute the corresponding actions $a = \sigma^{BN}(\theta)$, and the PDF on the outcomes x given by $\mathcal{M}(a)$.

Denote by the symbol "$\circ$" the composition of functions. To any mechanism $\mathcal{M} \in \mathbb{M}$, and any Bayesian-Nash equilibrium $\sigma^{BN} \in \Sigma^{BN}(\mathcal{M}, f)$, we associate a revelation-direct mechanism $\mathcal{D}_R = \mathcal{M} \circ \sigma^{BN}$. The outcome is then given by $\mathcal{D}_R(\theta) = \mathcal{M}(\sigma^{BN}(\theta))$. The set of revelation-direct mechanisms is denoted $\mathbb{D}_R(f)$.

A graphical representation of the construction of the revelation-direct mechanism is given in Figure 1. In this figure, there are three main sets we have to deal with: the set of types Θ^n, the set of actions A^n and the set of admissible PDFs over outcomes Δ_X. Note that σ^{BN} depends on the PDF f. We now have the following result, which is a corollary of the revelation principle.

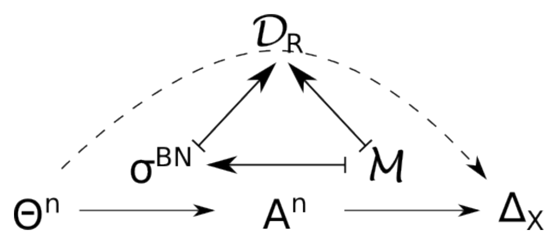

Figure 1: Construction of the revelation-direct mechanism

Theorem 2 *The set of constructed revelation-direct mechanisms coincide with the set of BIC direct mechanisms, i.e.,*

$$\forall f \in \Delta(\Theta^n), \mathbb{D}_R(f) = \mathbb{D}_{BIC}(f).$$

Proof. By definition, for any $\mathcal{D}_{BIC}$, the truthful strategy profile is a Bayesian-Nash equilibrium. Therefore, the constructed mechanism $\mathcal{D}_{BIC} \circ \sigma^{truth}$ is the mechanism $\mathcal{D}_{BIC}$. Hence, any BIC direct mechanism is a revelation-direct mechanism.

Reciprocally, let us a consider a revelation-direct mechanism $\mathcal{D}_R = \mathcal{M} \circ \sigma^{BN}$. We need to show that $\mathcal{D}_R$ is BIC. Denote by σ^R the players' strategies when playing $\mathcal{D}_R$. Assume player j's type is θ_j. The PDF over outcomes is given by the return function:

$$\begin{aligned}
\varphi_j(\sigma_j^R(\theta_j), \sigma_{-j}^R, \mathcal{D}_R, f_{-j}) &= \mathbb{E}_{\theta_{-j}\rightsquigarrow f_{-j}}\left[\mathcal{D}_R(\sigma_j^R(\theta_j), \sigma_{-j}^R(\theta_{-j}))\right] &(7)\\
&= \mathbb{E}_{\theta_{-j}\rightsquigarrow f_{-j}}\left[\mathcal{M}(\sigma_j^{BN}(\sigma_j^R(\theta_j)), \sigma_{-j}^{BN}(\sigma_{-j}^R(\theta_{-j})))\right] &(8)\\
&= \varphi_j(\sigma_j^{BN}(\sigma_j^R(\theta_j)), \sigma_{-j}^{BN} \circ \sigma_{-j}^R, \mathcal{M}, f_{-j}) &(9)
\end{aligned}$$

Suppose now that all players but j are truthful, that is $\sigma_{-j}^R = \sigma_{-j}^{truth}$. Therefore, $\sigma_{-j}^{BN} \circ \sigma_{-j}^R = \sigma_{-j}^{BN}$. As a result, and since σ_j^{BN} is the best reply of σ_{-j}^{BN} when playing mechanism $\mathcal{M}$, we have the following upper bound on player j's utility of the outcome:

$$u_j(\theta_j, \varphi_j(\sigma_j^R(\theta_j), \sigma_{-j}^{truth}, \mathcal{D}_R, f_{-j})) = u_j(\theta_j, \varphi_j(\sigma_j^{BN}(\sigma_j^R(\theta_j)), \sigma_{-j}^{BN}, \mathcal{M}, f_{-j})) \tag{10}$$

$$\leq u_j(\theta_j, \varphi_j(\sigma_j^{BN}(\theta_j), \sigma_{-j}^{BN}, \mathcal{M}, f_{-j})). \tag{11}$$

The equality is achieved for $\sigma_j^R = \sigma_j^{truth}$. Therefore, for the revelation-direct mechanism, for any player, truthfulness is a best reply to the other players' truthfulness. Equivalently, the revelation-direct mechanism is BIC. □

For the mechanism designer, there are two advantages to using the revelation-direct mechanism, namely: (i) he knows that it is in the best interest of the players to act truthfully; (ii) he learns, in the sense of having a better approximation of, the PDF of the players' types. From the point of view of the players, the equilibrium strategy profile is "simple," that is, not many computations are involved in determining the equilibrium strategies.

3.2 Principal's Objective Function

The principal's objective could be to achieve an optimal result in terms of profit, utility, cost, etc., or to reach an outcome having some desirable properties, e.g., fairness. To illustrate, let us consider a liberalized electricity market where the market operator is interested in satisfying consumer demand at the lowest cost. To do so, the operator asks the electricity generators to submit their supply functions, which give the quantity they are willing to deliver and the corresponding price. Player j's supply function is his action a_j, given by his strategy $\sigma_j(\theta_j)$. The mechanism designer aggregates the supply functions of the different producers and computes the outcome x, i.e., the price per unit to be paid to all generators whose supplies are needed. As alluded to before, this outcome depends on the players' actions, and there is no a priori certainty that the generators will report their true supply functions (actions). The role of the market operator is of course to design a mechanism that induces a truthful strategy and optimizes his objective, which, in this case is to have the lowest price.

Definition 7 *A principal's objective function $\mathcal{P} \in \mathcal{F}(\mathbb{D} \times \Delta(\Theta^n), \mathbb{R})$ is a function that, to a direct mechanism $\mathcal{D} \in \mathbb{D}$ and a PDF $f \in \Delta(\Theta^n)$, associates a real number $\mathcal{P}(\mathcal{D}, f)$.*

To illustrate, here are possible examples of the mechanism designer's objective function:

$$\mathcal{P}(\mathcal{D}, f) = \mathbb{E}_{\theta \rightsquigarrow f}[p(\mathcal{D}(\theta))], \tag{12}$$

$$\mathcal{P}(\mathcal{D}, f) = \mathbb{E}_{\theta \rightsquigarrow f}\Big[\sum_{j \in N} u_j(\theta_j, \mathcal{D}(\theta)) - \lambda \sqrt{Var\big\{u_j(\theta_j, \mathcal{D}(\theta))\big\}_{j \in N}}\Big], \tag{13}$$

$$\mathcal{P}(\mathcal{D}, f) = -\mathbb{E}_{\theta \rightsquigarrow f}\Big[Var\Big\{\mathbb{E}_{\theta'_{-j} \rightsquigarrow f_{-j}}[u_j(\theta_j, \mathcal{D}(\theta_j, \theta'_{-j}))]\Big\}_{j \in N}\Big], \tag{14}$$

In the first case, the principal wants to maximize the expected value of some function p of the outcome. In the second case, the principal wants to maximize the expected value of the sum of outcome utilities minus a weighted value of the standard deviation of the utility of the outcome. In the last case, the principal wants to minimize the expected value of the variance of the individual expected outcome utilities.

We now define what is meant by an ideal mechanism.

Definition 8 *A direct mechanism $\mathcal{D}^{ideal} \in \mathbb{D}$ is ideal, if it maximizes the principal's objective function, i.e.,$\mathcal{D}^{ideal} \in \arg\max_{\mathcal{D} \in \mathbb{D}} \mathcal{P}(\mathcal{D}, f)$. Let $\mathcal{P}^*_{ideal}(f) = \mathcal{P}(\mathcal{D}^{ideal}, f)$.*

Definition 9 *A mechanism $\mathcal{M}^* \in \mathbb{M}$ is BIC-optimal, if it maximizes the principal's objective function at a Bayesian-Nash equilibrium, i.e., $\mathcal{M}^* \in \arg\max_{\mathcal{M} \in \mathbb{M}} \mathcal{P}(\mathcal{M} \circ \sigma^{BN}, f)$, with $\sigma^{BN} \in \Sigma^{BN}(\mathcal{M}, f)$. Let $\mathcal{P}^*_{BIC}(f) = \mathcal{P}(\mathcal{M}^* \circ \sigma^{BN}, f)$.*

A corollary of Theorem 2, which states that revelation-direct mechanisms are BIC direct mechanisms, is the following result.

Theorem 3 *It holds that*

$$\mathcal{P}^*_{BIC}(f) = \sup_{\mathcal{D}_{BIC} \in \mathbb{D}_{BIC}(f)} \mathcal{P}(\mathcal{D}_{BIC}, f). \tag{15}$$

Proof. By definition, we have the following equality:

$$\mathcal{P}^*_{BIC}(f) = \sup_{\substack{\mathcal{M} \in \mathbb{M} \\ \sigma^{BN} \in \Sigma^{BN}(\mathcal{M}, f)}} \mathcal{P}(\mathcal{M} \circ \sigma^{BN}, f). \tag{16}$$

Yet, the set of direct-revelation mechanisms $\mathcal{M} \circ \sigma^{BN}$ we can obtain with $\mathcal{M} \in \mathbb{M}$ and $\sigma^{BN} \in \Sigma^{BN}(\mathcal{M}, f)$ is actually the entire set of direct-revelation mechanisms. According to Theorem 2, this set is also equal to the set of BIC direct mechanisms. This proves the result. $\square$

To recapitulate, the mechanism designer would ideally like to construct a direct mechanism that yields the value $\mathcal{P}^*_{ideal}(f)$. However, this construction requires the knowledge of the different players' types, which is not available in practice. The alternative is to proceed with the traditional construction of a revelation-direct mechanism, that is, the mechanism $\mathcal{D}_R = \mathcal{M} \circ \sigma^{BN}$, yielding $\mathcal{P}^*_{BIC}(f)$ as its BIC-optimal value. To compare the two results, we introduce the ideality gap which compares the two defined values, that is,

$$G(f) = \mathcal{P}^*_{ideal}(f) - \mathcal{P}^*_{BIC}(f) \geq 0.$$

The following theorem characterizes a condition under which this gap is zero.

Theorem 4 *There exists a mechanism $\mathcal{M}$ for which there exists a Bayesian-Nash equilibrium σ^{BN} such that the principal's objective at the equilibrium is $\mathcal{P}(\mathcal{M} \circ \sigma^{BN}, f) = \mathcal{P}^*_{BIC}(f)$ if and only if there exists a $\mathcal{D}^{ideal}_{BIC}$.*

Proof. If there exists a direct mechanism $\mathcal{D}^{ideal}_{BIC}$, which is ideal and BIC, then it is an ideal mechanism at the truthful strategy profile σ^{truth}, which is a Bayesian-Nash equilibrium. Therefore, the mechanism $\mathcal{D}^{ideal}_{BIC}$ and its Bayesian-Nash equilibrium are such that $\mathcal{P}(\mathcal{D}^{ideal}_{BIC} \circ \sigma^{truth}, f) = \mathcal{P}(\mathcal{D}^{ideal}_{BIC}, f) = \mathcal{P}^*_{\mathcal{D}}(f)$.

Reciprocally, suppose there exists a mechanism $\mathcal{M}$ and a Bayesian equilibrium σ^{BN} such that $\mathcal{P}(\mathcal{D}^{ideal}_{BIC} \circ \sigma^{truth}, f)$. Then direct-revelation mechanism $\mathcal{D}_R$ constructed with $\mathcal{M}$ at σ^{BN} has the same principal's objective value and, according Theorem 2, it is BIC; thus, $\mathcal{D}_R$ is a direct mechanism that is both ideal and BIC. $\square$

An implication of the above theorem is that if all ideal direct mechanisms $\mathcal{D}^{ideal}$ are not BIC, then no mechanism can achieve $\mathcal{P}(\mathcal{M} \circ \sigma^{BN}, f) = \mathcal{P}^*_{\mathcal{D}}(f)$. Indeed, in Section 4 we will provide an example where such a mechanism does not exist. In such cases, there is a good chance that the ideality gap $G(f)$ is positive, although it is still possible that a sequence of mechanisms and their Bayesian-Nash equilibria have a principal's objective value limit equal to $\mathcal{P}^*_{\mathcal{D}}(f)$, in which case the ideality gap would still be equal to zero.

Now, if the choice of the set of actions is part of the definition of the mechanism, then the following result would enable us to determine this set of actions.

Theorem 5 *If there exists a BIC-optimal mechanism $\mathcal{M}^*$, then there exists a BIC-optimal direct mechanism.*

Proof. A revelation-direct mechanism based on a BIC-optimal mechanism is a BIC direct mechanism with the same principal's Bayesian-Nash objective value and is therefore BIC-optimal too. $\square$

An important implication of this theorem is that, when searching for a BIC-optimal mechanism, we can restrict our research space to direct mechanisms.

4 Computation of Equilibria with the Return Function

Recalling that the set of best-reply strategies σ_j^{BR} for player j against strategies σ_{-j} with a PDF f_{-j}, and given a mechanism $\mathcal{M} \in \mathbb{M}$ is given by

$$\Sigma_j^{BR}(\sigma_{-j}, \mathcal{M}, f_{-j}) = \arg\max_{\sigma_j \in \Sigma} \ \mathbb{E}_{\theta_j \rightsquigarrow f_j} \left[u_j\left(\theta_j, \varphi_j\left(\sigma_j(\theta_j), \sigma_{-j}, \mathcal{M}, f_{-j}\right)\right) \right], \tag{17}$$

the objective of this section is to show how the Bayesian-Nash equilibrium can be computed using the return function. To do this, we will iteratively compute the return function using a learning process of the types. Our approach is similar to the idea of fictitious play, however with an important difference. Indeed, whereas in fictitious play, we determine a player's best reply to the other players' mixed strategies, here, we compute the best action for each player given his evaluation of how his actions affected the outcome in the previous iteration.

Note that in the rest of this section, we assume that the PDF f over types and the mechanism $\mathcal{M}$ are fixed. Therefore, the focus is on the determination of the Bayesian equilibrium of the game for given f and $\mathcal{M}$.

4.1 Computing the Return Function

Denote by $\{\sigma^i\}_{i \in \mathbb{N}}$ a sequence of strategies. This sequence converges toward a Bayesian equilibrium if

$$\forall i \in \mathbb{N}, \sigma^{i+1} \in \Sigma^{BR}(\sigma^i, \mathcal{M}, f), \tag{18}$$

with σ^0 randomly given. Instead of directly considering the players' strategies, our approach consists, at each iteration, in maximizing $u_j\left(\theta_j, \varphi_j^i\right)$ with respect to a_j, and next compute the strategies using $\sigma_j^i\left(\theta_j\right) = a_j^i$. In the sequel, we determine the next term $\varphi_j^{i+1}(a_j)$ using the following approximation, which is in the spirit of (18):

$$\varphi_j^{i+1}(a_j) \approx \varphi_j(a_j, \sigma_{-j}^i, \mathcal{M}, f_{-j}).$$

The following algorithm develops this idea, and in particular the learning aspect involved in the computations:

Algorithm 1 Optimization of the parameterized mechanism

while the convergence criterion is not verified **do**
  Generate a set of players with type θ^i according to PDF f.
  Compute the actions a_j^i maximizing the utility of the approximated return function $u_j^i\left(\theta_j, \varphi_j^i(a_j^i)\right)$.
  Given actions a^i, compute an outcome x^i, according to PDF $\mathcal{M}\left(a^i\right)$.
  Update the return functions φ_j^{i+1} given the actions a_j^i and the outcome x^i.
  Increment i.
end while

If the convergence is met, then the resulting Bayesian-Nash equilibrium is the best-reply strategy to the computed return function.[1] The first and the third steps of this algorithm are as difficult as the problem's inputs, i.e., the PDF f and the mechanism $\mathcal{M}$. In some problems, the mechanism may require solving a large optimization problem, which may be quite difficult of itself. That being said, it is the second step that is most demanding in most applications, as it involves n non-linear optimization problems with a potentially large action space. To get around some of these difficulties, we will use some local optimization tools (e.g., a gradient-based method), interpolation and perturbation schemes. Finally, the fourth step represents the learning process used by a player to update his return function. We now deal in detail with the steps of the above algorithm.

To empirically estimate the return function of player j, that is, $\varphi_j^{i+1}(a_j)$ for all $j \in N$, using the approximation

$$\varphi_j^{i+1}(a_j) \approx \varphi_j(a_j, \sigma_{-j}^i, \mathcal{M}, f_{-j}),$$

[1] It is worth mentioning that this algorithm can easily be parallelized, to speed up the learning process.

we need to generate actions a_j, $j \in N$. As an action is a consequence of θ_j, we randomly draw types using their PDFs. We call an observation for player j at iteration i, the couple

$$(a_j^i, x^i) \in A \times X.$$

Technically speaking, the return function φ_j for player $j \in N$ will be computed by a mapping container with keys $a \in A$. The values of the mapping container representing the return functions will be mapping containers with keys $x \in X$.

In our implementation, we will use the following smoothing procedure to compute $\varphi_j^{i+1}(a_j^i)$:

$$\varphi_j^{i+1}(a_j^i) = \mu_i x^i + (1-\mu_i)\varphi_j^i(a_j^i), \quad \mu_i \in (0,1)\,, \quad \forall j \in N, \tag{19}$$

that is, a weighted sum of the last observed outcome x^i and the previous estimation of the return function $\varphi_j^i(a_j^i)$. The sequence can be selected using the following result:

Theorem 6 *Suppose that, for any type $\theta_j \in \Theta$ and any action a_j,$j \in N$, the utility function $u_j(\theta_j, .)$ is bounded and that the sequence $u_j(\theta_j, x^i)$ for $a_j^i = a_j$ converges, i.e.,*

$$\exists u_j^\infty(a_j) \in \mathbb{R}_+, u_j^\infty(a_j) = \lim_{i \to +\infty} \mathbb{E}_{x^i | a_j^i = a_j}[u_j(\theta_j, x^i)]. \tag{20}$$

Also, assume that types are bounded and that $\sum \mu_i$ diverges but that $\sum \mu_i^2$ converges. Then, $u_j(\theta_j, \varphi_j^i(a_j))$ converges in probability to u_j^∞.

Proof. See Appendix. □

The sequence $\mu_i = 2/(i+1)$ satisfies the requirements in the above theorem, and will be used in the sequel. One intuitively appealing property of this sequence is that $\mu_i > (1-\mu_i)\mu_{i-1}$, which means that the last observation is more weighted than each of the previous ones.

4.2 Interpolation

The accuracy of the estimation of the return function depends on the number of observed actions. Since the set of actions may be very large, or even infinite, and thus not all possible actions could be considered at each iteration, we use an interpolation scheme to estimate φ_j. We suppose from now on that the return function φ_j is continuous in action a_j. This is a realistic hypothesis, as the PDF over others' types obtained through $\theta_{-j} \rightsquigarrow f_{-j}$ will smooth (through averaging) the return function. Denote by $d_A(a_j, a_j'^i)$ a distance between the actions a_j and a_j', and by $w(d_A(a_j, a_j'^i))$ a weight assigned to this distance, which is decreasing in $d_A(a_j, a_j'^i)$.[2]

Let

$$B_A^i(a_j, \alpha) = \{a_j' \in A : | : d_A(a_j, a_j'^i) \le \alpha\},$$

be the ball of center a_j and radius α, where α is an arbitrary positive number. Given $\theta_j \in \Theta$, the interpolated value for player j when he chooses action a_j is estimated by

$$\bar{u}_j(\theta_j, \varphi_j^i(a_j)) = \frac{\sum\limits_{a_j'^i \in B_A^i(a_j,\alpha)} u_j(\theta_j, \varphi_j^i(a_j'))w(d_A(a_j, a_j'))}{\sum\limits_{a_j'^i \in B_A^i(a_j,\alpha)} w(d_A(a_j, a_j'))}. \tag{21}$$

Remark 6 *To avoid being stuck at a solution that is not optimal, we perturb the solution (vector of actions) obtained at the last iteration, and re-feed the approximation in (19). In our case, we will simply add a random number drawn from a given interval to each element of the last action vector. As observations pile up, this interval is narrowed.*

[2] To illustrate, in Section 4, we will take $w(d\,(a,a')) = 1/(1+5d\,(a,a'))^2$.

4.3 Parametrized Mechanisms

Recall that the set of mechanisms is given by $\mathbb{M}$, and up to now, we have not imposed any restrictions on this set. In practice, the mechanism designer will consider, for obvious tractability reasons, a "manageable" set of possibilities. We will assume from now on that the set $\mathbb{M}$ contains elements of the form $\mathcal{M}^t$, where $t, t \in T$ is a given parameter that can be adjusted (or optimized) by the mechanism designer. For instance, in our example in the next section, t represents a couple of real numbers, and hence, $T = \mathbb{R}_+ \times \mathbb{R}_+$. Following this parametrization, the return functions and the Bayesian-Nash equilibria will also depend on t. Then, the mechanism designer solves the following optimization problem:

$$\max_{t \in T} \mathcal{P}(\mathcal{M}^t \circ \sigma_t^{BN}, f), \tag{22}$$

$$\text{subject to} \quad : \quad \sigma_t^{BN} \in \Sigma^{BN}(\mathcal{M}^t, f), \tag{23}$$

where σ_t^{BN} is a Bayesian-Nash equilibrium for the mechanism $\mathcal{M}^t$.

To solve the above optimization problem, we propose the following algorithm:

Algorithm 2 Optimization of the parameterized mechanism

while the convergence criterion is not verified **do**
- Generate a set of players with type θ^i according to PDF f.
- Compute the actions a_j^i maximizing the utility of the approximated return function $u_j^i\left(\theta_j, \varphi_j^i(a_j^i)\right)$.
- Given action a_j^i, compute an outcome x^i according to the PDF $\mathcal{M}^{t^i}\left(a^i\right)$.
- Update the return functions φ_j^{i+1} given the actions a_j^i and the outcome x^i.
- Compute the principal's objective function for the last iterations and find a better parameter t^{i+1}.
- Increment i.

end while

5 Illustrative Example: A Cake-Cutting Problem

To illustrate the theory developed above, we provide an example of a cake-cutting problem, which is of great use in operations research, as it allows the modelling of an assignment problem that includes the preferences of the agents involved. For instance, some shift-scheduling and matching problems exactly fit the cake-cutting problem (CCP) formalism. This problem can be stated as follows: given a cake and a set of players having additive utility functions over the subsets of the cake, the problem is how to allocate the cake to optimize a certain objective, while satisfying some constraints, such as fairness. This class of problems has been the subject of numerous papers; see, e.g., Brams and Taylor (1995), Brams and Taylor (1996), Robertson and Webb (1998), and recent papers, e.g., Mossel and Tamuz (2010) and Chen et al. (2010).

5.1 The Model

To simplify the computations, while still being able to illustrate our approach, we suppose that the cake is initially cut into a set K of homogenous portions. An outcome is a matrix

$$x = \{x_{jk}\} \in [0,1]^{N \times K},$$

where x_{jk} is the portion $k \in K$ allocated to player $j \in N$. Note that the set X of admissible outcomes (allocations) is given by

$$X = \{x \in [0,1]^{N \times K} | \forall k \in K, \sum_{j \in N} x_{jk} \leq 1\}.$$

Denote by $\theta_{jk} \geq 0$ the utility of player $j \in N$ for portion $k \in K$. Thus, the type of player j is the vector θ_j, and his utility function over any subset x_j of the cake is $u_j\left(\theta_j, x_j\right) = \sum_{k \in K} \theta_{jk} x_{jk}$. We normalize the utility function of each player for the whole cake to one. Therefore, if $1_K = \{1\}_{k\in K}$ is the allocation of the cake, then $u_j(\theta_j, 1_K) = \sum_{k\in K} \theta_{jk} = 1$. As a result, the set Θ of types is the polytope $\Theta = \{\theta_j \in \mathbb{R}^K_+ : | : \sum_{k\in K} \theta_{jk} = 1\}$.

One approach to allocating the cake would be to solve the following linear optimization problem:

$$\begin{aligned} &\max_x \; u \\ &\text{subject to} : \sum_{k\in K} a_{jk}x_{jk} = u, \quad \forall j \in N, \\ &\qquad x \in X. \end{aligned} \tag{24}$$

We will refer to this approach as mechanism $\mathcal{M}^0$, that is, the mechanism that associates an outcome x to a vector a of actions. We define an admissible action for player j to be one that satisfies the constraints $a_{jk} \geq 0, k \in K$, and $\sum_{k\in K} a_{jk} = 1$. The normalization of the admissibility constraints on the types and actions has two implications, namely: (i) the set of actions and types are the same, i.e, $A = \Theta$; and (ii) the mechanism $\mathcal{M}^0$ is a direct mechanism. Note that if the above optimization problem has multiple solutions, then we will simply choose one of them randomly.

An example of a PDF f of types is obtained through the following process: for each player $j \in N$, we randomly generate a vector in $[0, 1]^K$ according to the uniform distribution. Then, we normalize this vector by dividing each component by the sum of all components, hence obtaining a vector $\theta_j \in \Theta$. Note that this mechanism $\mathcal{M}^0$ is symmetric in the sense of (2), and the PDFs are symmetric too, due to our way of generating the vector $\theta_j, j \in N$. A consequence of these symmetries is that the return function φ_j is independent of j, and therefore we have $\varphi_j = \varphi$ for all $j \in N$.

5.2 A Simple Two-Player Example

In principle, the above problem can be solved for any number of players and any set K. However, for the sake of clarity, we focus here on the simplest possible setting of two players, and $|K| = 2$. Let $a_1 = (a_{11}, a_{12})$ and $a_2 = (a_{21}, a_{22})$. In this context, problem (24) becomes

$$\max \; u \tag{25}$$
$$\text{subject to} : u = a_{11}x_{11} + (1 - a_{11})x_{12}, \tag{26}$$
$$u = a_{21}(1 - x_{11}) + (1 - a_{21})(1 - x_{12}), \tag{27}$$
$$x_{11}, x_{12} \geq 0, \tag{28}$$

where x_{jk} represents the portion $k = 1, 2$ allocated to player $j = 1, 2$. The description of the type θ_j and the action a_j of each player j can now be reduced to one real variable each in $[0, 1]$ that is, θ_{j1} and a_{j1}. As a result, a strategy σ_j is a mapping from $[0, 1]$ into $[0, 1]$.

To simplify the theoretical analysis, we assume that the PDF f of types is obtained by drawing uniformly randomly θ_{j1} in $[0, 1]$, hence obtaining $\theta_{j2} = 1 - \theta_{j1}$.

Using the algorithm defined in (1), we compute a Bayesian-Nash equilibrium. In Figure 2, the dots represent the equilibrium strategy σ_1^{BN} of player 1, and the the line corresponds to the truthful strategy σ_1^{truth}. As the mechanism is symmetric, the equilibrium strategy of player 2 is the same. As we can see on this figure, the equilibrium strategy σ_j^{BN} consists of overvaluing the portion he desires less, and undervaluing the portion he prefers.

Also, in this simple example of two players and two attributes, we can draw the mechanism as done in Figure 3.

A linear approximation of the strategy in Figure 2 consists of choosing the action a_{11} such that

$$a_{11} = \sigma^{BN}(\theta_{11}) = a^*_{min} + (1 - 2a^*_{min})\theta_{11}.$$

The numerical value of a^*_{min} is approximately 0.3.

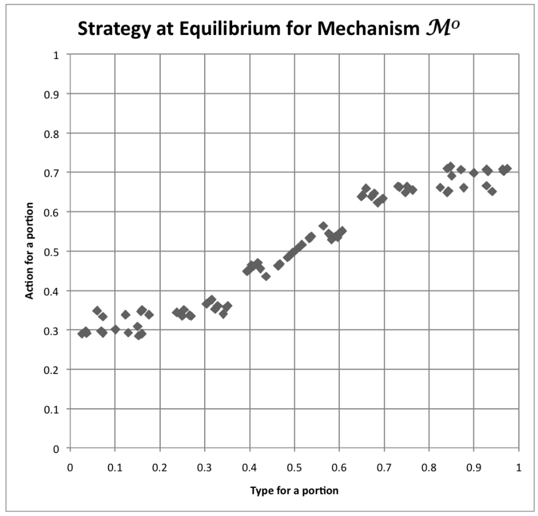

Figure 2: Bayesian-Nash Equilibrium of mechanism $\mathcal{M}^0$

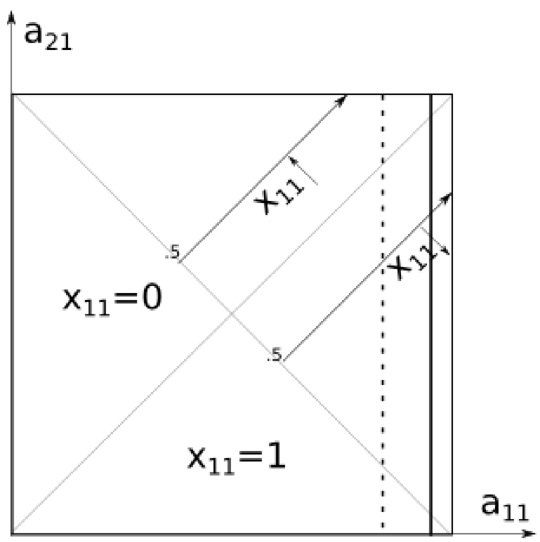

Figure 3: Mechanism of the cake-cutting problem with 2 attributes and 2 players

Now, supposing that player 2 chooses to play the computed Bayesian-Nash equilibrium, we can evaluate the return function for player 1, and write his payoff if he chooses action a_{11} and has a type θ_1:

$$u_1(\theta_1, \varphi(a_1, \sigma, \mathcal{M}, f)) = \int_{u_2} u_1(\theta_{11}, \mathcal{M}(a_1, \sigma(\theta_2)))f(\theta_2)d\theta_2, \tag{29}$$

$$= \int_{a_2=a^*_{min}}^{1-a^*_{min}} \frac{u_1(\theta_{11}, \mathcal{M}(a_1, a_2))}{1-2a^*_{min}} da_2. \tag{30}$$

If $a_{11} \leq 1/2$, we can separate the integral into the cases where $a_{21} \leq a_{11}$, $a_{11} \leq a_{21} \leq 1 - a_{11}$ and $a_{21} \geq 1 - a_{11}$. The three resulting terms are

$$\frac{1}{1-2a^*_{min}} \left[\int_{a^*_{min}}^{\theta_{11}} \left[\theta_{11} + (1-\theta_{11})(1 - \frac{1}{2-a_{11}-a_{21}})\right] da_{21} \right], \tag{31}$$

$$\frac{1}{1-2a^*_{min}} \left[\int_{a_{11}}^{1-a_{11}} \frac{1-\theta_{11}}{2-a_{11}-a_{21}} da_{21} \right], \tag{32}$$

$$\frac{1}{1-2a^*_{min}}\left[\int_{1-a_{11}}^{1-a^*_{min}}\Big[\theta_{11}(1-\frac{1}{a_{11}+a_{21}})+(1-\theta_{11})\Big]da_{21}\right]. \tag{33}$$

Calculating these integrals yields

$$(1-2a^*_{min})u_1(\theta_{11},\varphi(a_1)) = 2(a_{11}-a^*_{min}) + \ln\frac{4(1-a_{11})^2}{2-a_{11}-a^*_{min}} + \theta_{11}\ln\frac{2-a_{11}-a^*_{min}}{4(1-a_{11})^2(1+a_{11}-a^*_{min})}. \tag{34}$$

Therefore, level curves, for which $v_1(\varphi(a_1))$ is constant, are described by the following equation:

$$\theta_{11} = \frac{k-2a_{11}-\ln\frac{4(1-a_{11})^2}{2-a_{11}-a^*_{min}}}{\ln\frac{2-a_{11}-a^*_{min}}{4(1-a_{11})^2(1+a_{11}-a^*_{min})}}, \qquad \text{for } a_{11}\le 1/2 \tag{35}$$

with $k=(1-2a^*_{min})u_1(\theta_{11},\varphi(a_1))+2a^*_{min}$. Therefore, we get $u_1(\theta_{11},\varphi(a_1))=\frac{k-2a^*_{min}}{1-2a^*_{min}}$. The level curves for expected utility $u_1(\theta_{11},\varphi(a_1))$, as functions of θ_{11} and a_{11}, at the Bayesian-Nash equilibrium are shown in Figure 4, where the best-reply strategy is drawn in red.

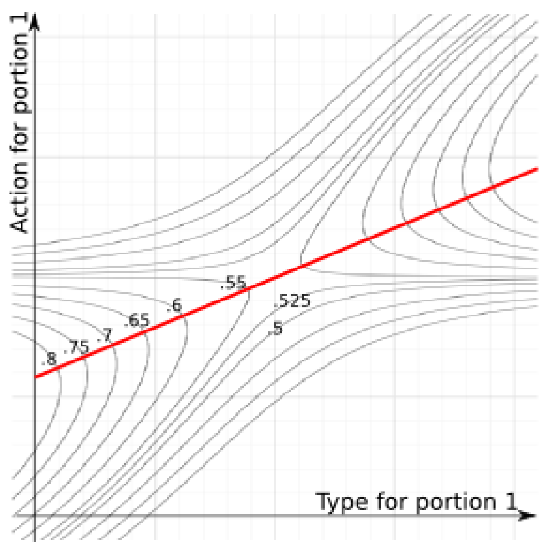

Figure 4: Level curves of expected utilities of a player at a Bayesian-Nash equilibrium

The best-reply action of player 1 is the action that maximizes the expected value of $u_1(\theta_{11},\varphi(a_1))$, given a value of θ_{11}. Therefore, if $\theta_{11}=0.2$ for instance, the largest value he can obtain for $u_1(\theta_{11},\varphi(a_1))$ is 0.75, which is reached by choosing action $a_{11}=0.35$. As we can easily see, the best-reply strategy does not coincide with the computed Bayesian-Nash equilibrium, but is very close to it. This shows that the use of the return-function method is efficient for computing the Bayesian-Nash equilibrium.

5.3 The Principal's Objective Function

To pursue our illustration, let us assume that the mechanism designer is interested by some fairness considerations when allocating the cake to the two players. This idea can be captured by defining the principal's objective function $\mathcal{P}(\mathcal{D},f)$ as follows (here $n=2$):

$$\mathcal{P}(\mathcal{D},f) = \lambda\mathbb{E}_{\theta\rightsquigarrow f}\Big[\frac{1}{n}\sum_{j\in N}u_j(\theta_j,\mathcal{D}(\theta))\Big] - (1-\lambda)\,\mathbb{E}_{\theta\rightsquigarrow f}\Big[\sqrt{:Var:\big(\{u_j(\theta_j,\mathcal{D}(\theta))\}_{j\in N}\big)}\Big], \tag{36}$$

$$= \mathbb{E}_{(\theta_1,\theta_2)\rightsquigarrow f}\Big[\lambda\frac{u_1(\theta_1,D(\theta))+u_2(\theta_2,D(\theta))}{2} - (1-\lambda)\,|u_1(\theta_1,D(\theta))-u_2(\theta_2,D(\theta))|\Big], \tag{37}$$

that is, a weighted sum of the average utilities and the standard deviation, with $\lambda \in [0,1]$. In the sequel, we use $\lambda = 1/2$.

Now, using the return-function method, we can compute the Bayesian-Nash equilibrium σ_0^{BN}. The value of the objective function of the principal is then given by[3]

$$\mathcal{P}(\mathcal{M}^0 \circ \sigma^{BN}, f) \approx 0.543. \tag{38}$$

Note that this is also the value that we would get by cutting each of the two pieces of cake into two subpieces and giving each of these subpieces to one player. Indeed, each player would get a utility of 0.5, and the standard deviation would be equal to zero.

Although the choice of mechanism $\mathcal{M}^0$ is intuitive, there is no reason to believe that it is a BIC-optimal mechanism. (Indeed, we will show later on that $\mathcal{M}^0$ is not BIC-optimal.) However, in this particular setting of $\lambda = 1/2$, we have the following interesting result.

Proposition 1 *The direct mechanism $\mathcal{M}^0$ is an ideal direct mechanism.*

Proof. The perfect direct mechanism could easily be obtained by choosing a solution of the following linear program:

$$\max_x u_1/2 + u_2/2 - \delta \tag{39}$$

$$\text{subject to :} u_1 = \theta_{11}x_{11} + (1-\theta_{11})x_{12} \tag{40}$$

$$u_2 = \theta_{21}(1-x_{11}) + (1-\theta_{21})(1-x_{12}) \tag{41}$$

$$\delta \geq u_1 - u_2 \tag{42}$$

$$\delta \geq u_2 - u_1. \tag{43}$$

If $u_1 = u_2$ at the optimum, then the point (x_1, x_2) is feasible in the program (25). Thus, as the objective functions are equivalent, and there are no additional constraints, the allocation found is the same as the one we would have found using program (25).

Suppose now that $u_1 > u_2$ at the optimum. Then, $\delta = u_1 - u_2$. Thus, the objective function can be written $3u_2/2 - u_1/2$. This is obviously a strictly decreasing function in both x_1 and x_2. This leads to a contradiction in the open set $u_1 > u_2$. We have a similar result when assuming $u_1 < u_2$.

Therefore, the optimum is always the same as the one found with program (25), which means that the two direct mechanisms are equivalent. In particular, $\mathcal{M}^0$ is an ideal mechanism. □

We can use this ideal direct mechanism $\mathcal{M}^0$ to compute the ideal value, and obtain

$$\mathcal{P}\left(\mathcal{M}^0, f\right) = \mathcal{P}^*_{ideal}(f) = 0.611. \tag{44}$$

As we already found that the two players do not behave truthfully in equilibrium, we note that the mechanism $\mathcal{M}^0$ is not ideal at equilibrium. Therefore, what remains to be done is to find a BIC-optimal mechanism. This issue is tackled in the next subsection.

5.4 Mechanism Optimization

To start, we state the following "impossibility" results.

Proposition 2 *For our cake-cutting problem, there does not exist a mechanism $\mathcal{M}$, such that $\mathcal{P}(\mathcal{M} \circ \sigma^{BN}, f) = \mathcal{P}_{\mathcal{D}}(f)$, where σ^{BN} is a Bayesian-Nash equilibrium of $\mathcal{M}$.*

[3]Observe that, from Figure 4, we see that the average of the utilities is a about 0.68, when the utilities vary almost uniformly between 0.54 and 0.84. Therefore, the standard deviation is about 0.15, which gives us a principal's Bayesian-Nash objective value of $0.68 - 0.15 \approx 0.54$, which is very close to the value of 0.51 obtained above by our numerical simulations.

Proof. An ideal direct mechanism provides the following solution:

1. If the players have different types, then any ideal direct mechanism will provide a unique solution, i.e., the solution (allocation) given by the linear program (39).
2. If the players have the same type θ_j, then the ideal direct mechanisms may give different allocations to the players, but all having the same utility of 0.5 for each player. The reason is that the average of their utilities is necessarily 0.5 because the sum of the utilities is $u_1(\theta_1, x_1) + u_2(\theta_2, x_2) = u_1(\theta_1, x_1 + x_2) = u_1(\theta_1, 1_K) = 1$ and the standard deviation is minimized when both players receive 0.5.

To complete the proof, it suffices to observe that none of these ideal direct mechanisms is BIC, as players have incentives to overbid the parts they do not want (see Figure 2). Therefore, by applying Theorem 4, the proposition is proved. □

Note that the use of "our" in the above theorem refers to our definition of the objective of the principal, $\mathcal{P}(\mathcal{D}, f)$, and the definition of the PDFs.

Conjecture 1 *The ideality gap $G(f)$ is positive.*

Now, the problem with mechanism $\mathcal{M}^0$ is that each player has an incentive to overvalue (undervalue) the portions of the cake he prefers less (more). This can easily be seen from Figure 2, which shows the strategy of player 1; a similar observation can be made for player 2. The explanation is that, as long as player 1 (similarly for player 2) claims his utility of his preferred share is higher than the other player's utility of that same share, he will appear to be less satisfied with the received outcome than he actually is. This is due to the discontinuity of the mechanism at points $a_{11} = a_{21}$.

Let us consider Figure 3 to explain this in more detail. Suppose that player 1 really prefers portion 1. Then he will be interested in areas where x_{11} is high, that is, mainly the bottom. Then, the upper area is very good too (with values of x_{11} in $[1/2, 1]$). The right area is relatively bad (with values of x_{11} in $[0, 1/2]$). If the other player is truthful, he will choose $a_{21} = \theta_{21}$, which is distributed all over $[0, 1]$, but has more values in the middle area. Obviously, player 1, who prefers portion 1, should claim $a_{11} > 1/2$ to avoid the left area and have the right area instead.

If he chooses to be truthful, then he will obtain the vertical solid line on the right of the following figure. But if he plays the dotted vertical line on the left, then he will almost always get better. Thus, he will not be truthful and will play the dotted line.

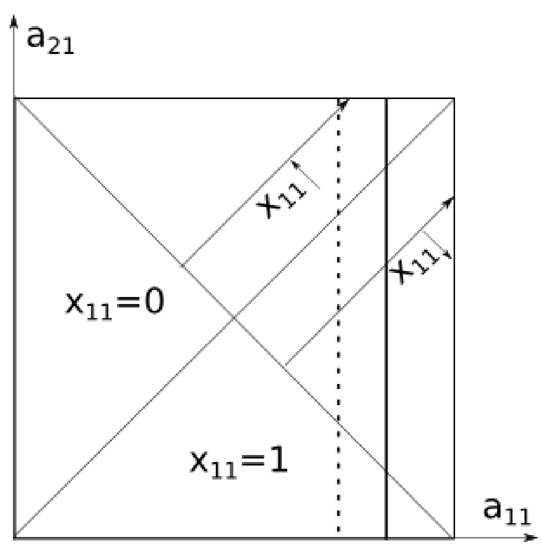

Figure 5: Outcome of the mechanism

Part of the problem comes from the fact that, when we are in the right side, since $x_{12} = 0$, player 1's utility only come from portion 1. But since he has undervalued his utility for portion 1, the linear program considers that he is more disappointed than he actually is. Thus, player 1 is taking advantage of the fact that the allocations of each portion are extreme, that is, often equal to 0 or 1.

To get around this problem, we modify the mechanism to include the following relationship:

$$\frac{x_{j1}}{x_{j2}} = h\left(\frac{a_{j1}}{a_{j2}}\right), \quad j = 1, 2, \tag{45}$$

where $h(\cdot)$ is increasing in its argument. To be more specific, we first impose the following two properties on h, namely:

$$h(1) = 1, \quad h\left(\frac{1}{y}\right) = \frac{1}{h(y)},$$

that is, if $a_{j1} = a_{j2}$, then $x_{j1} = x_{j2}$, and permuting a_{j1} and a_{j2}, implies that x_{j1} and x_{j2} must be also permuted. Next, we specify the function $h(\cdot)$ as follows:

$$h(y) = y^{t_1},$$

where t_1 is a positive real number. The relationship in (45), then becomes

$$x_{j1}(1 - a_{j1})^{t_1} = x_{j2}a_{j1}^{t_1}, \quad j = 1, 2.$$

Since directly adding the above equations as constraints to linear program (39) may lead to an infeasible program, we proceed differently and add them to the objective function in the form of penalty terms. We denote by

$$t_2\gamma_j = t_2|x_{j1}(1 - a_{j1})^{t_1} - x_{j2}a_{j1}^{t_1}|, \quad j = 1, 2,$$

these penalty terms, where t_2 is a weight to be optimized.

Given these modifications, we can now define a parametrized mechanism $\mathcal{M}^t$, where $t = (t_1, t_2)$, defined by the solution of the following linear program:

$$\begin{aligned}
&\max_x u_1/2 + u_2/2 - \delta - t_2\gamma_1 - t_2\gamma_2 \qquad (46)\\
\text{subject to :}\, & u_1 = a_{11}x_{11} + (1 - a_{11})x_{12}\\
& u_2 = a_{21}(1 - x_{11}) + (1 - a_{21})(1 - x_{12})\\
& \delta \geq u_1 - u_2\\
& \delta \geq u_2 - u_1\\
& \gamma_1 \geq x_{11}(1 - a_{11})^t - x_{12}a_{11}^{t_1}\\
& \gamma_1 \geq x_{12}a_{11}^t - x_{11}(1 - a_{11})^{t_1}\\
& \gamma_2 \geq x_{11}(1 - a_{21})^t - x_{12}a_{21}^{t_1}\\
& \gamma_2 \geq x_{12}a_{21}^t - x_{11}(1 - a_{21})^{t_1}.
\end{aligned}$$

Note that if we let t_1 go to zero, then we get $x_{j1} = x_{j2}, j = 1, 2$. If t_1 tends to $+\infty$, then $\gamma_j = 0$, $j = 1, 2$, and we are back to the linear program in (39).

Applying Algorithm (2), we obtain $t_1^* \approx 2$ and $t_2^* \approx 1.5$. Denote by $\sigma_{t^*}^{BN}$ the corresponding Bayesian-Nash equilibrium $\sigma_{t^*}^{BN}$. The principal's objective value of mechanism $\mathcal{M}^{t^*}$ at this equilibrium is given by

$$\mathcal{P}(\mathcal{M}^{t^*} \circ \sigma_{t^*}^{BN}, f) \approx 0.564.$$

To assess this value, we can compare it to the other two previously computed values, namely,

$$\begin{aligned}
\mathcal{P}(\mathcal{M}^0 \circ \sigma^{BN}, f) &\approx 0.543,\\
\mathcal{P}_{ideal}^*(f) &= 0.611.
\end{aligned}$$

We note that $\mathcal{P}(\mathcal{M}^{t^*} \circ \sigma_{t^*}^{BN}, f)$ is somehow in the middle between the value achieved with mechanism $\mathcal{M}^0$ and the ideal value, which would have been obtained if the principal had known the two players' types. The improvement in the principal's objective can be measured by

$$\frac{\mathcal{P}(\mathcal{M}^{t^*} \circ \sigma_{t^*}^{BN}, f)}{\mathcal{P}(\mathcal{M}^0 \circ \sigma^{BN}, f)} \approx 1.04.$$

We have computed the Bayesian-Nash strategies at equilibrium in Figure 6.

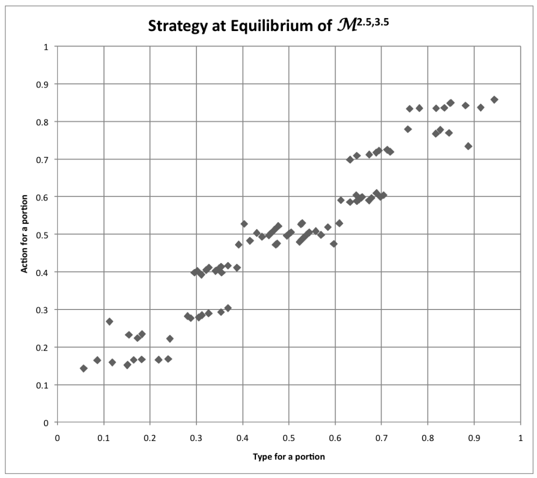

Figure 6: Bayesian-Nash equilibrium for the computed optimal mechanism

As we see in this figure, the Bayesian-Nash equilibrium is closer to the truthful strategy profile than for the mechanism M^0.

6 Conclusion

In this paper, we introduced a new approach: the return function φ, for the computation of Bayesian-Nash equilibria in the context of mechanism design. The conceptual developments shown here were triggered by a problem of shift scheduling, where management is willing to account for some employee preferences when allocating them to different shifts. The employees will be asked to submit their preferences and a system will produce an allocation that also takes into account other elements, such as past allocation, seniority and fairness. The numerical example presented here was meant to illustrate what can be done in this area. Needless to say, a lot of development is still required before achieving a truly operational large-scale system. For instance, the way we estimated the return function, especially its updating and its interpolation, could definitely be improved by taking into account some specific features of the actual problem. Also, the computation of the players' best strategies is done by maximizing the utility of the return function. We used a basic local optimizing approach, with the truthful action as the initial point. This can also be improved, with, for instance, a sensitivity analysis.

Appendix: Proof of Theorem 6

Let $\epsilon > 0$. We denote $\alpha_{ki} = \Big[\prod_{r=k+1}^{i}(1-\mu_r)\Big]\mu_k$ the weight of the observation made at iteration k for the return function at iteration i, $u_n^i = u_n(\varphi_n^i(a_n))$, $u_n^i = \mathbb{E}_{x_n^i}[u_n(x_n^i)]$ and $u_n^\infty = u_n^\infty(a_n)$. Then, we need to show that $\lim \mathbb{P}(|u_n^i - u_n^\infty| \geq \epsilon) = 0$. We know that

$$u_n^i = \sum_{k=0}^{i} \alpha_{ki} u_n^k \quad \text{and} \quad \sum_{k=0}^{i} \alpha_{ki} = 1 \tag{47}$$

First, we will prove that $\lim \prod_{r=0}^{i}(1-\mu_r) = 0$. As a matter of fact, if $\sum \mu_i$ diverges (necessarily to $+\infty$ since μ_i is positive), then $\sum \ln(1-\mu_i)$ diverges to $-\infty$. Thus, $\lim \prod_{r=0}^{i}(1-\mu_r) = \lim \exp(\sum_{r=0}^{i}\ln(1-\mu_r)) = 0$. As a result, for any integer I, $\lim \sum_{k=0}^{I-1}\alpha_{ki} = \lim \prod_{r=I}^{i}(1-\mu_r)\underbrace{\sum_{k=0}^{I-1}\alpha_{k,I-1}}_{=1} = 0$.

Let us now prove that $\lim \mathbb{E}[u_n^i] = u_n^\infty$. Note that we have the following inequality:

$$|\mathbb{E}[u_n^i] - u_n^\infty| \leq \sum_{k=0}^{i}\alpha_{ki}|\mathbb{E}[u_n^k] - u_n^\infty| \tag{48}$$

Since we know that $\lim \mathbb{E}[u_n^i] = u_n^\infty$, we know there exists an integer I_1 so that for any $i \geq I_1$, we have $|\mathbb{E}[u_n^i] - u_n^\infty| \leq \epsilon/4$. Since $\sum_{k=I_1}^{i}\alpha_{ki} \leq 1$, we now have:

$$|\mathbb{E}[u_n^i] - u_n^\infty| \leq \sum_{k=0}^{I_1-1}\alpha_{ki}|\mathbb{E}[u_n^k] - u_n^\infty| + \epsilon/2 \tag{49}$$

Moreover, since utility functions are bounded by a real number M, we know that $|\mathbb{E}[u_n^k]-u_n^\infty| \leq 2M$. And, as $\lim \sum_{k=0}^{I_1-1}\alpha_{ki} = 0$, there exists a integer $I_2 \geq I_1$ for which, whenever $i \geq I_2$, we have $|\sum_{k=0}^{I_1-1}\alpha_{ki}| \leq \epsilon/(4M)$. Therefore, we have $\forall i \geq I_2, |\mathbb{E}[u_n^i] - u_n^\infty| \leq \epsilon$, which means that $\lim \mathbb{E}[u_n^i] - u_n^\infty = 0$.

We also have $|u_n^i - \mathbb{E}[u_n^i]| \leq |u_n^i - u_n^\infty| + |u_n^\infty - \mathbb{E}[u_n^i]|$. Consequently, when $i \geq I_2$, if $|u_n^i - u_n^\infty| \leq \epsilon$, then $|u_n^i - \mathbb{E}[u_n^i]| \leq 2\epsilon$. Therefore,

$$\forall i \geq I_2, \mathbb{P}(|u_n^i - u_n^\infty| \geq \epsilon) \leq \mathbb{P}(|u_n^i - \mathbb{E}[u_n^i]| \geq 2\epsilon) \tag{50}$$

We can now use the Bienaymé-Tchebychev inequality to say that $\mathbb{P}(|u_n^i - \mathbb{E}[u_n^i]| \geq 2\epsilon) \leq \frac{Var(u_n^i)}{4\epsilon^2}$. Therefore, the last thing we need to do is to prove that $\lim Var(u_n^i) = 0$.

In order to prove that, let $\delta > 0$. Since $Var(u_n^i)$ is positive, we will only need to show that for a high enough n, the variance is less than δ. Notice that, since utility functions are bounded by M, their variances are bounded by M^2.

$$Var(u_n^i) = \sum_{k=0}^{i}\alpha_{ki}^2 Var(u_n^i) \leq M^2\sum_{k=0}^{i}\alpha_{ki}^2 \tag{51}$$

Remark that $\alpha_{ki} = \Big[\prod_{r=k+1}^{i}(1-\mu_r)\Big]\mu_k \leq \mu_k$. Yet, $\sum \mu_i^2$ converges, which means that there exists an integer I_3 so that, for any $i \geq I_3$, $\sum_{k=i}^{+\infty}\mu_k^2 \leq \delta/(2M^2)$. Therefore, for any $i \geq I_3$, we have $\sum_{k=I_3}^{i}\alpha_{ki} \leq \delta/(2M^2)$. We now have:

$$\forall i \geq I_3, Var(u_n^i) \leq M^2\sum_{k=0}^{I_3-1}\alpha_{ki}^2 + M^2\sum_{k=I_3}^{i}\alpha_{ki}^2 \tag{52}$$

Yet, $\forall i \geq I_3$, $\sum_{k=0}^{I_3-1}\alpha_{ki}^2 \leq \Big(\prod_{r=I_3}^{i}(1-\mu_r)\Big)^2$, whose limits is 0. Therefore, there is an integer $I_4 \geq I_3$, so that for any $i \geq I_3$, $\Big(\prod_{r=I_3}^{i}(1-\mu_r)\Big)^2 \leq \delta/(2M^2)$. From that we deduce that $\forall i \geq I_4, Var(u_n^i) \leq \delta$, which proves that $\lim Var(u_n^i) = 0$, and that u_n^i converges in probability to u_n^∞.

References

A. Abdulkadirolu and T. Sönmez. School choice: A mechanism design approach. *American Economic Review*, 93: 729–747, 2003.

T.G. Bourbeau, B. Crainic, M. Gendreau, and J. Robert. Design for optimized multi-lateral multi-commodity markets. *European Journal of Operational Research*, 163:503–529, 2005.

J.B. Brams and A.D. Taylor. An envy-free cake division protocol. *The American Mathematical Monthly*, 102:9–12, 1995.

J.B. Brams and A.D. Taylor. *Fair Division - From cake-cutting to dispute resolution*. Cambridge University Press, 1996.

J.-M. Chen and H.-L. Cheng. Effect of the price-dependent revenue-sharing mechanism in a decentralized supply chain. *Central European Journal of Operations Research*, 20:299–317, 2012.

Y. Chen, J.K. Lai, D.C. Parkes, and A.D. Procaccia. Truth, justice, and cake cutting. *Proceedings of the 24th AAAI Conference on Artificial Intelligence (to appear)*, 2010.

P. Dasgupta, P. Hammond, and E. Maskin. The implementation of social shoice rules: some results on incentive compatibility. *Review of Economic Studies*, 46:185–216, 1979.

N.R. Devenur and T.P. Hayes. The adwords problem: Online keyword matching with budgeted bidders under random permutations. In *Proceedings of the 19th ACM conference on Electronic commerce*, pages 71–78, 2009.

A. Gibbard. Manipulation of voting schemes: a general result. *Econometrica*, 41:587–601, 1973.

B. Holmstrom. *On incentives and control in organizations*. PhD thesis, Stanford University, 1977.

S. Jain and S. Raghavan. A queuing approach for inventory planning with batch ordering in multi-echelon supply chains. *Central European Journal of Operations Research*, 17:95–110, 2009.

A.M. Manelli and D.R. Vincent. Multidimensional mechanism design: Revenue maximization and the multiple-good monopoly. *Central European Journal of Operations Research*, 137:153–185, 2007.

M. Mes, M. van der Heijdenet, and P. Shurr. Interaction between intelligent agent strategies for real-time transportation planning. *Central European Journal of Operations Research*, 2011.

E. Mossel and O. Tamuz. Truthful fair division. In *Proceedings of the 3rd International Symposium on Algorithmic Game Theory*, pages 288–299, 2010.

R. Myerson. Incentive-compatibility and the bargaining problem. *Econometrica*, 47:61–73, 1979.

P.A. Pathak. The mechanism design approach to student assignment. *Annual Review of Economics*, 3:513–536, 2011.

J. Robertson and W. Webb. *Cake-Cutting Algorithms: Be Fair If You Can*. AK Peters Ltd, 1998.

X. Su and S. Zenios. Recipient choice can address the efficiency-equity trade-off in kidney transplantation: A mechanism design model. *Management Science*, 52:1647–1660, 2006.

W. Vickrey. Counterspeculation, auctions and competitive sealed tenders. *Journal of Finance*, 16:8–37, 1961.

G. Vulcano, G. van Ryzin, and C. Maglaras. Optimal dynamic auctions for revenue management. *Management Science*, 48:1388–1407, 2002.