

Mechanism Design for the Energy Market



Tim van Heugten

Mechanism Design for the Energy Market

THESIS

submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE

in

COMPUTER SCIENCE

by

Tim van Heugten
born in Delft, the Netherlands



Algorithmics Group
Department of Software Technology
Faculty EEMCS, Delft University of Technology
Delft, the Netherlands
www.ewi.tudelft.nl

© 2011 Tim van Heugten.

Cover picture: (The Midnight Sun by Anda Bereczky) Panoramic shot following the sun above the arctic circle. The picture illustrates the fluctuations in the sun's intensity, but also its predictable behaviour.

Mechanism Design for the Energy Market

Author: Tim van Heugten
Student id: -
Email: stimme@gmail.com

Abstract

Currently, the electricity market is shifting from fossil fuels to renewable resources. The lack of controllability of the output of solar cells and wind turbines conflicts with the requirement that suppliers match their production with the demand at all times. Our solution is to drop this requirement and instead use flexibility on side of the consumer to align the demand with an optimal production schedule.

In this thesis, we first investigate this scheduling problem. We then consider the setting in which consumers and suppliers have private information about their jobs and costs. In this context we propose the Transfer Redistribution Mechanism, which is budget balanced and individually rational. Under the assumption that consumers report truthfully, the mechanism is efficient. We conjecture that in practice truth-telling is a best strategy for consumers.

We present experimental results that show that increased flexibility of jobs reduces the costs of suppliers. Furthermore, the consumers are found to benefit when the flexibility of their jobs is increased, thereby supporting the conjecture that truth-telling is a best strategy.

Thesis Committee:

Chair: Prof.dr. C. Witteveen, Faculty EEMCS, TU Delft
University supervisor: Dr. M.M. de Weerd, Faculty EEMCS, TU Delft
Committee Member: Dr.ir. F.A. Kuipers, Faculty EEMCS, TU Delft

Preface

In a few weeks after this writing, I will no longer be a student. With this report I will end a period of my life that has lasted for ten years.

Over the years I have experienced a lot, and learned a lot and probably did some things wrong as well. However, I would not change much if had the chance to. Perhaps, maybe, try to identify my passion for Computer Science a bit earlier.

At any rate, I found it. And in time, for that matter. Soon I will be ready for the big bad world. For preparing me for that, I would like to thank some people.

First of all, I must thank my parents for supporting me all these years, and keeping their confidence in me even when things did not look so promising. Second I must thank my brother, for ever pretending to overtake me and finally having done so. Third and fourth my sisters for completing our happy family.

I wish to thank Micaela for pushing me to study when such was needed, and for taking care of me when it was appreciated. B'leef for providing me the brake from university that I so badly needed, and my lunch-buddies for saving me a seat and supporting me whenever it showed I did not take a CS bachelor.

Finally, I thank Mathijs for his help during my research and for giving me directions when I could no longer find them, Cees for his motivational and challenging visits to the graduate room, and Fernando for being a part of my thesis committee.

Tim van Heugten
Delft, the Netherlands
August 9, 2011

Contents

Preface	ii
Contents	iii
1 Introduction	1
1.1 Research Questions	2
1.2 Outline of this Thesis	2
2 Mechanism Design	4
2.1 Game Theory	4
2.2 Mechanism Design	6
2.3 Auction Mechanisms	9
3 Problem Statement	13
3.1 Demand Scheduling Problem	13
3.2 Complexity of DSP	14
3.3 Scheduling Algorithm	15
3.4 Game Theoretic Setting	16
4 Existence of a Mechanism for DSP	18
4.1 Properties for DSP	18
4.2 A Groves Class Mechanism for DSP	19
4.3 AGV for DSP	21
5 The Transfer Redistribution Mechanism	24
5.1 Mechanism Outline	24
5.2 Transfers to Suppliers	25
5.3 Transfers to Consumers	27
5.4 Mechanism Properties	31
6 Experiments	33

6.1	Schedule Costs Reduction by Increased Flexibility	33
6.2	Consumer Transfer Benefits With Power Proportional Weights	34
6.3	Consumer Transfer Benefits With Cost Proportional Weights	35
6.4	Consumer Transfer Benefit and Initial Flexibility	37
6.5	Scale and Computability	38
7	Conclusions and Future Work	40
7.1	Future work	41
	Bibliography	43
A	Non-monotonicity of VCG	45
B	Non-IC example for TRM using CPW	47
C	No Stable Nash Equilibrium for TRM, Using CPW	49

Chapter 1

Introduction

The majority of the production of electricity is currently based on the use of fossil fuels. Because of the associated production of greenhouse gases, it is considered an important cause of the greenhouse effect [11]. Also, the resources for fossil fuel are limited, and expected to become less abundant during the coming century. Therefore, in the years to come, energy production is expected to shift from fossil fuels to renewable resources more and more. The benefits of the use of renewable resources, compared to traditional energy production, are obvious. Less obvious, however, are the drawbacks of its use, the biggest drawback being lack of controllability.

When using solar cells or wind turbines, the amount of energy produced is not controllable like it is with traditional resources. As a result, we see that energy aggregators keep a significant portion of traditional energy suppliers in their portfolio. This is necessary, because the aggregators make contracts in advance about the amounts of energy they will be producing or consuming. The production of energy from renewable resources is predicted, but these predictions can be wrong. A buffer of traditional, controllable, production is therefore required [9], in order to cope with the lack of controllability of renewable resources.

In the current market, the energy flow is demand driven. When the demand is high, the suppliers are expected to increase their production, and fulfill the need. Aggregators base their long term contracts and day-ahead trades on predicted consumptions for their consumers. With the use of controllable production, this strategy works, but with an increase in uncontrollable production, it no longer suffices.

However, we observe that for both solar power and wind energy a certain degree of predictability exists. The aim of this thesis is to exploit the predictability of energy production, and create a production driven market. To turn a demand driven market into a production driven market, the flexibility has to shift from suppliers to consumers. We believe that some flexibility on the consumers' side already exists, but is not yet utilized. Part of the challenge will be to expose the existing flexibility, and then to make an optimal schedule for the announced demand and the predicted available production.

Because of the complexity of the scheduling problem, the mechanism cannot be applied on a very large scale. However, a very promising scale would be to apply the mechanism to a block of households which is capable of its own energy production using renewable

resources. In this setting the scale of the scheduling problem is not too large to solve, but the diversity of jobs is large enough to enable good utilization of the fluctuating production. Also, by maximizing the use of energy where it is produced, the load for the distribution network is reduced. Finally, since the demand can be coordinated, the ratio of peak and average demand will decrease. Therefore, the required peak capacity becomes lower, so when the diversity of local energy sources is increased, one could imagine the block of households becoming completely self-supporting.

1.1 Research Questions

In this thesis we make a model of the energy market, and investigate the possibilities and benefits of utilizing the flexibility in the system. With the assumption that solar and wind energy production is cheap, good utilization of these energy sources is reflected by a low costs of the executed schedule. Therefore, the emphasis of this thesis is on cost reduction, not making any further distinction between the energy sources. The research focuses on two aspects of utilisation of flexibility. The first is the scheduling of the moments of consumption.

The research concerning this aspect aims to answer the following questions:

1. How hard is the problem of scheduling demand, such that the costs of energy production are minimized?
2. What algorithms can be used to solve the scheduling problem?
3. What are the benefits of added flexibility on the cost of the resulting schedule?

The second aspect is concerned with the mechanism necessary to expose the flexibility in the system, required for creation of the optimal schedule. This aspect is investigated by answering the following questions:

4. Does a mechanism exist, that is efficient, incentive compatible, budget balanced and individually rational?
5. If not, what properties can be achieved in a mechanism?
6. How much can participants gain from manipulation?

1.2 Outline of this Thesis

The contents of this thesis are divided over seven chapters and three appendices. The chapter being read, Chapter 1, is the introduction to the context of this thesis. In the next chapter, Chapter 2, the field of mechanism design is introduced briefly. Then the mechanism properties relevant in the context of this thesis are explained. Finally, two well known auction mechanisms are introduced, and their properties explored.

The formal model of the scheduling problem will be introduced in Chapter 3. There, the computational complexity of the scheduling problem is analysed, answering Research

Question 1. Also, an answer to Research Question 2 will be presented. The problem will be transferred to the game theoretic setting, by identifying the properties that change when entering the realm of private information.

In Chapter 4 we investigate Research Question 4. Literature has provided a negative result for the general setting. However, in this chapter we identify some properties specific to our problem that could help circumvent the negative result. Unfortunately, two well known mechanisms, VCG and AGV, are found to be unable to deliver all the desired properties.

In Chapter 5, a new mechanism is introduced based on a mechanism with verification. Two sides of the mechanism are investigated separately, the suppliers' side first, the consumers' side second. For the implementation of the suppliers' side of the mechanism, two variants will be investigated. In one variant the transfers for consumers depend solely on their reported, but verifiable, type. In the other, the transfers are derived from VCG transfers, and depend on the influence of the consumer on the cost of the schedule. Finally, the two sides are united and the properties of the mechanism as a whole are analysed. This answers Research Question 5.

After establishing a theoretical foundation for the mechanism, some experimental investigation will be performed in Chapter 6. The benefits of added flexibility for the cost of the schedule are investigated. The effects for the consumers of the two variants of the mechanism are investigated. This yields a promising answer to Research Question 6. The chapter concludes with a short investigation of the computational scalability of the mechanism.

In the last chapter, Chapter 7, the conclusions of the thesis are formulated. The properties of the mechanism are reviewed, and put into context. The chapter ends with some recommendations for future work.

Finally, the appendices provide more detailed examples for some claims made in the different chapters.

Chapter 2

Mechanism Design

In this thesis, the aim is to design a mechanism that can be used to create the optimal schedule of electric power consumption. Before we focus on the problem, we first give an introduction into mechanism design.

2.1 Game Theory

Mechanism design is a field of game theory, sometimes referred to as *reverse game theory*, focusing on the design of the game. In game theory the emphasis is on the analysis of the behaviour of players competing in a game, and on establishing their best actions, in order to achieve their desired outcome.

In general, we say that the behaviour of the players is fully described by their *type*. For each player $i \in I$ in the game, let $\theta_i \in \Theta$ be the type of player i , as an element from the set of types Θ . The combined actions of the players produces the *outcome* of a game. Let O be the set of all possible outcomes of a game.

The benefit of a certain outcome, $o \in O$, for a player is given by its *valuation* v_i . The value depends on the type of the player and the outcome of the game. However, the preference over the outcomes of the game is influenced by another value as well. This is a monetary *transfer*, \mathcal{T}_i , made to the player by the mechanism. The value of the transfer can depend on the outcome of the game too, but is usually not directly dependent on the type of the player.

The preference for an outcome is determined by the *utility* of a player $u_i(\cdot)$. The utility is a function of the type of the player and the outcome of the game. In general, the utility can be written as the sum of the valuation and the transfer, $u_i(\cdot) = v_i + \mathcal{T}_i$.

Given the type of a player, there are several actions that the player can take. The set of actions selected are denoted as the *strategy* of the player. Let $\sigma_i \in \Sigma$ be the strategy of a player, from his set of possible strategies Σ . The combination of the strategies of all players defines the outcome of the game. Therefore, sometimes we will refer to the utility as a function of the players type, strategy and the strategy of other players. Each player will generally choose the strategy that results in the outcome of the game, considering the strategies of other players, that maximizes its utility.

Definition 2.1.1 (Strategy). *A strategy is the plan or decision rule that decides the action a player will take, given a state of the game and the preferred outcome of the player, defined by its type.*

To measure the preference of an outcome of a game, the utility is used for individual players. However, generally the players have a preference for different outcomes. In that case, the preference of individual players cannot be used to determine the best outcome for the group as a whole. To measure the benefits of the outcomes for the group of players as a whole, the social welfare is used.

Definition 2.1.2 (Social welfare). *The social welfare of an outcome is the sum of the valuations of all players in the game for that outcome.*

$$SW(o) = \sum_{i \in I} v_i(o)$$

2.1.1 Solution concepts

The players each have a choice about the strategy they will follow. We will assume, throughout this thesis, that the players are *utility maximizing* and behave in a risk neutral way. Given the objective of each player to maximize its own utility, we can identify three kinds of solutions, in terms of determination of strategies.

The weakest of concepts, but probably also the best known, is that of Nash equilibrium [19]. Let the subscript $-i$ denote the set of players *other* than i .

Definition 2.1.3 (Nash equilibrium). *Given a type profile θ_i , a strategy profile σ is a Nash equilibrium when:*

$$\forall i \in I, \sigma'_i \neq \sigma_i, u_i(\theta_i, \sigma_i(\theta_i), \sigma_{-i}(\theta_{-i})) \geq u_i(\theta_i, \sigma'_i(\theta_i), \sigma_{-i}(\theta_{-i})) \quad (2.1)$$

The Nash equilibrium defines the situation where no player i can improve his utility by unilaterally changing his strategy. In this situation it is best for each player to follow a certain strategy σ_i , only when all other players follow their equilibrium strategy. However, when one player would deviate from its strategy, it might be interesting for another to deviate as well, to obtain a better outcome.

A stronger concept is that of a Bayes-Nash equilibrium.

Definition 2.1.4 (Bayes-Nash equilibrium). *A strategy profile σ is in Bayes-Nash equilibrium if, given its expectations $\tilde{\theta}_{-i}$ about the types of other players:*

$$\forall i \in I, \sigma'_i \neq \sigma_i, \mathbb{E} \left[u_i(\theta_i, \sigma_i(\theta_i), \sigma_{-i}(\tilde{\theta}_{-i})) \right] \geq \mathbb{E} \left[u_i(\theta_i, \sigma'_i(\theta_i), \sigma_{-i}(\tilde{\theta}_{-i})) \right] \quad (2.2)$$

In this concept, the best strategy is based on an expectation about the types of other players. Therefore, the players need not know the exact valuation for outcomes by the other players, but only have knowledge about the distribution of valuations. The strategy to choose is the best response to the distribution of the strategy profile of other players. Therefore, the best strategy might deviate from the strategy that would win, given the actual strategies of other players.

The strongest solution concept is a dominant strategy.

Definition 2.1.5 (Dominant strategy). *Given $\theta_i \in \Theta$, a strategy σ_i is dominant if:*

$$\forall \theta_{-i} \in \Theta_{-i}, \sigma_{-i}, \sigma'_i \neq \sigma_i, u_i(\theta_i, \sigma_i(\theta_i), \sigma_{-i}(\theta_{-i})) \geq u_i(\theta_i, \sigma'_i(\theta_i), \sigma_{-i}(\theta_{-i})) \quad (2.3)$$

This states that whatever strategy the other players follow, for player i it is always optimal to follow strategy σ_i instead of σ'_i .

2.2 Mechanism Design

As each player in a game is optimizing its strategy to increase its own utility, the overall result might be very poor for some players. Therefore, in mechanism design, the focus is on the outcome of the utility for the group as a whole. The goal of mechanism design is to specifically choose the rules for the game, such that the social welfare is maximized.

The outcome of the game can be described by the social choice function of that game.

Definition 2.2.1 (Social choice function). *The social choice function $f(\theta)$ chooses an outcome from O for a game, given the player-types $\theta = (\theta_1, \dots, \theta_I)$.*

The aim is to specify the social choice function by the rules of the game, i.e. to restrict the actions a player can choose, and to manipulate his utility, such that all strategies together result in the desired outcome. In fact, we can restrict our attention a little. Instead of reasoning about all actions in all situations, it is sufficient to simply obtain a statement from the players about their type, when we can apply the revelation principle [7, 17].

Definition 2.2.2 (Revelation principle). *If a mechanism exists that implements a choice function $f(\theta)$ under dominant strategy, Nash or Bayes-Nash equilibrium, then there exists an equivalent direct mechanism, where the equilibrium strategy for each player is truth telling.*

In this thesis, we will work with direct mechanisms only.

Definition 2.2.3 (direct mechanism). *A tuple $(f(\theta), \mathcal{T})$, where:*

- $f(\theta) : \theta \rightarrow O$ is an allocation function.
- $\mathcal{T}(\theta) = (\mathcal{T}_1, \dots, \mathcal{T}_I)$ defines the monetary transfers made to each of the participants.

Although the direct mechanism makes it slightly easier to reason about, Parkes [23] also warns us about the computational properties of such a mechanism, and the requirement for each player to make a report about its type. In some situations it might be desirable for player not to reveal more of their type, when it is imperative that they will not improve the outcome by doing so. This would be possible in an indirect mechanism, when players can respond to each others actions, and iteratively add information as they see fit.

Now that the mechanism is outlined, the properties of the mechanism can be discussed. First the social choice function: a mechanism is said to be efficient when it maximizes the social welfare. Later, we will use the notation $f^*(\theta)$ for an allocation function that maximizes the social welfare.

Definition 2.2.4 (Efficient mechanism). *A mechanism is efficient when its allocation function maximizes the social welfare.*

$$\forall \theta \in \Theta, f(\theta) = \arg \max_{o \in O} \sum_{i \in I} u_i(\theta_i, o) \quad (2.4)$$

For the other properties of mechanisms, there exist different levels of application. The strongest sense is *ex post*, meaning after the fact. In this concept we can apply the mechanism, and look back to see that a property held its truth, no matter to what instance the mechanism was applied. A weaker concept is *interim*, meaning that each player knows about its type and the rules of the mechanism, but only has expectations about the types of others. The weakest concept is *ex ante*, meaning before the fact. With an *ex ante* property, the property holds when the players know the rules of the mechanism and have expectations about their types and the types of others, but no strict knowledge about the other types and therefore the outcome of the mechanism. If a property holds in an *ex ante* or *interim* concept, the property will be true in expectation over all instances, given the known distributions of player types. However, it is not guaranteed that the properties hold for all instances.

Here we only introduce those concepts of properties that we will need later on. The first property that is introduced defines the cost, or profit, of employing the mechanism. When the transfers made to the players do not sum up to 0, the maintainer of the mechanism might need to add value to the mechanism. The mechanism is then said not to be budget balanced, or, when the maintainer has a net negative transfer to the mechanism, it is weakly budget balanced. The desired property in this thesis, however, is strong budget balancedness. Clearly, a mechanism which is not budget balanced is undesirable, since it would cost money to execute. On the other hand, a weakly budget balance mechanism would generate money, but when this money finds its way back to the players it influences their strategy. To prevent this, the best choice is a strong budget balanced mechanism.

Definition 2.2.5 (Ex post budget balanced mechanism). *A mechanism is budget balanced when, the sum of all transfers is 0.*

$$\sum_{i \in I} \mathcal{T}_i = 0 \quad (2.5)$$

Two other properties of mechanisms that will be discussed are properties that must hold for each player. First, the property that players are not better off when not participating. This implicitly makes the assumption that not participating yields a utility of 0.

Definition 2.2.6 (Ex post individually rational). *A mechanism is ex post individually rational if the ex post utility is nonnegative.*

$$\forall \theta \in \Theta, i \in I, u_i(\theta_i, f(\theta), \mathcal{T}_i) \geq 0 \quad (2.6)$$

The weaker variant that will be used is *ex ante* individual rationality.

Definition 2.2.7 (Ex ante individually rational). *A mechanism is ex ante individually rational if the ex ante utility is nonnegative.*

$$\forall i \in I, \mathbb{E} \left[u_i(\tilde{\theta}_i, f(\tilde{\theta}), \mathcal{T}_i) \right] \geq 0 \quad (2.7)$$

When applying a mechanism it is desirable that the players cannot manipulate the outcome of the game by making untrue claims about their type. Therefore, a desirable property of the mechanism is to incentivise the players to only make truthful reports about their type. The best way to enforce this, is to make sure that the best strategy they have is to make truthful report. Again we identify two levels of this property.

Definition 2.2.8 (Bayes-Nash incentive compatible (BNIC)). *A direct mechanism is Bayes-Nash incentive compatible if the truthful report is a Bayes-Nash equilibrium of the game described by the mechanism.*

In the Bayes-Nash variant, the players must act before they know the types of other players. Given the prior knowledge about the distribution of other players types, truthful reporting is their best strategy.

The stronger variant is dominant strategy incentive compatibility.

Definition 2.2.9 (Dominant strategy incentive compatible (DIC)). *A direct mechanism is dominant strategy incentive compatible if the truthful report is a dominant strategy equilibrium of the game described by the mechanism.*

In this variant it is clear that a truthful report about the players type is always the best strategy, no matter what the other players types are.

2.2.1 Mechanism with verification

In some situations, it is not possible to design a mechanism which is incentive compatible for all properties of the players' types. For example, in task allocation, the players might represent their task with a smaller execution time in order to be scheduled earlier. The mechanism would not be able change its allocation thusly, such that truthful reporting becomes a dominant strategy.

However, under certain conditions it is possible to employ a *mechanism with verification*, introduced by Nisan and Ronen [22]. In a mechanism with verification, there are two stages in the mechanism: one where the players communicate and decide on the allocation, and then an execution stage where the agreed allocation is executed. The transfers are established only after the execution is completed. During the execution stage, the mechanism is able to verify some of the properties of the players. In the example above, the execution time of the tasks would be known to the mechanism, after all the tasks have been executed.

Since the transfers in a mechanism with verification are only established after the execution stage, the actual, verified, properties of the players can be taken into account. The transfers therefore can depend both on the reported types, and on the verified types of the players. Players that reported something different from their verifiable type can be "punished". In effect, the mechanism makes it undesirable for the players to report anything but their true types, and truthfulness is enforced by the verification mechanism.

2.3 Auction Mechanisms

Auction mechanisms are used to distribute goods among interested participants (players). Well known auction mechanisms are the single item auctions English auction and Dutch auction.

2.3.1 Single Item Auction

In a single item auction, only one item is up for auction, therefore, only one player can acquire the good. We will consider English auction and Dutch auctions. Both these auction mechanisms are *efficient*, i.e. they maximize the social welfare of all players, by allocating the object to the player that values it the most. However, there is an important difference between the two, in how they determine which player values the object the most.

In an English auction, the price increases as the players cry out their new higher bid, for which they wish to obtain the object. As soon as the latest bid exceeds a player's valuation, that player will stop bidding. Any remaining players can bid just slightly more than the last bid to obtain the good. The winning player, therefore, can obtain the item for a price that is unrelated to his own valuation.

In the Dutch auction, on the other hand, the price decreases, until one player is willing to pay that price for the good. Then he makes his bid, and obtains the object for the price he bids. This price is fully determined by the value of the object for this player.

Using the revelation principle, we can construct two forms of direct mechanisms, based on the previous two auctions. While the strategies for players in the direct equivalents are the same, the expected revenue for the center can be different [16]. The second price sealed bid auction is the direct equivalent of an English auction, where each player submits only one bid. The player with the highest bid wins the objects, and the player making that bid pays only the second highest bid. The other is a first price sealed bid auction, in which the players submit their bids, and equivalently to the Dutch auction, the highest bid wins, and the winning player pays the amount of his own bid.

In a first price auction, the bids that players make do not typically reflect the true valuation for an object. After all, if they would bid exactly that value, their utility would be zero. Contrastingly, the bids that players make in a second price auction, are typically the true valuations for the object. After all, they will pay the price of the second best bid, which is already lower than their valuation, therefore there is no incentive to decrease the bid. Furthermore, would the player make a higher bid, it is possible that he ends up paying more than his valuation of the object. As a result the second price sealed bid auction, also known as a Vickrey [24] auction, is incentive compatible. On the other hand, first-price sealed bid auction is not.

2.3.2 VCG Mechanism

The previously described sealed bid auction mechanism can be extended to the auction of multiple objects. There, the winner pays the bid that had the second highest value for the item. Because there is only one item for sale, that is exactly the social welfare that would

have resulted when the winner was not present. Generalizing the setting from a single-item to a combinatorial auction, following the same idea, charging the player the social welfare that would have resulted without him, produces a dominant strategy mechanism.

It turns out that practically all mechanisms that implement a dominant strategy fall in the same class of mechanisms. This class of mechanisms is called the Groves class of mechanisms, proposed by Vickrey [24], Clarke [5] and Groves [8]. The Groves class of mechanisms are defined by the outcome they choose, and the transfers that are made by the players. The definition of the transfers, however, contains a function that is not defined by the Groves class, therefore, there exist infinitely many mechanisms that fall in the Groves class.

Definition 2.3.1 (Groves class of mechanisms). *A direct mechanism $(f(\theta), \mathcal{T})$ is a Groves mechanism if and only if:*

- $\forall \theta_i \in \Theta, f(\theta) \in \arg \max_{o \in \mathcal{O}} v_j(\theta_i, o)$, it executes $f^*(\theta)$.
- $\forall i \in I, \mathcal{T}_i(\theta_i) = (\sum_{k \neq i} v_k(\theta_{-i}, f^*(\theta))) - h_i(\theta_{-i})$, where h_i is a functions that does not depend on the type of i .

The outcome of a Groves mechanism is the outcome that maximizes social welfare, according to the reported types of the players. The transfers to each player are defined by the reported valuation of the other players for the outcome, and a value that is independent of the players report. The value of h_i , the *charge*, is a constant from the point of view of player i . Therefore, in order to maximize the payments, the player should aim to maximize the other players' valuation for the outcome of the mechanism. Thus, the Groves class of mechanisms has the property stated in Lemma 2.3.2.

Lemma 2.3.2. *Every mechanisms in the Groves class is efficient and incentive compatible.*

For the proof of Lemma 2.3.2, we refer to the work of Cavallo [3].

The Groves class of mechanisms define a broad set of mechanisms. We are free to pick the h_i , that will fulfill our needs best. When the mechanism is required to be ex post individually rational, then a Clarke tax can be used.

Definition 2.3.3 (Clarke tax). *A Clarke tax is the choice for the charge that is defined as:*

$$h_i = \max_{o \in \mathcal{O}} \sum_{k \neq i} v_k(o)$$

We define the mechanism from the Groves class of mechanisms that uses the Clarke tax as its charge as a VCG mechanism. Let $f(\theta_{-i})$ be the outcome that maximizes the social welfare when i does not participate, i.e. the outcome that determines the Clarke tax.

Definition 2.3.4 (VCG mechanism). *A VCG mechanism is a mechanism in which the social choice function maximizes the social welfare, and the transfers are defined as:*

$$\mathcal{T}_i = \sum_{k \neq i} v_k(\theta_k, f(\theta)) - \sum_{k \neq i} v_k(\theta_k, f(\theta_{-i})) \quad (2.8)$$

The difference between the two sums of the VCG transfer is the difference of the summed valuation of the players other than i , for the outcomes with and without i . Again, the transfers assigned to player i are independent of the bid of player i itself. The VCG mechanism achieves most of the desired properties, as claimed in Lemma 2.3.5.

Lemma 2.3.5. *The VCG mechanism is efficient, incentive compatible and individually rational when the no negative externalities property holds.*

The proof for the individual rationality part of the lemma is a variation on the proof by Nisan [20].

Proof for Lemma 2.3.5. The mechanism falls in the Groves class, since the mechanism is efficient, and the transfers follow the Groves definition. The Clarke tax is independent on the players type, and the other term follows straight from the mechanism. Therefore, the mechanism is efficient and incentive compatible.

A mechanism is individually rational when the utility is non-negative for each player. Using the transfers from the mechanism, the utility for a player is given by: $v_j(f(\theta)) + \sum_{k \neq i} v_k(f(\theta)) - \sum_{k \neq i} v_k(f(\theta_{-i})) \geq \sum_k v_k(f(\theta)) - \sum_k v_k(f(\theta_{-i})) \geq 0$. The first inequality follows from the no negative externalities property, $v_j(f(\theta_{-i})) \geq 0$. The second step holds because the allocation function selects the optimal outcome. \square

While the VCG mechanism does achieve most of the desired properties, it is not budget balanced. In the next section a mechanism is introduced that is budget balanced.

2.3.3 AGV Mechanism

When the requirements of ex post individual rationality and dominant incentive compatibility are relaxed, it is possible to construct an ex ante budget balanced auction mechanism. This is the AGV mechanism, named after its inventors d'Aspremont and Gérard-Varet [6]. Sometimes it is referred to as AAGV, since it was simultaneously developed by Arrow [1]. In this mechanism the transfers that are made to a player are reclaimed from the other players. In effect, no transfers are made to or taken from the group of players as a whole.

The AGV mechanism is a Bayes-Nash incentive compatible mechanism that achieves budget balancedness. Transfers in this mechanism are solely determined by the expected valuations that players obtain from the expected outcome.

Definition 2.3.6 (AGV mechanism). *The AGV allocation function selects the outcome that maximizes the social welfare. Let n be the number of players, and let ESW_{-i} be the expected social welfare for the players other than i , that is $\mathbb{E}[SW_{-i}] = \sum_{k \neq i} \mathbb{E}[v_k(o)]$, the AGV transfers then are defined as:*

$$\mathcal{T}_i = ESW_{-i} - \frac{1}{n-1} \sum_{k \neq i} ESW_{-k} \quad (2.9)$$

The expected social welfare is calculated by evaluation of the outcomes for all possible player types, and multiplying them by their probability. This assumes that the probability distributions are known beforehand, and that the same probability distributions are known to all players.

That this mechanism is budget balanced is easily observed, by summing the transfers of all players. A proof for the incentive compatibility property is provided by Krishna [13]. Finally, the AGV mechanism is ex ante individually rational.

2.3.4 Bilateral trade

So far we have been considered with the situation where one auctioneer, or center, has an item for sale, and several players are bidding to obtain it. Turned around, in a reversed auction, with the center desiring to acquire an object and several sellers offering, the VCG and AGV mechanisms are equally applicable. The important difference to keep in mind then, is that we assume the players to have a negative valuation, or cost, associated with losing the item. Stated slightly differently, the sellers have had costs for creating the item in the first place. Either way, the winner of these auctions is the one with the highest valuation still, i.e. the least negative valuation, or in terms of cost it is the player that associates the lowest cost with the item.

Nevertheless, the calculations stay the same. Using a negative value for the valuation, it turns out the center himself needs to transfer value to the players, instead of the other way around. Quite expected, since it is the center in this situation that obtains an item. Even budget balancedness is preserved under the AGV mechanism in this reversed auction.

However, it is in general not possible to combine both settings in one and retain all the properties. That is to say, one cannot have several buyers bidding while several sellers are offering and maintain the properties defined earlier for these mechanisms. More formally, we get the following impossibility result by Myerson and Satterthwaite [18]:

Lemma 2.3.7. *In the bilateral trade problem, there is no mechanism that is efficient, Bayes-Nash incentive compatible, individually rational, and at the same time weakly budget balanced.*

Chapter 3

Problem Statement

In this thesis we try to decrease energy production costs by adapting the moment of consumption to times when production is cheap. We consider jobs that cannot be preempted, with a given deadline and try to match their moment of execution with convenient, low cost, moments for the suppliers. Each supplier has, for each time unit, a nondecreasing function of cost per unit energy production.

3.1 Demand Scheduling Problem

Here we present the formal model of the problem. The problem is named the Demand Scheduling Problem (DSP), since we assume the costs for the suppliers to be fixed, and try to minimize energy production costs by adapting the schedule of the jobs, i.e. scheduling the demand.

The problem takes a number of jobs $j \in J$, each job representing something the consumer needs done. For each consumer $i \in I$ we have a set of jobs J_i , such that when job j belongs to consumer i , we have that: $j \in J_i$. Each job consists of a set of properties, defining how the job can be scheduled.

Definition 3.1.1 (Job). *A job, j , in the DSP is defined by its properties:*

- a_j , the arrival time of the job, when it can first be executed,
- s_j , the duration of the job,
- p_j , the power requirement per unit time for the complete duration of the job, and
- d_j , the deadline, when the job must be completed.

The model uses a concept of discretized time. We require the jobs to be feasible, so $a_j + s_j \leq d_j$. Also, the jobs must fall within the time window under consideration; $a_j > t_0$ and $d_j \leq t_f + 1$, with t_0 the first unit of time and t_f the final unit of time that is considered in the schedule. Furthermore, these parameters together imply the flexibility (number of possible starting times) to be $f_j = d_j - a_j - s_j + 1$.

To provide the consumers with energy, we consider a number of suppliers $s \in \mathcal{S}$.

Definition 3.1.2 (Supplier). *A supplier, s , in DSP is defined by a cost function:*

- $\phi_s(p_t^s, t)$, the unit cost with a power production of p_t^s , at time t .

In general, the cost functions are monotonic in p_t^s , but typically not in t . The total cost endured by the supplier in order to create the energy is $\Phi_s(p_t^s, t) = p_t^s \phi_s(p_t^s, t)$.

In Section 2.2 the notion of an *efficient mechanism* was introduced. The objective of the mechanism constructed in this thesis is for it to be efficient. However, this requires that the valuation of the outcomes of the mechanism are defined and known to the schedule maker. For the suppliers, the valuation is defined, as it is the cost associated with the production of power that a supplier produces in the schedule. For the jobs, on the other hand, the valuation is not defined as a property of the problem. Although, from the context the jobs originate from, consumers wanting to get an energy consuming task done, it is to be expected that some valuation is associated with the jobs.

Unfortunately, in the real world it would be impractical to require a valuation report for all the jobs that consumers present to the mechanism. First of all, because the consumers would need some way to communicate this valuation, but second, it would constantly interrupt their daily business to make their valuation for a job known to the mechanism. While these two difficulties make the utilization of per job valuations problematic, most important is probably that it would require that consumers define a quantitative valuation for every job. This task is unreasonable, because consumers simply lack the information necessary to give these quantitative valuations [15].

Since a lot of the mechanism properties depend on the efficiency property, later there will be some assumptions made to provide some form of indication for the valuation by the consumers. For the schedule to be efficient it will then be sufficient to minimize the total cost for the suppliers.

3.2 Complexity of DSP

In order to investigate the complexity of DSP, we use the bin packing problem [10], for which it is known that its computational complexity is NP-complete. We show that the bin packing problem can be reduced to DSP. The bin packing problem is defined as follows:

Definition 3.2.1 (Bin packing). *Given a list $L = (l_1, \dots, l_n)$ of integers and a bin size V , is there a way to distribute the integers over B bins, such that for each bin $k = 1, \dots, B$ it holds that its size $|S_k| = \sum_{l_j \in S_k} l_j$ is less than or equal to the bin size: $|S_k| \leq V$.*

To perform the reduction from bin packing to Demand Scheduling Problem, we take B units of time, to represent the B bins. Then, we create a job for each integer l_j in L . The integers from L can be assigned to all of the bins, so it must be possible for the jobs to be assigned to any of the units of time in the DSP instance. Therefore, we take the complete time-span as the allowed time, using parameters a_j and d_j . The integers can only

be assigned to one bin, so we require the time-span of the jobs to be 1. Finally, we set the power demand per unit time of the jobs equal to the integer value l_j .

The properties of the equivalent DSP instance become: $s_j = 1$, $p_j = l_j$, $a_j = 1$ and $d_j = B + 1$ for all $j = 1, \dots, n$. Consider one supplier, and let the cost for this supplier at moment t be $\Phi(D_t > V, t) = \infty$, and bounded for demand lower than or equal to V . The question to be answered is: does a schedule exist, such that $\sum_t \Phi(D_t, t)$ is bounded. The answer to this question answers the bin packing problem, the correctness of this reduction will be proven in the following.

Lemma 3.2.2. *Bin packing yes-instances map to DSP yes-instances.*

Proof of Lemma 3.2.2. If S_t is the bin to which integer l_j is assigned in the bin packing problem, we execute it in time unit t , so $e_{j,t}$ is 1 only for this particular t . Therefore, $D_t = \sum_j e_{j,t} p_j = \sum_j e_{j,t} l_j = \sum_{l_j \in S_t} l_j$ is equal to the size of S_t .

So, if $|S_t| \leq V$ then also $D_t \leq V$ and thus $\Phi(D_t, t)$ is bounded. This holds for any t , so if the bin size $|S_t| \leq V$ for all t , then the cost for each t is bounded, and therefore $\sum_t \Phi(D_t, t)$ is bounded as well. \square

Lemma 3.2.3. *DSP yes-instances map to bin packing yes-instances.*

Proof of Lemma 3.2.3. If $\sum_t \Phi(t, D_t)$ is bounded, we see that for all t it holds: $\Phi(D_t, t)$ is bounded, so D_t must be less than or equal to V . Because of the definition of D_t , and the construction of the reduction, we know that $D_t = |S_t|$. Combining the two previous observations we see that $|S_t| = D_t \leq V$. So, the bin packing instance must have a solution. \square

Using this reduction from an NP-complete problem to DSP, we observe Corollary 3.2.4.

Corollary 3.2.4. *Demand Scheduling Problem is NP-hard.*

3.3 Scheduling Algorithm

Given the definition from Section 3.1 it is possible to compute a schedule that minimizes the costs for the suppliers. In this thesis we will restrict ourselves to supplier cost functions that can be modeled as a mixed integer quadratic constraint program (MIQCP). This allows us to use the MIQCP solver from the IBM ILOG CPLEX Optimizer package. The CPLEX Optimizer package is well known for its efficiency on solving scheduling problems in general. However, this MIQCP solver does not provide the exact optimal solution to the problem, but an approximation.

Before the problem can be solved by a MIQCP solver, the problem must be transformed into an MIQCP. Patterns for the construction of linear programs can be found in literature, see for example [4] and [12]. The MIQCP is a natural extension, and the construction follows along the same lines.

The supplier cost function must have a restricted form, in order to fit in a MIQCP. This is achieved by letting the unit cost for a certain production be defined as the maximum value of a set of linear constraint functions. Each constraint functions forms a lower bound, as a function of produced supply, on the unit cost for the supplier. These functions, of the form

$ap_t^s + b$, are defined by the two parameters a and b , and this is also how they will return in the MIQCP. The p_t^s used in the previous equation is the power that the supplier provides, this must be a non-negative real value. Let q_s be the number of linear functions supplier s has.

The MIQCP for the DSP is given below. An extra set of coefficients, $e_{j,t}$, is introduced, indicating if job j is executed at time t . The schedule establishes a begin time b_j and an implied finish time $b_j + s_j$ with each task. The value of $e_{j,t}$ is defined as 1 if $b_j \leq t < b_j + s_j$, and 0 otherwise.

Equation 3.1 determines the objective of the MIQCP, it shows that the sum over all suppliers and all units of time of the cost $\Phi_{s,t}$ that supplier s has in unit of time t must be minimized. Each $\Phi_{s,t}$ is bounded below by the constraint function of the supplier for that unit of time, as shown in Equation 3.2. The production by suppliers and demand from jobs are related by Equation 3.3: at any time the sum of the production matches the sum of the demand.

Because the jobs cannot be preempted and must be executed between the arrival time and deadline, the final three equations are required. To ensure that the begin time b_j falls within the proper range, the allowed values are restricted by $a_j \leq b_j \leq d_j - s_j$. The execution coefficient is linked to the begin time by Equations 3.4 and 3.5. The value M is a large constant, such that the two equations together ensure that $e_{j,t}$ is never 1 before the begin time or after the more than s_j later than the begin time. Finally, Equation 3.6 guarantees that the execution coefficient is 1 in as many units of time as the time span of the job.

$$\min \sum_{s \in \mathcal{S}} \sum_t \Phi_{s,t} \quad (3.1)$$

subject to

$$(a_t^{s,q} p_t^s + b_t^{s,q}) p_t^s \geq \Phi_{s,t} \quad \begin{array}{l} s = 1 \dots m \\ t = t_0 \dots t_f \end{array} \quad (3.2)$$

$$\sum_{s \in \mathcal{S}} p_t^s = \sum_{j \in \mathcal{J}} e_{j,t} p_j \quad \begin{array}{l} q = 1 \dots q_s \\ t = t_0 \dots t_f \end{array} \quad (3.3)$$

$$(e_{j,t} - 1)M + b_j \leq t \quad \begin{array}{l} j = 1 \dots n \\ t = t_0 \dots t_f \end{array} \quad (3.4)$$

$$(e_{j,t} - 1)M - b_j \leq s_j - t - 1 \quad \begin{array}{l} j = 1 \dots n \\ t = t_0 \dots t_f \end{array} \quad (3.5)$$

$$\sum_t e_{j,t} = s_j \quad j = 1 \dots n \quad (3.6)$$

3.4 Game Theoretic Setting

In Section 3.1 we have seen the formalization of the problem. Given a problem instance the optimal schedule can be calculated straightforwardly. However, the straightforward method

may not be efficient, considering the complexity of the problem, as shown in Section 3.2. A subset of the possible problem instances, those where the suppliers have linear unit cost function, can be solved by the algorithm in Section 3.3.

However, this straightforward calculation of the schedule requires full knowledge of the problem. The entire type information for all jobs and suppliers should be known to the algorithm before it can begin calculating the schedule. In the real setting, the jobs are created by consumers, and the job properties are not inherently known to the schedule maker. In fact, *all* the properties that define a job or supplier are supposed to be private information. Therefore, the suppliers and consumers have a possibility to provide incorrect information to the schedule maker. It is a requirement of the mechanism to ensure truthful reports from the participants. It will be the goal of the following chapters to find a mechanism that achieves this.

Chapter 4

Existence of a Mechanism for DSP

In the literature it is stated that in a bilateral trade setting, with no restriction on the types, it is not possible to have a mechanism that is efficient, incentive compatible, individually rational and budget balanced (Myerson and Satterthwaite [18]). In the context of DSP, however, there are some restrictions on the types of the suppliers and jobs. Furthermore, there are restrictions on the schedule that the mechanism produces. In the first section, these restrictions will be formalized. In the other sections in this chapter, the possibilities of a mechanism with the desired properties will be investigated, given these restrictions.

4.1 Properties for DSP

In this section some of the properties of the problem, that follow from the practical context, will be accumulated. The first property is one that restricts the possible outcomes of the mechanism.

Whenever a consumer reports a job to the mechanism, it should not be possible that the job does not get executed. Therefore, it is a requirement of the mechanism's outcome that all the jobs are scheduled at a feasible moment, as is the situation in current practice. This, of course, begs the question whether this is a reasonable obligation for the mechanism. How, after all, can the mechanism ensure that such a feasible schedule exists.

In fact, such a guarantee can be derived from current practice. If the mechanism would be unable to ensure the existence of a feasible schedule, this would mean that in certain units of time there is a bigger demand for energy than what the suppliers can provide. However, when the mechanism is applied and jobs report their flexibility, the schedule that would result from executing the jobs when they first arrive remains feasible, this is in fact the exact schedule that is executed in current practice. Therefore, the mechanism has the possibility to execute the schedule that would have resulted without the mechanism. Then, by Assumption 1, it is implied that a feasible mechanism exists. The assumption is even stronger, and states that each job's cost, in current practice, is below the valuation for the execution of that job. Later, this stronger interpretation will be used to prove individual rationality for the mechanism.

Assumption 1. *The currently employed energy market is individually rational.*

In the previous chapter it was argued that it is not possible to acquire a quantitative valuation from the consumers. Therefore, the valuation for the jobs must be captured in qualitative properties, as best possible. With Assumption 1, a first qualitative property has been provided, that can be seen as an observation from current practice. The next property, however, is not supported by observation, and must be seen more as a requirement for the mechanism.

When the jobs report their flexibility to the mechanism, the job is considered to be indifferent about the moment of execution within the feasible window. This property results in Assumption 2. This assumption makes it easier for the mechanism to establish the efficient schedule, since it does not need to consider a preference from the job, other than the reported window.

Assumption 2. *The valuations for the execution of a job is constant between its arrival time and deadline.*

We do not consider this last assumption to be too restrictive for practical purposes. Even though it is possible that a preference exists for some jobs, for example, to have them completed as soon as possible. When this preference is stronger than the anticipated cost benefit of the delayed completion, the job report could be restricted to the most preferred moment of execution only. By not restricting the feasible window, therefore, the job indicates that the variations of its valuation for the moment of execution are smaller than the anticipated difference in cost. The efficiency is then dominated by the cost for the suppliers, which supports the assumption that variations in valuation for moment of execution can be neglected, i.e. it can be considered constant.

4.2 A Groves Class Mechanism for DSP

In order for the mechanism to maximize the social welfare, it is necessary that the mechanism has access to the true types of the jobs and suppliers. Therefore it is a requirement of the mechanism to be incentive compatible. One class of efficient mechanisms where truth telling is a dominant strategy is the Groves class of mechanisms. In this section, a Groves mechanism will be constructed for DSP. Then the IR and BB properties will be investigated.

Let us first look at the transfers for the jobs only. The Groves transfers are given by a constant, K_j , added to the sum of valuations of the other participants.

$$\mathcal{T}_j = K_j + \sum_{k \neq j} v_k(f(\theta))$$

Every transfer scheme of this form is efficient and incentive compatible. Since the mechanism is also required to be individually rational, let K_j be defined by the Clarke tax.

$$\mathcal{T}_j = -\max_{o \in O} \sum_{k \neq j} v_k(o) + \sum_{k \neq j} v_k(f(\theta))$$

Using Assumption 2, and the property that all jobs are always executed, the valuation for all jobs other than j is constant. The sums over the other jobs cancel each other out, and the transfer for a job j thus only depend on the suppliers' valuation:

$$\mathcal{T}_j = -\max_{o \in O} \sum_{s \in S} v_s(o) + \sum_{s \in S} v_s(f(\theta)) \quad (4.1)$$

For the suppliers, a similar construction can be performed. Application of Grove mechanism with a Clarke tax to the suppliers gives a transfer of:

$$\mathcal{T}_s = -\max_{o \in O} \sum_{k \neq s} v_k(o) + \sum_{k \neq s} v_k(f(\theta))$$

Again the valuations of jobs cancel out, applying Assumption 2 and the observation that all jobs are always executed. The transfers for suppliers therefore become:

$$\mathcal{T}_s = -\max_{o \in O} \sum_{\substack{k \neq s \\ k \in S}} v_k(o) + \sum_{\substack{k \neq s \\ k \in S}} v_k(f(\theta)) \quad (4.2)$$

The transfers for all participants follow the transfers of the VCG mechanism in Equation 2.8. The transfer scheme is therefore efficient, incentive compatible and individually rational. However, the transfers produced by this mechanism are not budget balanced. As stated in Theorem 4.2.1, it is not possible to create a mechanism that combines these four properties, since the problem remains similar to bilateral trade, even with the two previous assumptions.

Theorem 4.2.1. *For the private information setting of DSP, no efficient mechanism exists that is individually rational and budget balanced where truth telling is a dominant strategy, despite Assumption 1, the current energy market is IR, and Assumption 2, consumer valuation is constant for execution of jobs within their deadline.*

This theorem can be proven by presenting a counter example for the VCG mechanism previously derived.

Proof of Theorem 4.2.1. The mechanism is a Groves mechanism, so it is efficient and incentive compatible by Lemma 2.3.2. By application of the Clarke tax, the mechanism became a VCG mechanism. By Lemma 2.3.5, the mechanism is therefore also individually rational. This proof will consist of an example, showing that the mechanism is not weakly budget balanced on all instances.

In the example there is one unit of time. Let there be 5 jobs, having a power demand of 1. There are 5 suppliers that have a cost function given by Equation 4.3, where the following relation holds $0 < a < b$.

$$\phi_s(p) = \begin{cases} a & \text{if } p \leq 1, \\ b & \text{otherwise.} \end{cases} \quad (4.3)$$

The valuation when all suppliers and jobs participate is $5a$. When a job is removed, the cost is $4a$, and with one supplier removed, the cost becomes $4a + b$.

The transfers for each job are:

$$p_j = 4a - 5a = -a$$

The transfers for the five suppliers are:

$$p_s = 4a + b - 4a = b$$

Summing the transfers of all participants yields:

$$\sum_j p_j + \sum_s p_s = 5(b - a) > 0 \quad (4.4)$$

Clearly, since $b > a$, the mechanism implementer must make positive transfers to the players, i.e. the mechanism runs a deficit.

Now suppose some other mechanism exists, that is both incentive compatible and efficient. By the revenue equivalence principle, the expected transfers for the jobs in that mechanism is different from the transfers in this mechanism only by a constant K . The expected transfers for the suppliers differ only by a constant L .

For a job with a power consumption of 0, the transfer in this mechanism is 0. Suppose the other mechanism is also individually rational. Then, it must hold that $K \geq 0$. Similarly, for suppliers not participating in the optimal schedule, the transfer in this mechanism is 0. Therefore, it follows that $L \geq 0$.

By adding constants K and L to the transfers, the expected deficit can only increase. Thus, there does not exist an efficient mechanism for DSP that is incentive compatible, individually rational, and budget balanced. \square

4.3 AGV for DSP

As seen in the previous section, it is not possible to devise an efficient mechanism that is incentive compatible, individually rational and budget balanced. The AGV mechanism has a weaker strategy concept, that of Bayes-Nash incentive compatibility, and is efficient and budget balanced. Krishna et. al. [14], state that an efficient mechanism that is incentive compatible, individually rational and budget balanced only exists when the VCG is weakly budget balanced. In this section we will confirm a negative result for the AGV mechanism.

The AGV mechanism bases the transfers on expectations of valuation by the participants. So far, only qualitative assumptions about the valuation of the jobs have been made, in this section it becomes necessary to establish a more quantitative analysis. Using Assumption 1, a lower bound on the valuation for the jobs can be defined. After all, for each job, the valuation must be higher than what is charged for it. Therefore, there exists a value, v_j^0 , that must be larger than or equal to the highest price possible that is charged to j in the current market. This value is a lower bound for the valuation of the jobs, as shown in Equation 4.5. The sum of all lower bounds will also be used later on, let $v_{tot}^0 = \sum_{j \in J} v_j^0$.

$$\forall j \in J, v_j \geq v_j^0 \quad (4.5)$$

In the following example we calculate AGV transfers, for a single time unit schedule. Consider the situation where we have n jobs, $j \in J$, and m suppliers, $s \in \mathcal{S}$. The valuation of the suppliers is given by their production costs $v_s = -\Phi_s(p_i^s)$. Once a schedule has been created, the costs of each supplier are also determined. We will refer to the cost of a supplier using Φ_s . The total cost can therefore be written as $\sum_{s \in \mathcal{S}} \Phi_s = \Phi_{tot}$.

The AGV transfers are defined in Definition 2.3.6 but are repeated here using z as the total number of participants:

$$T_i = ES W_{-i} - \frac{1}{z-1} \sum_{k \neq i} ES W_{-k} \quad (4.6)$$

The important values that determine the AGV transfers are the expected social welfare for the other participants. In the DSP this can be split in two terms. First, the expected social welfare with one job removed is:

$$ES W_{-j} = v_{tot}^0 - v_j^0 - \Phi_{tot} \quad (4.7)$$

Second, the situation with one supplier removed is:

$$ES W_{-s} = v_{tot}^0 - \Phi_{tot} + \Phi_s \quad (4.8)$$

Note that these values are expectations: although the job valuations will not change, the supplier costs are estimates based on a priori known probability distributions over the supplier and job types. Substituting Equations 4.7 and 4.8 in Equation 4.6, the expression for the transfers for the jobs is:

$$\begin{aligned} T_j &= ES W_{-j} - \frac{1}{n+m-1} \sum_{k \in J \setminus \{j\}} ES W_{-k} - \frac{1}{n+m-1} \sum_{s \in \mathcal{S}} ES W_{-s} \\ &= ES W_{-j} - \frac{n-1}{n+m-1} (v_{tot}^0 - \Phi_{tot}) - \frac{m}{n+m-1} (v_{tot}^0 - \Phi_{tot}) + \frac{1}{n+m-1} (v_{tot}^0 - v_j^0 - \Phi_{tot}) \\ &= v_{tot}^0 - v_j^0 - \Phi_{tot} - \frac{n+m-1}{n+m-1} (v_{tot}^0 - \Phi_{tot}) + \frac{1}{n+m-1} (v_{tot}^0 - v_j^0 - \Phi_{tot}) \\ &= -v_j^0 + \frac{1}{n+m-1} (v_{tot}^0 - v_j^0 - \Phi_{tot}) \end{aligned} \quad (4.9)$$

Unfortunately, using the assumption that the current energy market is IR, the same cannot be said about the AGV mechanism. After all, the expected utility for a job, u_j , is the sum of its valuation and its transfers.

$$\begin{aligned} u_j &= v_j + T_j \\ &= v_j - v_j^0 + \frac{1}{n+m-1} (v_{tot}^0 - v_j^0 - \Phi_{tot}) \\ &\geq \frac{1}{n+m-1} (v_{tot}^0 - v_j^0 - \Phi_{tot}) \end{aligned} \quad (4.10)$$

The inequality in Equation 4.10 is not a general guarantee that IR will be satisfied. For example, with only one job, the expected utility is $-\frac{1}{n+m-1} \Phi_{tot}$, negative. In general, from

the assumption that the current energy market is IR we can derive $v_{tot}^0 \geq \Phi_{tot}$, but not that $v_{tot}^0 - v_j^0 \geq \Phi_{tot}$. Without any further assumptions about the job valuation, the utility cannot be guaranteed to be positive. Therefore, the AGV mechanism for this problem is not individually rational, since it does not achieve ex ante IR, the weakest form of IR.

Chapter 5

The Transfer Redistribution Mechanism

In the previous chapter it was proven that an efficient mechanism that is individually rational, budget balanced and incentive compatible cannot exist. Also, the AGV mechanism can be shown not to be individually rational, based on observations made about the current market. Another practical difficulty of applying the AGV mechanism is its requirement of the existence of expectations about the types of the participants.

Therefore, the two mechanisms that are usually considered, VCG and AGV, are not suited for this problem. In this chapter we search for an alternative and introduce the Transfer Redistribution Mechanism. We then investigate some of its properties. This mechanism will be efficient, budget balanced and individually rational, without the need of a shared expectation of the types of the participants. Although the mechanism cannot be shown to be strictly incentive compatible, the participants will be unable to exploit this property without a proper expectation about the types of other participants.

5.1 Mechanism Outline

In this section, we introduce the Transfer Redistribution Mechanism (TRM), that achieves budget balancedness. We will show that it is possible to create an efficient budget balanced mechanism that is ex ante IC and IR, under Assumption 1, individual rationality of the current energy market. We enforce IC for most of the private information by using a mechanism with verification. For the particular piece of information that cannot be verified, there is no interim strategy for either truth telling or not truth telling. We speculate that for the intended setting truth telling will be a better strategy in expectation.

Definition 5.1.1 (Transfer Redistribution Mechanism). *The Transfer Redistribution Mechanism is a direct mechanism where:*

- Allocation is defined by the schedule that minimizes total supplier cost
- Transfers are defined as follows:

- Suppliers receive VCG transfers
- Consumers pay a fraction of the sum of supplier transfers, determined by the sum of their jobs weights

The mechanism is applied in the following steps. The suppliers make a claim of their cost functions and consumers make a claim of their jobs to the mechanism, we call these their *bids*. Based on the bids, the scheduling algorithm is used to create an optimal schedule for the jobs. The algorithm returns the schedule that minimizes the total cost for the suppliers. In the schedule, all jobs that reported feasible constraints are scheduled, and, following from Assumption 1, it is assumed that there always exists a supplier that can deliver the energy for finite cost.

To calculate the transfers to suppliers, the algorithm is ran again, for selected variations of the input, as will be explained in the next sections. The jobs are assigned a weighing factor, determined by the type of the job, and, in another weighing scheme, also the types of the suppliers. The sum of supplier transfers is divided over the jobs proportionally to their weights. The weights will be determined later in this chapter, when we examine two alternative weighing schemes.

For the majority of the job properties, it is possible to apply a *mechanism with verification* as introduced in Section 2.2.1. The properties that are suited for verification are:

- power consumption,
- job arrival time,
- job deadline, and
- job time-span.

Concerning arrival time and deadline, it is only possible to detect a job that reported these values too loose, when the resulting schedule plans the job execution at a moment that is not actually feasible for the job. It is not possible to verify that the reported values are too tight. Since this is the only available way left, in which the jobs can game the mechanism, it will be the aim of the two weighing schemes to incentivise the jobs to maximize the reports of deadline and minimize the reports of arrival time.

5.2 Transfers to Suppliers

Although the mechanism aims to incorporate supplier and consumer constraints, in order to create a unifying solution framework, the mechanisms of dealing with supplier bids and consumer bids are to some extent independent. Suppliers are only competing with other suppliers, and not with the jobs. Therefore, it would be convenient to be able to consider the suppliers' side and job side of the mechanism independently. What is required to do so, is to eliminate the uncertainty of suppliers about the job types, and instead create a façade of the jobs, that is invariant during the execution of the mechanism. This way, the suppliers no longer need to take into account the possibility that the jobs behaviour will deviate from their reported types.

To achieve this, the mechanism will only use the reported job types when calculating the supplier transfers. Since these are known at the start of the mechanism, and do not change during execution, the suppliers have reliable information about what they will be compensated for. Having their transfers depend solely on the reports of the jobs, and not on the actual executions, their side of the mechanism is similar to a single sided combinatorial auction. For a single sided combinatorial auction, we can use the VCG mechanism. The transfers to the suppliers are given in Equation 5.1.

$$\mathcal{T}_s = \Phi_{-s} - \Phi_{tot} + \Phi_s \quad (5.1)$$

Using VCG, the mechanism will be able to create a schedule that maximizes social welfare. To see this, first observe that the welfare of the jobs is independent of the schedule. This follows from the fact that all jobs will be executed within their constraints, combined with Assumption 2, job valuation is constant within the constraints. Second, the welfare of the suppliers is maximized when their costs are minimized. This is exactly the objective of the allocation function. As a result, the mechanism is efficient, independent of the job transfers. The mechanism also achieves the other desirable properties for the suppliers, see Lemma 5.2.1, inherited from the VCG mechanism.

Lemma 5.2.1. *The Transfer Redistribution Mechanism is IC for the suppliers, and truthful bidding is a dominant strategy.*

Proof of Lemma 5.2.1. The mechanism employs VCG transfers for the suppliers, so these properties are inherited from that mechanism. \square

5.2.1 Constraints on Transfers

The goal of the mechanism is to decrease the production costs for the suppliers. Ultimately, it is the consumers who are the cause for the production costs, and it will be the consumers that are paying for these costs. In order to align the objective of the mechanism, which is minimizing supplier cost, with the objective of the consumers, minimizing their transfers, there must be a positive effect on the consumer transfers from a decrease in supplier cost. This way, when the consumers maximize their utility, they will also minimize supplier cost. Suppose there is a job \hat{j} , whose properties possibly increase the total cost, and will never decrease the cost, when it replaces j in a problem instance. This can for example be a lack of flexibility.

Comparing the costs for the settings, one with \hat{j} , and one with j , we see the following relation:

$$\hat{\Phi}_{tot} \geq \Phi_{tot} \quad (5.2)$$

This relationship follows from the change in job properties, and from the allocation mechanism selecting the optimal schedule. For the correlation between transfers and costs, it is desired that transfer monotonicity is satisfied.

Definition 5.2.2 (Transfer monotonicity). *Transfer monotonicity is a property of the payments for the schedule. It is satisfied when the transfers are a monotone function of the cost*

of the schedule, formally:

$$\hat{\Phi}_{tot} \geq \Phi_{tot} \implies \hat{\mathcal{T}} \geq \mathcal{T} \quad (5.3)$$

However, the value of \mathcal{T} is determined by VCG transfers. While the left hand side holds in our context, unfortunately, monotonicity is not guaranteed by VCG transfers, so the right hand side is not naturally implied. Indeed, it is possible to construct an example where the relation is not satisfied. Such an example is given in Appendix A.

By enforcing constraints on the shape of the cost functions of the suppliers, it is possible to achieve monotonicity of the transfers for a single time unit. However, since the problem is all about job flexibility, monotonicity per time unit is not sufficient to achieve monotonicity over all. Therefore, it seems, VCG transfers cannot be used on the suppliers' side, because it cannot satisfy transfer monotonicity.

The alternative, to resort to a different mechanism on the suppliers' side, would require relaxation of the incentive compatibility requirement on the suppliers' side. Then, for example, a sealed bid first price auction could be used. If all suppliers follow their equilibrium strategy in that setting, the implication is satisfied. However, incentive compatibility is no longer guaranteed, and knowledge about the competition becomes an important factor in determining one's strategy.

So, although transfer monotonicity cannot be strictly satisfied using VCG, there are no good alternatives. However, in the problem context, the number of suppliers and jobs will be relatively large. As the number and diversity of jobs and suppliers increases, the atomicity of the jobs becomes less prominent, since by reordering the jobs a range of schedules can be created, each with only a slight cost difference from the others. In the example from Appendix A, it was a combination of atomicity and the alignment of the steps in the cost functions of the suppliers, that led to the non-monotonic behaviour. Intuitively, when the instances become larger and uncertainty grows, the exact conditions when untruthful reports benefit appear less obvious, and the best response strategy less clear. Even when a job can create its strategy with complete knowledge about the types of others, it still faces the challenge of calculating the strategy that would benefit it the most. In the following sections, because of this intuition, it is presumed that transfer monotonicity holds.

5.3 Transfers to Consumers

In the previous sections, the schedule and subsequently the supplier transfers were determined. The supplier transfers were created such that they are independent of the transfer scheme for the consumers. In this section we establish two variants for the transfers to consumers, again we consider this to be independent of the supplier transfers. In order to have this independence, the solution concepts that are considered are ex ante IR and BNIC. The supplier types are therefore exposed to the consumers only as probability distributions. The mechanism will only depend on properties that are independent of the underlying distribution.

Because the mechanism objective is to be budget balanced it is clear that the sum of transfers of the consumers must match the sum of transfers made to the suppliers. One way

to guarantee this is to assign a weight to each consumer, and divide the transfers according to the weights. The cost for one consumer is then calculated using this expression:

$$\mathcal{T}_i = \frac{w_i}{\sum_{k \in I} w_k} \sum_{s \in S} \mathcal{T}_s \quad (5.4)$$

How the jobs of a consumer contribute to its weight depend on the weighing scheme, and will be considered later. However, when a consumer owns one job only, then the transfer for that job and the consumer are identical. Note, that while a consumer owns several jobs, and is responsible for the reports about these jobs, he cannot use his influence on the reports of those jobs for his benefit. The consumer is not able to improve the resulting schedules by manipulation of the reports for the jobs he owns. The mechanism can be said to be coalition proof.

5.3.1 Power Proportional Weights

An easy way to establish weights for the consumers is to just set it equal to the total power consumption of its jobs, this scheme is defined in Definition 5.3.1. However simple this weighing might be, it already provides the desired properties of the mechanism, even in a strong solution concept.

Definition 5.3.1 (Power proportional weights). *The TRM weights for a consumer that are proportional to its power consumption are called power proportional weights (PPW). The weights are defined as:*

$$w_i^p = \sum_{j \in J_i} p_j s_j$$

Individual rationality is achieved as an *ex post* property following Lemma 5.3.2.

Lemma 5.3.2. *The TRM, using PPW, is ex post individually rational.*

Proof for lemma 5.3.2. A mechanism is *ex post* individually rational when the utility is greater or equal to zero for all types. In Section 4.3 the lower bound for a job's valuation was set to v_j^0 . It seems reasonable to assume that the valuation is proportional to the total power consumption of the job. So, the lower bound becomes $v_j^0 = q^0 p_j s_j$.

$$\begin{aligned} c_j &= \frac{w_i^p}{\sum_{j \in J} w_i^p} \mathcal{T} = \frac{p_j s_j}{\sum_{j \in J} p_j s_j} \mathcal{T} \\ &\leq \frac{p_j s_j}{\sum_{j \in J} p_j s_j} \sum_{j \in J} q^0 p_j s_j = p_j s_j q^0 \\ &\leq v_j \end{aligned}$$

Using this result, the utility of each job becomes greater or equal to zero.

$$\begin{aligned} u_j &= v_j - c_j \\ &\geq v_j - v_j = 0 \end{aligned}$$

□

Also, the mechanism, when using weights proportional to the power consumption, has as an *ex ante* strategy to report a bigger flexibility. The justification is given in Lemma 5.3.3.

Lemma 5.3.3. *In the TRM, using PPW, truthful reporting of arrival time and deadline is a weakly dominant strategy when transfer monotonicity holds.*

Proof of Lemma 5.3.3. The utility of the consumers is determined by their valuation and transfers. The valuation of each job is independent of the report. Furthermore, the weights contributions of the jobs are not influenced by the reports of arrival time and deadline, and other properties are verified.

The only influence on the consumer transfer is the total transfers for the suppliers. With transfer monotonicity it follows that the transfers are minimized by minimizing the cost of the schedule. The expected cost, and therefore the total transfer will never increase by reporting less flexibility. Therefore, truthful reporting is a weakly dominant strategy. \square

Although this weighing scheme indeed achieves incentive compatibility, the benefits for a consumer to report more flexible jobs are very limited. The decrease in total transfers that is achieved by extending one job's flexibility is, after all, shared with all other consumers. The consumer responsible for the decrease will only benefit a fraction, proportional to its weight. When the mechanism is deployed on larger scale, the effects for a single consumer might become negligible. Therefore, a stronger benefit is warranted, in order to improve the applicability of the mechanism. This will be the objective of the next section.

5.3.2 Cost Proportional Weights

In the previous section it was established that it is possible to have a mechanism that is IC and IR, when transfer monotonicity holds. However, using the previously derived weighing scheme, the benefits for a consumer of extending its jobs' flexibility are minimal. The problem in the previous approach is that transfers are divided among the consumers according to the energy their jobs require, while this may not be proportional to the increase in transfers they induce. In this section, a different weighing scheme is introduced, that trades incentive compatibility for a bigger benefit.

With VCG, the utilities represent the benefit for a player, reduced by the harm he causes to others. So while every player focuses on his own utility, in the end it is the social welfare that is maximized. The weighing scheme in this section is based on the philosophy of VCG transfers to achieve the same goal. In fact, the weights *are* the VCG transfers. But instead of charging the VCG transfers to the consumers directly, they now define the portion of the total transfers that a consumer has to pay. Except for a difference in notation, the consumer weights defined in Definition 5.3.4 are exactly similar to the transfers in Equation 4.1.

Definition 5.3.4 (Cost Proportional Weights). *The cost proportional weight (CPW) for the TRM is a weight for a consumer, proportional to the cost increase induced by that consumer's jobs. The weight is given by:*

$$w_i^M = \Phi_{tot} - \Phi_{tot}^{-i} \quad (5.5)$$

When a consumer minimizes its weight, it thereby minimizes its transfer. Therefore, the consumer has an incentive to minimize the difference in Equation 5.5. Minimizing this is equivalent to maximizing the social welfare.

As with VCG transfers, for each weight in the weighing scheme it is required to obtain the suppliers cost of the schedule when all the jobs of one consumer are removed. This means that we have to calculate the transfers to the suppliers again for every consumer. The algorithm for calculating the necessary values is shown in Algorithm 1. The Schedule(\cdot) method is used to calculate the optimal schedule, given the set of job and supplier bids in the parameters. The implementation of this method could for example be the MIQCP from Section 3.3. The Cost(\cdot) method calculates the overall cost of the schedule, i.e. the sum of the suppliers' costs $\sum_{s \in S} \Phi_s(\cdot)$, and Cost(\cdot, p) calculates the cost for supplier p in the schedule, this being equivalent to Φ_s .

The bulk of the work in the algorithm is done in the Schedule(\cdot) method, solving the NP-hard scheduling problem. Therefore, it is clear that this mechanism will be intractable. For the second price calculations it is necessary to compute a schedule as of the as the number of suppliers, to establish the supplier transfers. For the consumer weights, it is necessary to construct an additional number of schedules, as much as there are consumers. The complete mechanism is thus $O((n+m)S)$, where n and m are the number of consumers and suppliers respectively, and S is the complexity of the scheduling algorithm. As shown in Lemma 3.2.4, the scheduling problem is NP-hard, therefore, complexity of the scheduling dominates the complexity of the algorithm.

Algorithm 1: Price Calculation Algorithm

Input: Bids from consumers \hat{J} , and bids from the suppliers \hat{S}

Output: A schedule of execution, and a vector of prices

```

1 schedule  $\leftarrow$  Schedule( $\hat{J}, \hat{S}$ )
2  $T_{tot} \leftarrow 0$ 
3 for  $s \in S$  do
4   | sched  $\leftarrow$  Schedule( $\hat{J}, \hat{S} \setminus \{\hat{S}_s\}$ )
5   |  $T_s \leftarrow$  Cost(sched)  $-$  Cost(schedule)  $+$  Cost(schedule,  $s$ )
6   |  $T_{tot} \leftarrow T_{tot} + T_s$ 
7  $w_{tot} \leftarrow 0$ 
8 for  $i \in I$  do
9   | sched  $\leftarrow$  Schedule( $\hat{J} \setminus \{\hat{J}_i\}, \hat{S}$ )
10  |  $w_i \leftarrow$  Cost(sched)  $-$  Cost(schedule)
11  |  $w_{tot} \leftarrow w_{tot} + w_i$ 
12 for  $i \in I$  do
13  |  $T_i \leftarrow T_{tot} \cdot w_i / w_{tot}$ 
14 return (schedule,  $T$ )

```

The result returned by the algorithm contains the schedule that is to be executed. It also contains the transfers that have to be payed following this mechanism. With these transfers,

calculated using the weights as formulated in Equation 5.5, the mechanism is ex ante IR.

Lemma 5.3.5. *The TRM, using CPW, is ex ante individually rational for consumers under Assumptions 1 and 2.*

Proof of Lemma 5.3.5. A mechanism is ex ante individually rational when the expected utility, before the type is known, is greater than or equal to zero. Let \tilde{q} be the expected number of jobs owned by consumer i , and let n be the total number of jobs in the problem.

$$\begin{aligned}\mathbb{E}[u_i] &= \sum_{j \in J_i} (\mathbb{E}[v_j]) - \mathbb{E}[\mathcal{T}_i] \geq \tilde{q} \cdot v_j^0 - \sum_{j \in J_i} \frac{\mathbb{E}[\mathcal{T}]}{n} \\ &\geq \tilde{q} v_j^0 - \frac{\tilde{q} \cdot \sum_{k \in I} \sum_{j \in J_k} v_j^0}{n} = \tilde{q} \cdot v_j^0 - \tilde{q} \frac{n \cdot v_j^0}{n} = 0\end{aligned}$$

□

The mechanism is, however, not incentive compatible using these weights, notwithstanding transfer monotonicity. For example, Appendix B shows an example where the weight of a job decreases by reporting a tighter deadline, even though the total cost, and total transfers, increase.

The only strategy that would be profitable under all circumstances would be to acquire complete knowledge about the reported types of others, and calculate the best response. However, this is not an equilibrium strategy. When all participants behave this way, there exist problem instances that do not have a stable pure Nash equilibrium (Appendix C).

The consumers, therefore, do not have a strict equilibrium strategy. However, in practice, the jobs have to determine their strategy with incomplete knowledge. We conjecture that for the practical purpose of the mechanism, the best strategy is truthful reporting.

Conjecture 5.3.6. *Truth telling is a best response under uncertainty, for the TRM with CPW.*

In the next chapter, this conjecture will be investigated experimentally.

5.4 Mechanism Properties

In the previous three sections, piece by piece, the TRM was constructed. Several properties, for parts of the mechanism, have already been verified. In this section, the result of the mechanism as a whole is analyzed, to determine the global properties. Although these properties were intended to exist *by design*, it is still relevant to restate them here.

The important property for the mechanism maintainer is that he is not required to add value to the system for which he is running the mechanism. For the participants, it is preferred that no value needs to be destroyed. These properties are summarized in Lemma 5.4.1.

Lemma 5.4.1. *The Transfer Redistribution Mechanism is strongly budget balanced.*

Proof of lemma 5.4.1. The sum of transfers made to the suppliers is distributed over the consumers, according to their jobs weights. Therefore, the sum of transfers charged to the consumers is equal to what is paid to the supplier. The mechanism is budget balanced. \square

When Conjecture 5.3.6 holds, the reports from jobs and suppliers is truthful. The mechanism can then create the schedule using complete information. Since the allocation function maximizes social welfare, Lemma 5.4.2 follows.

Lemma 5.4.2. *When Conjecture 5.3.6 holds, the Transfer Redistribution Mechanism is efficient.*

Proof of Lemma 5.4.2. All jobs are always scheduled, so there is no influence of the mechanism on the welfare of the consumers, following Assumption 2. From Conjecture 5.3.6 it follows that the jobs report truthfully. The mechanism is incentive compatible for the suppliers. The mechanism, therefore, has complete knowledge about the job and supplier types. Since the suppliers costs are minimized by the resulting schedule, the social welfare is maximized. \square

Combining the results of the lemmas in this section and those in the previous sections, the following theorem follows as a unifying claim.

Theorem 5.4.3. *The Transfer Redistribution Mechanism is budget balanced, ex ante individually rational, and efficient given the reported types when the Conjecture 5.3.6 holds.*

Proof of Theorem 5.4.3. The proof for efficiency follows from Lemma 5.4.2, and budget balancedness follows from Lemma 5.4.1. From Lemma 5.2.1 we get ex post incentive compatibility for the suppliers, which implies ex ante incentive compatibility, and finally we have ex ante incentive compatibility for the consumers from Lemma 5.3.5. \square

The mechanism is not inherently incentive compatible. However, it is conjectured that the best response under uncertainty is truth-telling. In the next chapter some experimental results are analyzed to investigate this claim.

Chapter 6

Experiments

In the previous chapter we have established some theoretical lower bounds for the performance of the TRM. However, because the lower bounds are bounds in expectation, there is a possibility that in some instances these lower bounds are not met for all jobs. On the other hand, the bounds are lower bounds, so it is to be expected that the actual performance of the mechanism is better than these values. In this chapter we will make an analysis of the mechanism by applying it in an experimental setup.

Throughout this chapter, the consumers will own only one job. The weights and transfers are therefore associated equally with that job as with its owner. Sometimes the two will be used interchangeably, and when weights and transfers of a job are mentioned, they should be thought of as belonging to the owner of the job.

The scheduling algorithm used for the experiments is the MIQCP solver from the IBM ILOG CPLEX Optimizer package. The optimizer is set to continue optimizing until the best integer value is within 0.5% of the best non integer value. The returned schedule costs are therefore a 0.5% approximation of the exact solution.

6.1 Schedule Costs Reduction by Increased Flexibility

The aim of the mechanism is to reduce the production costs of energy by utilizing the flexibility on the consumers' side. In this experiment only the direct costs are considered, not the transfers. Therefore, this experiment verifies the benefits for the suppliers, and not necessarily for the consumers. However, when transfer monotonicity holds, the consumer transfers will decrease when the cost decreases. This experiment is to verify a relation between flexibility in the system and the cost of the resulting schedule.

The experiment starts with 30 problem instances, each with 18 jobs that have no flexibility. For the iterations of the experiment, the flexibility of each job is updated, while the arrival time and deadline of the jobs remain centered around the values in the original problem. The 30 instances the experiment starts with are generated at random. Then, each iteration the flexibility of every job is increased with a value of 0, 1 or 2 each with equal probability. The total flexibility in the system therefore increases.

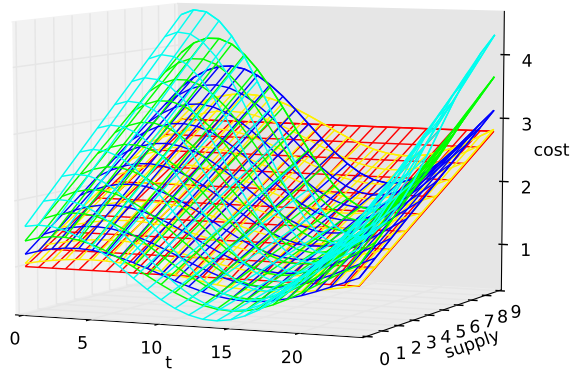


Figure 6.1: Cost profiles of suppliers, during cost experiments

In all the iterations, the cost profile of the five suppliers stays the same. A plot of the cost profiles is shown in Figure 6.1. The cost profiles are a linear function for each t , with perturbations following a sine-function, to introduce some differences in the costs over time.

The results of running the scheduling algorithm are shown in Figure 6.2. In the figure, the costs of each schedule have been normalized on the cost of the corresponding problem instance without added flexibility. It is clear that there exists a trend of decreasing costs, when the flexibility increases.

Benefits of the flexibility increase range from a few percent for instances that by chance already had a cheap configuration, to over 20 percent for instances that had an unfortunate initial distribution. The average benefit, exceeds 10 percent when the average flexibility is 3. Therefore, this experiment proves that adding flexibility to the system has a positive effect on production costs. The magnitude of the effect would naturally vary with different cost profiles for the suppliers.

6.2 Consumer Transfer Benefits With Power Proportional Weights

The experiments in the previous section show that more flexible jobs yield a cheaper schedule. When schedule cost and transfers are correlated, this will also yield lower transfers for the consumers. In this experiment, the effect of a flexibility increase for one job is studied, when the other jobs remain the same. The weighing scheme that is used in this section is the

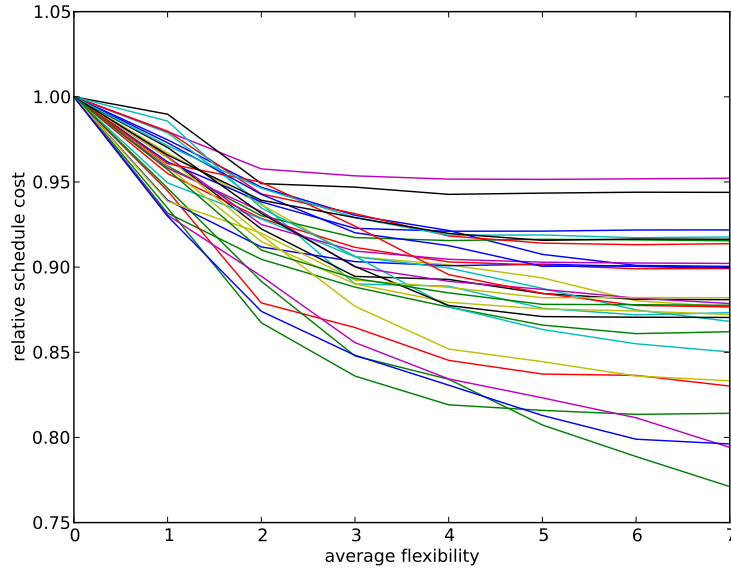


Figure 6.2: Relative cost with added flexibility

power consumption proportional weight (PPW) from Section 5.3.1. The expected results are a slight decrease in transfers.

In this experiment 25 jobs and 6 suppliers are created randomly. The transfers for this instance are calculated. Then, one by one, the flexibility of the jobs is increased a certain amount, and the transfers of the new instance calculated. The transfers of the jobs depend on the total transfer and the job's power consumption. The latter does not change during this experiment, so the only factor of influence is the total transfers. These values are plotted in Figure 6.3.

For some consumers the relative cost exceeds 1. However, because the scheduling algorithm produces an approximation, the observed increases fall within the uncertainty of the transfer ratio, and no conclusions can be drawn from them. The figure, therefore, shows no significant increase in the transfers of jobs with an increased flexibility. This is in line with Lemma 5.3.3, truthful reporting arrival time and deadline is a weakly dominant strategy. The benefits are, however, also small, since the benefit is shared among all consumers.

6.3 Consumer Transfer Benefits With Cost Proportional Weights

The benefit for a consumer to increase flexibility is small when the PPW scheme is used, as is shown in the previous section. In this section, the experiment is repeated, however the consumer transfers are calculated using the cost proportional weights (CPW) scheme. The aim of the experiments is to support Conjecture 5.3.6, consumers improve their utility by

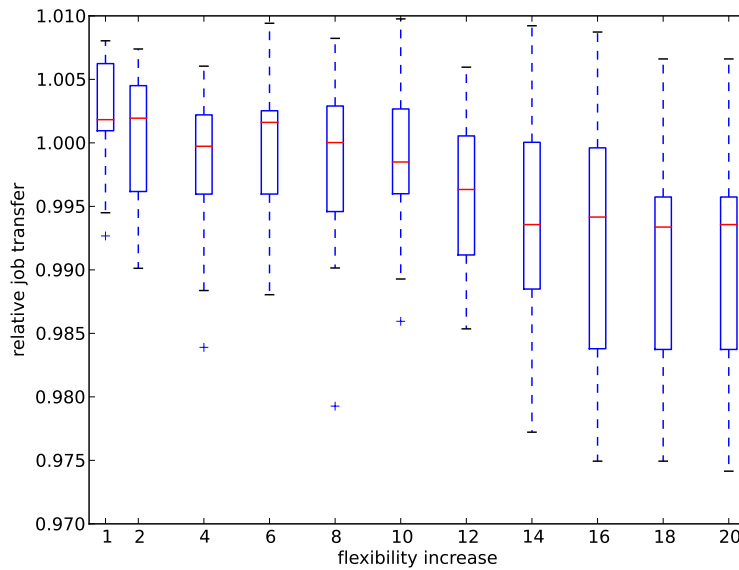


Figure 6.3: Normalized costs of jobs as flexibility increases, using power consumption proportional weights. Each box shows the cost for 25 jobs.

reporting the arrival time and deadline of their jobs truthfully. It is expected that the benefits of increased flexibility are bigger than in the previous experiment. This follows because the decrease in total transfers to the suppliers is similar, but by extending flexibility of its jobs, a consumer now potentially also decreases its share of the total transfer.

The same experimental setup is used as described before. The only difference is the weighing scheme, resulting in different transfers for the consumers. These transfers are plotted in Figure 6.4.

The graph shows a significant decrease of the average transfer, thereby supporting the expectation that consumers get a benefit for reporting more flexibility. The average transfer drops almost 20% when the jobs have maximum flexibility. But, when the jobs can be scheduled on half the time units in the problem, the decrease in transfer is already 10%. These results support Conjecture 5.3.6.

On the other hand, it appears that four jobs do not benefit at all, since the top of the box's whiskers remains at 1, and for a flexibility increase of 14 it is visible that there are four jobs with a relative cost around 1. Possibly this could be because their time window already contained their optimal moment in the schedule, so the transfer for them does not change by adding more flexibility.

A few points in the graph are above the relative cost of 1.00. This suggests that the transfers increase by the report of more flexibility. However, this conclusion cannot be drawn so easily. The scheduling algorithm used in this experiment approximates the exact value by an approximation of 0.5%. The uncertainty for the transfers are the result of

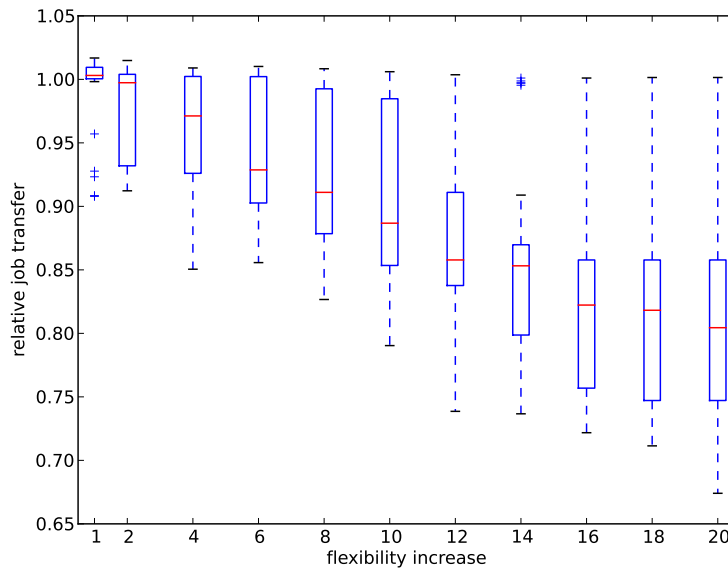


Figure 6.4: Normalized costs of jobs as flexibility increases, using cost proportional weights.

summing and dividing a lot of schedule-costs. Therefore the recorded increases, the biggest being 1.68%, fall within the uncertainty of the transfers.

6.4 Consumer Transfer Benefit and Initial Flexibility

In this section we are interested in the effect the flexibility of the other jobs has on the transfer benefits for a job extending its flexibility. In this experiment we consider one hand-crafted problem, where 15 jobs are spread evenly across the 25 time units in the problem. Six variations are constructed, varying the initial flexibility of all jobs in the instance between 0 and 5. For each of these six instances, the flexibility of each job in turn is increased further, up to an additional 15 units of time. The result would be 30 graphs, but instead Figure 6.5 shows averages, grouped per variation. The 5 suppliers remain the same throughout the experiment.

The graphs for the initial flexibilities from 0 to 4 show a trend that the benefit for an increased flexibility is bigger, when the other jobs also have more flexibility. This is not unexpected, since more flexibility for the other jobs enables more schedules, thereby making it possible to find one that decreases the cost and, consequently, transfers.

The graph for an initial flexibility of 5 has a less steep decline than that of 4, contradicting this trend. Indeed, the *relative* transfer benefit for the jobs is not as big as that for those in the instance with an initial flexibility. However, the absolute transfer averages when all jobs have their initial flexibility, i.e. where the flexibility increase is 0, are significantly

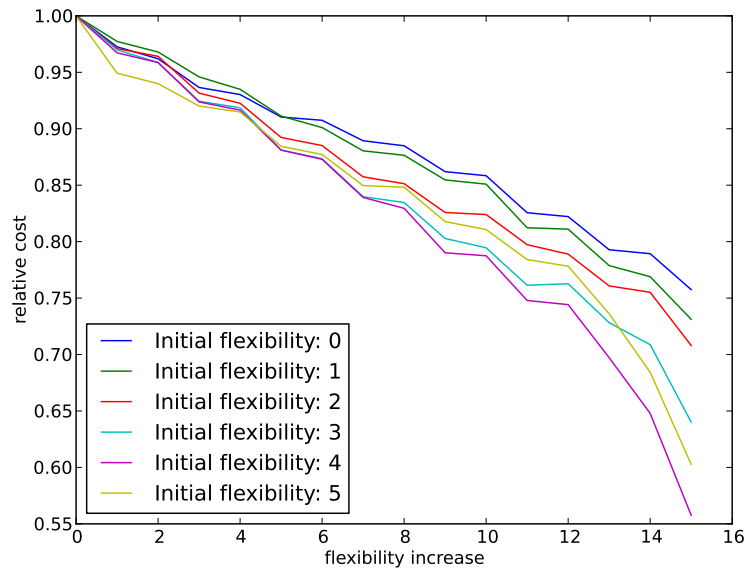


Figure 6.5: Cost decrease as flexibility is added to the system. Each point is the average of 15 schedules, resulting from increasing the flexibility for one job.

lower. This is also explained easily, since the initial instance has enough flexibility in itself to create a cheaper schedule. The relative benefit for one job to get additional flexibility thereby decreases, but in absolute terms the transfers are lower than those of the jobs in the graph with an initial flexibility of 4.

All in all, the size of the transfers decreases when the flexibility of the jobs in the problem increases. However, the costs for production of energy must be compensated for, so at some point the transfers can decrease no more. As the overall flexibility in the system increases, the benefit of adding one additional time unit flexibility becomes smaller.

6.5 Scale and Computability

From running the experiments it becomes clear that the runtime of the mechanism becomes an issue very quickly. This was to be expected from the theoretical complexity analysis. The dependence of calculation time for the scheduling algorithm on the parameters of the problem is shown in Figure 6.6.

Interestingly, only the number of jobs and flexibility of jobs have a real influence on the runtime of the scheduling algorithm. The other properties do not express a trend as strong as those two. The correlation between runtime and time span is particularly surprising. However, the peaks on either end appear where the jobs span almost the entire duration of the schedule, or almost none. From the scheduling algorithm's perspective this might be similar in the sense that it is either scheduling a bunch of small jobs, or a bunch of small

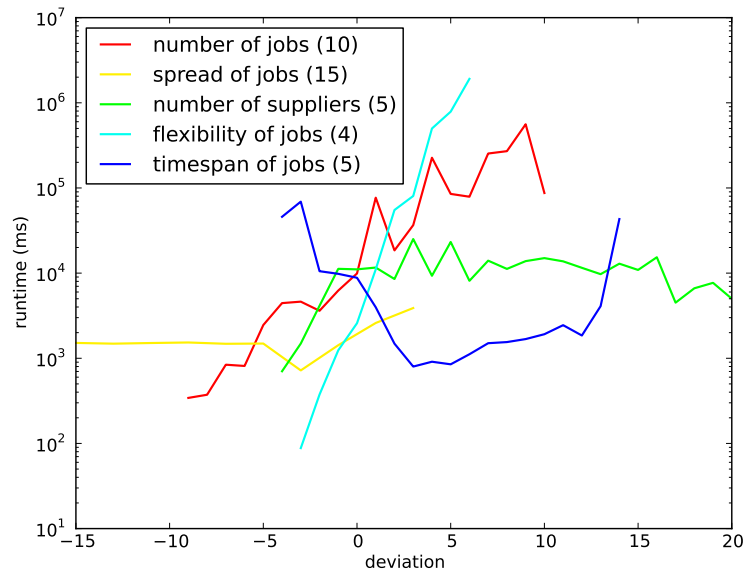


Figure 6.6: The effect of different problem parameters on runtime. For each graph, one parameter is varied, while the others remain at their base value (shown in brackets). The x-axis shows how much the parameter is changed relative to its base value.

gaps, which is, apparently, both difficult.

The trend of runtime to increase as the flexibility or number of jobs increase can be explained by a single proportionality. When the runtime is proportional to the number of schedules possible, this explains both.

This suggests that, in an effort to reduce runtime of the scheduling algorithm, one can focus on the reduction of the number of schedules possible. For example, the number of jobs can be reduced by grouping the jobs into clusters. The jobs that are to be clustered should have the same time properties, i.e. the same span, availability and deadline. By considering the cluster as one, the number of ways the jobs in the cluster can be scheduled reduce, so the problem becomes simpler.

By grouping the jobs, of course one also eliminates possible schedules. Therefore, care should be taken to see whether this does not influence the other properties of the mechanism. For example, Nisan [21] shows that the a VCG mechanism, when the optimal outcome is replaced with an outcome produced by an approximation or heuristic algorithm, is no longer necessarily truthful.

Chapter 7

Conclusions and Future Work

The goal of this research is to reduce the energy production costs. Therefore we exploit the flexibility of jobs on the consumers' side, which is assumed to be already present, however not used in current practice. Yet, before the flexibility can be utilized, it is necessary to expose this flexibility of the jobs involved. Therefore, the mechanism should be incentive compatible for the consumers reporting the job types.

Matching suppliers with consumers is an extended form of bilateral trade. From literature, it is known that for bilateral trade it is not generally possible to achieve an efficient, budget balanced and incentive compatible mechanism. However, the valuation of consumers for their jobs' execution is assumed to be constant for all feasible executions, and all jobs are always executed. As a result, the social welfare does not depend on the valuation of the jobs. Still, these restrictions are not enough to counter the impossibility result. In this thesis, it is shown that no Groves class mechanism exists that achieves budget balance for the problem. Furthermore, we show that the AGV mechanism cannot be used to create an individually rational mechanism under the previously mentioned assumptions.

To improve on this result, the applicability of a mechanism with verification is investigated. When a mechanism with verification can be used, the incentive compatibility requirement can be enforced afterwards, and no longer needs to be a property inherent to the mechanism. In the problem at hand, verification can indeed be used and enforces truthful reports for most of the properties of the jobs.

The properties that cannot completely be enforced by a mechanism with verification are the arrival time and deadline of the jobs. These properties can only be verified when false reporting results in a schedule that is infeasible for a job, or by means outside the scope of the mechanism. Declaration of an arrival time and deadline that are narrower than what is feasible cannot be verified. Therefore the mechanism must offer an incentive for consumers to report the arrival time and deadline of jobs as wide as possible.

After all, not all properties of the jobs can be verified, a mechanism with verification turns out not to be sufficient to remove the requirement of the mechanism to be inherently incentive compatible. Therefore, the mechanism is still bound by the impossibility result.

Budget balancedness and individual rationality are the properties that are most valued for a mechanism for the energy market. Because if it is not budget balanced value must be removed from the mechanism, and if it is not individually rational, consumers and suppliers

have no incentive to participate in the mechanism. The mechanism must thus drop incentive compatibility and, consequently, efficiency.

The mechanism achieves budget balancedness by dividing the total transfer made to suppliers over the consumers. The share of transfers that is assigned to a consumer depends on its job types. The transfers for the suppliers' side are determined by a VCG mechanism, thereby achieving truthful reports. However, with VCG transfers it is not guaranteed that the supplier transfers are a monotone function of the schedule cost.

We introduce the notion of *transfer monotonicity* that indicates whether the sum of transfers to suppliers in a mechanism are a monotone function of the cost of the optimal schedule. With the requirement that transfer monotonicity holds, the power proportional weighing scheme for the mechanism is efficient, incentive compatible, individually rational and budget balanced. From the experiments in Section 6.2 it becomes clear, however, that the benefits for consumers to report about their jobs truthfully are quite limited. Therefore, when the valuation for execution of a job within its time window is not strictly constant, unlike what was assumed, the benefits for the truthful reporting are eradicated.

To improve on this result the *cost proportional weighing* scheme is introduced. The experimental results in Section 6.3 show that the benefits obtained by increasing the reported flexibility are indeed larger. However, the mechanism is no longer incentive compatible using this weighing scheme, regardless of transfer monotonicity.

We conjecture that in the practical setting, truthful reporting is still the best strategy for the jobs. This is supported by the experiments, since the benefits that are observed from reporting more flexibility are significantly larger than the losses, if they can even be identified as such.

7.1 Future work

The scheduling algorithm that is presented and used for the experiments in this thesis uses a mixed integer quadratically constrained program solver. This solver does not return the exact value, but instead an approximation within a defined ratio from the exact solution. As a result, incentive compatibility might be compromised, as was observed by Nisan [21] for the VCG mechanism working with approximation algorithms. The effects of the use of an approximation algorithm should be verified for this problem, and if incentive compatibility is compromised a different scheduling algorithm is necessary before the mechanism can be deployed. Furthermore, the DSP instances that can be solved are limited to those in the mixed integer quadratically constrained program class. When more generic supplier cost functions have to be modeled, a different scheduling algorithm is required.

If a scheduling algorithm is found for which the mechanism remains incentive compatible, and that is able to solve problems with cost functions that are found in practice, Conjecture 5.3.6 must be verified for practical settings. After all, for the cost functions of real world suppliers, transfer monotonicity might not be satisfied. This affects the incentive compatibility for the suppliers.

Finally, the problem solved with the TRM is a static problem, the consumers and suppliers do not change during its execution. Therefore, the consumers and suppliers are required

to report their private information before the schedule starts. But when the mechanism is repeated time after time, the consumers and suppliers expose their private information to the other participants. Both Athey [2] and Cavallo [3] have researched auction mechanism in a dynamic setting. They observe that new possibilities for strategic behaviour emerge in a dynamic setting. The effect on the incentive compatibility by use of the TRM in a dynamic setting should also be investigated before it can successfully be applied.

Bibliography

- [1] K.J. Arrow and Harvard University. Harvard Institute of Economic Research. *The property rights doctrine and demand revelation under incomplete information*. Institute for Mathematical Studies in the Social Sciences, Stanford University, 1977.
- [2] S. Athey and I. Segal. An efficient dynamic mechanism. *Unpublished manuscript, Harvard University*, 2007.
- [3] R. Cavallo. *Social Welfare Maximization in Dynamic Strategic Decision Problems*. PhD thesis, Citeseer, 2008.
- [4] E. Çerezci and F. Gökpınar. A comparison of three linear programming models for computing least absolute value estimates. *Hacettepe Journal of Mathematics and Statistics*, 34:95–102, 2005.
- [5] E.H. Clarke. Multipart pricing of public goods. *Public choice*, 11(1):17–33, 1971.
- [6] C. d’Aspremont and L.A. Gérard-Varet. Incentives and incomplete information. *Journal of Public economics*, 11(1):25–45, 1979.
- [7] A. Gibbard. Manipulation of voting schemes: a general result. *Econometrica: Journal of the Econometric Society*, pages 587–601, 1973.
- [8] T. Groves. Incentives in teams. *Econometrica: Journal of the Econometric Society*, pages 617–631, 1973.
- [9] HK Hutting, JW Cleijne, and K. Sustainable. The price of large scale offshore wind energy in a free electricity market. In *1999 European Wind Energy Conference: wind energy for the next millennium: proceedings of the European Wind Energy Conference, Nice, France, 1-5 March 1999*, page 399. Earthscan/James & James, 1999.
- [10] D.S. Johnson, A. Demers, J.D. Ullman, M.R. Garey, and R.L. Graham. Worst-case performance bounds for simple one-dimensional packing algorithms. *SIAM Journal on Computing*, 3:299, 1974.

- [11] A. Kirby. *Climate in peril: a popular guide to the latest IPCC reports*. United Nations, 2009.
- [12] R. Kolisch and A. Sprecher. Psplib-a project scheduling problem library* 1:: Or software-orsep operations research software exchange program. *European Journal of Operational Research*, 96(1):205–216, 1997.
- [13] V. Krishna. *Auction theory*. Academic press, 2009.
- [14] V. Krishna, M. Perry, Center for Rationality, and Interactive Decision Theory. *Efficient mechanism design*. Hebrew University of Jerusalem, Center for Rationality and Interactive Decision Theory, 1997.
- [15] L. Lutzenhiser. Social and behavioral aspects of energy use. *Annual Review of Energy and the Environment*, 18(1):247–289, 1993.
- [16] P.R. Milgrom and R.J. Weber. A theory of auctions and competitive bidding. *Econometrica: Journal of the Econometric Society*, pages 1089–1122, 1982.
- [17] R.B. Myerson. Incentive compatibility and the bargaining problem. *Econometrica: journal of the Econometric Society*, pages 61–73, 1979.
- [18] R.B. Myerson and M.A. Satterthwaite. Efficient mechanisms for bilateral trading* 1. *Journal of economic theory*, 29(2):265–281, 1983.
- [19] J.F. Nash. Equilibrium points in n-person games. *Proceedings of the National Academy of Sciences of the United States of America*, 36(1):48–49, 1950.
- [20] N. Nisan. *Algorithmic game theory*. Cambridge Univ Pr, 2007.
- [21] N. Nisan and A. Ronen. Computationally feasible vcg mechanisms. In *ACM Conference on Electronic Commerce*, pages 242–252. Citeseer, 2000.
- [22] N. Nisan and A. Ronen. Algorithmic mechanism design. *Games and Economic Behavior*, 35(1/2):166–196, 2001.
- [23] D.C. Parkes and L.H. Ungar. Iterative combinatorial auctions: Theory and practice. In *Proceedings of the National Conference on Artificial Intelligence*, pages 74–81. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2000.
- [24] W. Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *The Journal of finance*, 16(1):8–37, 1961.

Appendix A

Non-monotonicity of VCG

The transfers in the VCG mechanism are not monotone in the efficiency of the outcome value. This can result in some unexpected behaviour. For example, in an ordinary auction, it is possible that some goods are sold with zero transfer, even though the valuation for the goods is non-zero for all participants. In the reversed auction too, there exists some unexpected behaviour. In the example below, the total VCG transfers will decrease, as a result of *increasing* total power demand in the system.

Take $n = 10$ suppliers of type I, their cost function given by Equation A.1. The values a and b obey the following relation: $0 < a \ll b$.

$$\phi_{s_I}(p) = \begin{cases} a & \text{if } p \leq u, \\ 2a & \text{if } u < p \leq \frac{n}{n-1}u, \\ 4a & \text{if } \frac{n}{n-1}u < p \leq v, \\ b & \text{otherwise.} \end{cases} \quad (\text{A.1})$$

One other supplier, of type II, is also present in the system. This supplier has a cost function given by Equation A.2.

$$\phi_{s_{II}}(p) = \begin{cases} 4a & \text{if } p \leq v, \\ b & \text{otherwise.} \end{cases} \quad (\text{A.2})$$

A graph of the cost function of both types of suppliers is shown in Figure A.1.

The problem starts with n jobs, each with a power consumption of u . The cost of the schedule in this situation is $n \cdot a \cdot u$. To establish the VCG transfers, one supplier at a time is removed from the problem, and the cost of the schedule with one supplier removed is calculated. Since the suppliers contributing to the optimal schedule are all type I, it suffices to do the calculation for a type I supplier only.

In this setting with a type I supplier removed, the problem consists of n jobs and n suppliers. With the assumption that $\frac{4}{3}u > v$ ¹, the cost of the schedule in this setting is

¹ This is necessary to ensure that the optimal schedule equals distributing the load evenly over all type I suppliers. Without this restriction, a couple of suppliers could take a load bigger than $\frac{n}{n-1}u$, thereby leaving the other suppliers free to produce u for the cost of $a \cdot u$. The actual fraction of v over u depends on the other parameters of the problem, such as n and ϕ_{s_I} .

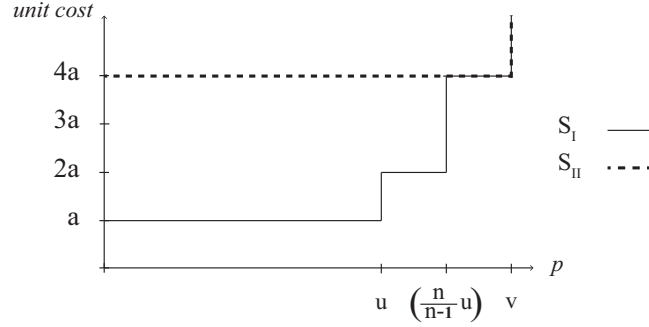


Figure A.1: Unit cost for the two types of suppliers as a function of demanded production.

$n \cdot 2a \cdot u$. Using these costs, the transfers for the suppliers can be calculated. For one supplier, the transfer is:

$$\begin{aligned}
 \mathcal{T}_s &= \Phi_{-s} - \Phi_{tot} + \Phi_s \\
 &= (n \cdot 2a \cdot u) - (n \cdot a \cdot u) + a \cdot u \\
 &= (n - 1)a \cdot u
 \end{aligned} \tag{A.3}$$

In the other setting, the jobs increase their power consumption from u to v . Let u and v obey the relation: $\frac{10}{9}u < v$, so it is not beneficial to assign the power demand to the type I suppliers only. Then, with a power consumption of v per job, the optimal schedule is to divide the power demand equally over the $n + 1$ suppliers. The cost of the schedule becomes $n \cdot 4a \cdot v$. Removing one supplier, and recalculating the schedule gives a cost too of $n \cdot 4a \cdot v$. The transfers in this setting are given in Equation A.4.

$$\begin{aligned}
 \hat{\mathcal{T}}_s &= n \cdot 4a \cdot v - n \cdot 4a \cdot v + 4a \cdot v \\
 &= 4a \cdot v
 \end{aligned} \tag{A.4}$$

The difference in transfer for one supplier, therefore, is $(n - 1)a \cdot u - 4a \cdot v$. With the assumption that $\frac{4}{3}u > v$, this difference is a positive value for $n = 10$. This would yield a positive difference per supplier, so the total transfer would be a factor 10 higher. Therefore the total transfers in this example *decrease* by increasing power demand from u to v , hence showing the non-monotonicity of the VCG transfers.

Appendix B

Non-IC example for TRM using CPW

In this appendix, an example is shown of a setting where a job can decrease its transfers by reporting a more restrictive deadline. The example consists of two units of time. One job, j , can be executed in either unit of time, while the other n jobs must be executed in the first unit of time. When j claims it too can only be executed in the first unit of time, let this be denoted by \hat{j} , its transfers decrease, while the total transfers do not.

Assume that the total transfers in the two settings do not change that much. Then, the weight of the job dominates their individual transfer. The example below shows a situation where a job's share of the summed weights becomes smaller, by reporting a tighter deadline.

In the first time unit, we find a number of suppliers, that have an aggregated cost function given by Equation B.1. The shape of the function for higher values is not of interest for this example, but should be shaped such that the transfers in both settings do not differ very much.

$$\phi_{s_i}(p) = \begin{cases} a & \text{if } p < 12, \\ \frac{3}{2}a & \text{if } p = 12, \\ \dots & \text{otherwise} \end{cases} \quad (\text{B.1})$$

For the second unit of time, two suppliers can provide energy, both for a cost of a per unit energy.

There exist 10 jobs in the instance with a power demand of 1, that can only be scheduled in the first unit of time. The flexible job, job j , can be scheduled in both the first and the second unit of time. It has a time span of 1 unit of time, and a power demand of 2.

In order to calculate the weights, it is necessary to find the schedule in five settings. These schedules are shown in Figure B.1. Two schedules with all jobs included, one with a flexible j , and one with an inflexible j . Two schedules for one of the other jobs removed. Finally, the schedule for when j is removed is identical for both reports of j , so it needs only to be calculated once.

Using the values of the example, job j gets the weight fraction given by Equation B.2.

$$\frac{w_i}{\sum_k w_k} = \frac{2a}{10a + 2a} = \frac{1}{6}a \quad (\text{B.2})$$

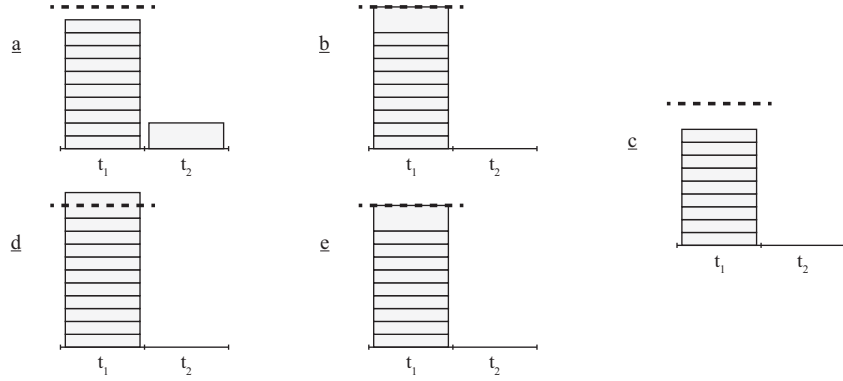


Figure B.1: Optimal schedules for: all suppliers, small jobs and j (a), all suppliers, all but one small jobs and j (b), all suppliers and small jobs (c), all suppliers, small jobs and \hat{j} (d), all suppliers, all but one small jobs and \hat{j} (e). The dotted line indicates the limit below which the unit cost is a .

When the j is replaced by hj , the weight fraction is given by Equation B.3.

$$\frac{w_j}{\sum_k \hat{w}_k} = \frac{8a}{10 \cdot 7a + 8a} = \frac{4}{39}a \quad (\text{B.3})$$

The result shows that the fraction of total transfers has decreased for job j , by reporting a tighter deadline. When the total transfers does not increase too much from reporting a tighter deadline, the job is able to decrease its cost. The increase in total transfers should be smaller than the decrease in weight fraction.

When the transfers in both situations are given by the total cost of the schedule containing all jobs, the weight fraction decrease would not be canceled by the increase in total transfers. The transfer for j would become $2a$, and for \hat{j} it is $1.8a$.

Note, that using these values for the transfers would be exactly the result when a first price auction is used for the suppliers, and the suppliers report the same cost function. Therefore, this example also shows that incentive compatibility cannot be restored by using a first price auction on the suppliers' side, instead of the VCG payments that the mechanism employs.

Appendix C

No Stable Nash Equilibrium for TRM, Using CPW

In this appendix, an example will be presented showing a problem instance where a stable Nash equilibrium for TRM, using CPW does not exist. The example has two units of time, and three jobs. Let the jobs be labeled j , k and l , then the power demand for the jobs is, respectively, 3, 2 and 2.

The cost function of the suppliers are identical in both units of time. The cost function for one of the suppliers is given in Equation C.1. This supplier is producing all the energy in the optimal schedule. The other suppliers must have a cost function such that the transfers remain similar for all three mutations of the instance.

$$\phi_s(p) = \begin{cases} a & \text{if } p \leq 4, \\ 1.1a & \text{if } 4 < p \leq 5, \\ 2a & \text{otherwise.} \end{cases} \quad (\text{C.1})$$

Both units of time are feasible moments of execution for all three jobs. However, by reporting only one possible unit of time for their execution, each job intends to minimize its transfer. Here, it is only shown that, for every combination of reports, one job is able to decrease its weight fraction by changing its reported type. Figure C.1 shows the schedules that result from the reports, ignoring symmetric equivalent schedules. The costs of the schedules, and consequently weights, for the jobs is shown in Table C.1.

In Figure C.1, the arrows indicate the order in which the schedules follow each other up. The job changing its report for each of the transitions is, from (a) to (c), job j , k and l , in that order. After three transitions the jobs have swapped places, but as the suppliers in both time units are identical, the costs and weights in (d) are identical to (a), and job j will change its report again.

	Φ_{tot}	Φ_{-j}	Φ_{-k}	Φ_{-l}	$\frac{w_j}{\sum_i w_i}$	$\frac{w_k}{\sum_i w_i}$	$\frac{w_l}{\sum_i w_i}$
a	7	3	2	2	.43	.29	.29
b	27	10	8.5	8.5	.37	.32	.32
c	12	5.5	2	4.5	.46	.17	.38

Table C.1: The costs and weights associated with the different schedules in this example.

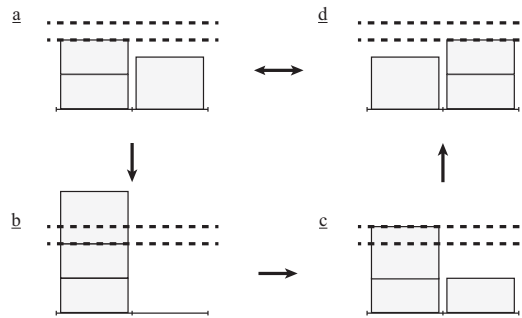


Figure C.1: By unilateral change of reported type, one job can always decrease its transfer. As figure (a) and (d) are symmetric, the process repeats itself after three mutations.