Institut für Informatik
der Technischen Universität München
Lehrstuhl für Informatik XVIII

Compact Bidding Languages and Supplier Selection for Markets with Economies of Scale and Scope

Stefan Schneider

Vollständiger Abdruck der von der Fakultät für Informatik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitzender:		Univ.-Prof. Dr. Helmut Seidl
Prüfer der Dissertation:		
	1.	Univ.-Prof. Dr. Martin Bichler
	2.	Univ.-Prof. Gudrun J. Klinker, Ph.D.

Die Dissertation wurde am 07.03.2011 bei der Technischen Universität München eingereicht und durch die Fakultät für Informatik am 30.07.2011 angenommen.

Abstract

Preference elicitation is a fundamental problem in single-unit combinatorial auctions, but it becomes prohibitive even for small instances of multi-unit combinatorial auctions. The bidders cannot express their preferences exactly as this would take a huge number of bids, typically leading to inefficient allocations.

Hence, markets with economies of scale and scope require more compact and yet expressive bidding languages. In this thesis, we propose an expressive bidding language allowing bidders to describe the characteristics of their cost functions. Bidders in these auctions can specify various discounts and markups, and specify pricing rules as logical functions. Finding the optimal allocation given these pricing rules is a strongly NP-hard optimization problem and we propose a mixed integer program to solve it.

Based on field data, we introduce a multi-item cost function and provide extensive computational experiments to explore the computational burden and the impact of different language features on the computational effort and total spend of the auctioneer. In addition, we explore characteristics of the knowledge representation of the bidding language.

Zusammenfassung

Kombinatorische Auktionen bieten die Möglichkeit Synergieeffekte zu nutzen, indem sie mehrere Güter in einer Auktion zusammenfassen. In einer kombinatorischen Auktion mit einer nicht trivialen Anzahl von Gütern ist allerdings bereits das Abfragen der relevanten Wertigkeiten problematisch, da die Bieter nicht so viele Einzelgebote abgeben können wie nötig wären, um die effiziente Allokation sicher zu bestimmen. Dieses Problem verschärft sich weiter, wenn pro Gut nicht nur eine Einheit versteigert wird, sondern auch Mengenrabatte berücksichtigt werden sollen.

Daher benötigen derartige Märkte kompaktere, und dadurch beherrschbare, aber dennoch erschöpfende Bietsprachen. In der vorliegenden Arbeit schlagen wir eine derartige Sprache vor, welche es den Bietern auf vielfältige Weise erlaubt ihre Kostencharakteristika in kompakter und intuitiver Form auszudrücken. Dazu können sie verschiedenartige Preismodifikatoren verwenden, welche ihrerseits von flexiblen und logisch kombinierbaren Bedingungen abhängen. Diese Bedingungen können sich dabei sowohl auf die gekaufte Menge als auch den erzielten Umsatz beziehen. Das Bestimmen der optimalen Allokation aus diesen komplexen Preisstrukturen ist erwiesenermaßen NP vollständig, und von daher in der Berechnung potentiell sehr aufwändig. Wir schlagen daher ein gemischt-ganzzahliges Optimierungsproblem vor, mit dessen Hilfe diese berechnet werden kann.

Abschließend evaluieren wir unseren Ansatz mit Hilfe eines selbstentwickelten Kostenmodells, dessen Charakteristika auf Echtdaten beruhen, welche wir in Feldversuchen sammeln konnten. Als Hauptergebnis sondieren wir, welche Problemgrößen in akzeptabler Zeit lösbar sind. Wir untersuchen dabei den Einfluss verschiedener Elemente der Bietsprache auf Berechnungsaufwand und Einsparungsmöglichkeiten gegenüber einfacheren Ansätzen.

Acknowledgments

Zuerallerst will ich mich bei Martin Bichler für die Gelegenheit bedanken, ein solch interessantes und spannendes Forschungsprojekt übernehmen zu dürfen. Die freizügig und gestaltende Arbeit daran hat mir viel Spass bereitet und ich habe vieles dabei gelernt.

A special thanks goes to Kemal Guler and Mehmet Sayal for a very inspiring collaboration and the great time in California.

Bei meine Kollgen Georg Ziegler, Pasha Shabalin, Tobias Scheffel, Jürgen Wolf, Alexander Pikovsky, Christian Markl, Oliver Hühn und Christian Hass möchte ich mich für die Diskussionen und wertvolle Anregungen bedanken.

Ich möchte Felix Schmid dafür danken sein Zimmer belagern zu dürfen und Maria Schmid für Ihre treibende Geduld.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Procurement is one of the key activities in the supply chain and occupies a very important role in the overall performance of a company. With margins dwindling, due to increased competition in nearly all industries, it becomes even more indispensable to firms to minimize their procurement cost by procuring at the best prices.

Economies of scale and scope describe key characteristics of a supplier's production function that influence the prices on procurement markets. Whereas economies of scale primarily refer to efficiencies associated with supply-side changes, such as increasing or decreasing the scale of production of a single product type, economies of scope refer to efficiencies associated with demand-side changes, such as increasing or decreasing the scope of marketing and distribution, of different types of products.

Auctions as an advanced mechanism for trading have successfully been used and eased negotiation in many environments. One of their main features is the ability to find prices that reflect the market situation closely by coordinating the competition.

1.1 Motivation

Using auctions in procurement is therefore a logical step and simple split-award auctions are regularly used in practice for multi-item, multi-unit negotiations. There the best bidder gets a predefined larger share of the volume for a particular good and the second best bidder gets the remaining share. This is done to on the one hand assure supply if one supplier is unable to fulfill his contract and on the other hand a minimal supplier pool in the long run. With significant economies of scale, suppliers face a strategic problem in these auctions. Since there is uncertainty about which quantity they will get awarded, they might speculate and bid less aggressively based on the unit cost for the smaller, more expensive, share. In other words, simple split award auctions do not allow suppliers to adequately express economies of scale.

In the recent years, driven by the new possibilities of the Internet, a growing literature is devoted to the design of optimization-based markets (aka. smart markets) (Gallien and Wein 2005), and in particular to combinatorial auctions, where bidders are allowed to submit bids on packages of discrete items (Cramton et al. 2006). The promise of these mechanisms is that by allowing market participants to reveal more comprehensive information about cost structures or utility functions, this can drastically increase allocative efficiency and lead to higher economic welfare. Unfortunately, the matching of complex preference profiles typically leads to hard optimization problems. The literature in this field is typically focused on multi-item but single-unit negotiations and respective auction formats do not easily extend to multi-unit markets with economies of scale. While preference elicitation is already a fundamental problem for bidders in single-unit combinatorial auctions, it becomes prohibitive in multi-unit combinatorial auctions. Markets with economies of scale and scope require a fundamentally different bidding language that allows to specify discount rules rather than a huge number of multi-unit package bids.

So far two central types of volume discounts are discussed in the literature: incremental discounts and total quantity discounts. Total quantity discounts have been described as a discount policy, where the supplier has specified a number of quantity intervals (aka. discount intervals), and the price per unit for the entire quantity depends on the discount interval in which the total amount ordered lies Goossens et al. (2007). In contrast, incremental volume discounts describe a discount policy, where the discounts apply only to the additional units above the threshold of the quantity interval. In business practice, such discount policies are often also defined on spend or on spend and quantity for one or more items. In addition, we will also allow for lump sum discounts, defining a one time reverse payment on overall spend or quantity. So far, optimization formulations only exist for incremental or for total quantity discount bids, defined on quantity purchased.

1.2 Contributions

We have investigated the factors limiting the use of well understood combinatorial auctions for a procurement setting. While the existing academic work covers important requirements, many real-world cases demand for a more powerful bidding language for practical applicability. In particular we focus on the following question:

How can bidders manageably express their cost structures incorporating (dis-)economies of scale and scope for use in a tractable procurement auction?

To answer this question, we did the following: We *introduce a compact bidding language for markets with economies of scale and scope*, referred to as $\mathcal{L}_{ESS}$. Our bidding language allows for two different types of discounts, which have already been discussed in the literature: incremental and total quantity discount bids. Our approach allows to *handle both types of volume discounts*, and we have seen several applications, where different bidders submit different types of volume discount bids. In addition to previous approaches, $\mathcal{L}_{ESS}$ allows for *lump sum discounts on total spend* to model economies of scope and *various conditions on spend or quantity* for the different discount types. As a result, $\mathcal{L}_{ESS}$ is considerably more expressive than previous approaches and gives suppliers high flexibility in specifying their offerings. Apart from expressiveness, we introduce *description length* as an important criterion for bid languages, since bidders cannot be expected to submit arbitrarily many parameters or bids. We will see that there are considerable differences between bundle bids, $\mathcal{L}_{ESS}$ bids with total quantity or with incremental volume discounts.

In this work, we will investigate the buyer's problem, who needs to select quantities from suppliers providing bids in $\mathcal{L}_{ESS}$ such that his costs are minimized and his demand is satisfied. We will refer to this problem as the *Supplier Quantity Selection ($\mathcal{SQS}$)* problem and propose a respective *mixed integer program* (MIP). Modeling matters and there are considerable differences in the solution time depending on different model formulations. We will also discuss *additional allocation constraints* as they are typically used for scenario navigation.

Procurement managers need a clear understanding of which problem sizes they can analyze in an interactive manner during the scenario analysis or in dynamic auctions. Therefore, we will show that $\mathcal{SQS}$ is $\mathcal{NP}$-complete, and report on an extensive *evaluation of the empirical hardness* of the supplier quantity selection problem. Similar analyses have recently been performed for the winner determination problem in combinatorial auctions by Leyton-Brown et al. (2009). It is important that problem instances for the experimental evaluation mirror real-world characteristics. We have *introduced a multiproduct cost function for markets with scale and scope economies* and generated bids based on these cost functions. $\mathcal{L}_{ESS}$ and the software framework used in this paper have already been used to support a number of high-stakes sourcing decisions with an industry partner. The synthetic bids matched the characteristics of those that we also found in the field. The experimental results show that realistic problem sizes of the $\mathcal{SQS}$ problem can be solved to optimality in a matter of minutes with IBM's CPLEX (version 12) and Gurobi 3.0. (All results reported are based on CPLEX.) We have also found that the shape of the underlying cost function and the demand can have a significant impact on the runtime of the problems and empirical evaluations need to be interpreted with care.

Previous work has only focused on the computational complexity of the winner determination problem. The cost curves in our experimental evaluation allowed us to *compare the total cost* achieved with $\mathcal{L}_{ESS}$ bids and different types of volume discounts and bids for split-award auctions. While we do not discuss mechanism design questions in our analysis, we assume a direct revelation mechanism where bidders submit bids in a way that reflects their cost curves as closely as possible. Even if bidding behavior in the lab or in the field might be different, this result suggests that a richer bidding language can lead to considerably lower cost and more efficient results in markets with economies of scale and scope with $\mathcal{L}_{ESS}$.

1.3 Structure of the Thesis

The thesis is organized as follows:

Chapter 2 gives an introduction into the fundamentals of multi-item and multi-unit procurement auctions and motivates our approach. Chapter 3 introduces our new bidding language and chapter 4 a MIP formulation to calculate the optimal allocation. Chapter 5 explains the design of our simulations. It describes the economic environment, our a priori assumptions about the bidding behavior and it defines and motivates the value model we created. In chapter 6 the results of our simulations are presented and chapter 7 finally draws conclusions and proposes some future research topics in this area. The Appendix contains an overview of the SPQR software platform used in our experiments and contains additional results of the conducted experiments.

Chapter 2

Combinatorial Auctions and their Applicability to Multi-unit Settings

Generally speaking, an auction is the process of trading items, where the goods are first announced and then bids are collected and in the end some of the bids are selected and realized. The difference between auctions and simple price negotiations is the fixed set of rules, which is known to all participants in advance.

In this chapter we will first give a short introduction into multi-unit and multi-item auctions and their different forms in general. Then we will take a closer look into combinatorial auctions as they are the theoretically very close to our problem, and also investigate which problems exist that are limiting their use in practical procurement settings. Part of this work has been published in Schneider et al. (2010). Finally we will study existing approaches to similiar problems.

2.1 Characterizations of Multi-item and Multi-unit Auctions

In the most basic form an auction involves only copy of a single item. If multiple identical instances of an item are sold at once the auction is called a multi-unit auction, and if multiple different items are involved, the action is called a multi-item or combinatorial auction. Naturally there also exist combinations of both types. Multi-item auctions are motivated by economies of scope where the value of an item for the buyer depends on the combination of items that he gets, and multi-unit auctions endorse economies of scale where the value of the items depends on the number of copies of an item that is traded.

If a single auctioneer sells goods or services to a competing set of bidders the auction is called a forward auction or sell auction and if a set of bidders competes for the right and duty to sell to a single auctioneer it is called a reverse or buy auction.

If all bids in an auction have to be submitted blindly in advance the auction is called a sealed-bid auction in contrast to an iterative auction where the bids are collected in multiple rounds. In most cases there is some sort of feedback mechanism guiding the bidders in between rounds.

2.2 Iterative Combinatorial Auctions

As outlined in chapter 1 combinatorial auctions are theoretically the mechanism of choice for our setting. In practical applications though, they never got accepted widely. As a first step we will therefore give a brief introduction into combinatorial auctions and examine why they are not suitable for complex procrement cases.

The typical process of an iterative combinatorial action (ICA) consists of the steps of bid submission, bid evaluation (aka winner determination, market clearing, or resource allocation) followed by feedback to the bidders, and is repeated until the stopping rule is fulfilled. The feedback is typically given in form of ask prices and the provisional allocation.

2.2.1 Bidding Languages

One of the most challenging parts of a multi-item combinatorial auction is the propagation of information from the bidders to the auctioneer, namely a set of pairs containing what the bidder wants and what he is willing to pay for it. In the classic English Auction setting with only one good sold at a time it reduces to the bidders announcing the amount they are willing to pay in order to receive the item.

In combinatorial auctions as we have seen the bidders do not only have the possibility to contemplate the entire set of individual items but also combinations of items called bundles. For each of the $2i - 1$ bundles the bidders now must communicate the amount they are willing to pay to the auctioneer in the worst case. In the literature this is called communication complexity and has been an active field of research not only in the environment of auctions.

Research in Artificial Intelligence has long dealt with questions of adequate knowledge representation and reasoning for a particular application domain. The most important decision to be made is the expressivity of the knowledge representation, where the optimal case is full expressivity, where all bundles can be priced.

Definition 1 (Expressiveness)**.** *A bidding language is fully expressive if it can express the valuation for all possible bundles.*

Typically, there are two downsides, the more expressive languages are, the harder it is to automatically derive inferences and the less understandable they are to humans. With the use of advanced software for the guidance of the bidders the understandability problem can be avoided to a certain amount.

On the other hand we know from information theory, that it is impossible to find a truly compact (linear) representation of all bid amounts. The proof can be read in Nisan (2006) and uses the fact that there are less than $2t$ strings of bit length t.

In the case of combinatorial auctions the most basic languages are based the boolean operations AND, OR and XOR. AND is commonly used to compose the bundles out of the individual items, whereas the bundles itself then are either connected with OR or XOR.

Definition 2 (XOR Bidding)**.** *The bidding language exclusive-OR (XOR) allows bidders to submit multiple sets of items that are paired with the amount he is willing to pay for this set. The auctioneer then is allowed to select at most one the item sets.*

Definition 3 (OR Bidding)**.** *The bidding language additive-OR (OR) allows bidders to submit multiple sets of items that are paired with the amount he is willing to pay for this set. The auctioneer then is allowed to select any combination of sets that have no item in common.*

The resulting OR and XOR bidding languages have in contrast to their similar composition very distinct characteristics. The OR bidding language for example is easier to understand for the bidders but it is not fully expressive. This follows from the fact that a bundle price cannot be lower than the sum of the prices of the items it is composed of.

2.2.2 Winner Determination

Given the private bidder valuations for all possible bundles, the efficient allocation can be found by solving the Winner Determination Problem (WDP). WDP can be formulated as a binary program using the decision variables $x_i(S)$ which indicate whether the bid of the bidder i for the bundle S belongs to the allocation:

Let $\mathcal{K} = \{1, \ldots, m\}$ denote the set of items indexed by k and $\mathcal{I} = \{1, \ldots, n\}$ denote the set of bidders indexed by i with private valuations $v_i(S) \geq 0$, $v_i(\emptyset) = 0$ for bundles $S \subseteq \mathcal{K}$. In addition we assume free disposal: If $S \subset T$ then $v_i(S) \leq v_i(T)$.

$$
\begin{aligned}
&\max_{x_i(S)} \sum_{S \subseteq \mathcal{K}} \sum_{i \in \mathcal{I}} x_i(S) v_i(S) \\
&\text{s.t.} \\
&\sum_{S \subseteq \mathcal{K}} x_i(S) \leq 1 \qquad \forall i \in \mathcal{I} \\
&\sum_{S: k \in S} \sum_{i \in \mathcal{I}} x_i(S) \leq 1 \qquad \forall k \in \mathcal{K} \\
&x_i(S) \in \{0, 1\} \qquad \forall i, S
\end{aligned}
\qquad \textbf{(WDP)}
$$

The first set of constraints guarantees that any bidder can win at most one bundle, which is only relevant for the XOR bidding language. The XOR language is used because it is fully expressive compared to the OR language which allows a bidder to win more than one bid. Subadditive valuations, where a bundle is worth less than the sum of individual items, cannot be expressed using the OR bidding language. The second set of constraints ensures that each item is only allocated once. Much research has focused on solving the winner determination problem, which is known to be NP-hard Park and Rothkopf (2005); Rothkopf et al. (1998); Sandholm (1999).

Having determined the winning bids, the auctioneer needs to decide what the winners should pay. A simple approach is for bidders to pay the amount of their bids. However, this creates incentives for bidders to shade their bids and might ultimately lead to strategic complexity, i.e., to speculation and inefficient allocations.

2.2.2.1 Pricing

The VCG auction is a generalization of the Vickrey auction for multiple heterogeneous goods. In this auction bidders have a dominant strategy of reporting their true valuations $v_i(S)$ on all bundles S to the auctioneer, who then determines the allocation and respective Vickrey prices. The VCG design charges the bidders the opportunity costs of the items they win, rather than their bid prices.

A difficulty in the combinatorial (multi-item) case is that the opportunity costs that every bidder has to be pay, cannot be expressed as the sum of the (second best) item bids. Given X^*_{-j} as the optimal allocation excluding bidder j the price paid by a winning bidder j on bundle S is calculated as

$$p_j(S) = \sum_{S \subseteq \mathcal{K}} v_j(S)x_j(S) - (\sum_{S \subseteq \mathcal{K}} \sum_{i \in \mathcal{I}} v_i(S)x^*_i(S) - \sum_{S \subseteq \mathcal{K}} \sum_{i \in \mathcal{I}} v_i(S)x^*_{-j}(S))$$

This requires to find X^*_{-j} for each winner, a problem that is as difficult as the WDP.

Although it has a simple dominant strategy, VCG design suffers from a number of practical problems since its outcome can be outside of the *core* Ausubel and Milgrom (2006b); Rothkopf (2007). The revenue can decrease as more bidders are added or as some bidders raise their bids, and can be unacceptably low or even zero. The VCG mechanism is susceptible to shill bidding and collusion.

Formally, let N denote the set of all bidders $\mathcal{I}$ and the auctioneer with $i \in N$, and $M \subseteq N$ be a coalition of bidders with the auctioneer. Let $w(M)$ denote the coalitional value for a subset M, equal to the value of the WDP with all bidders $i \in M$ involved. (N, w) is the coalitional game derived from trade between the seller and bidders. Core payoffs π are then defined as follows:

$$Core(N, w) = \{\pi \geq 0 | \sum_{i \in N} \pi_i = w(N), \sum_{i \in M} \pi_i \geq w(M) \ \ \forall M \subset N\}$$

This means, there should be no coalition $M \subset N$, which can make a counteroffer that leaves themselves and the seller at least as well as the currently winning coalition. In their seminal paper, Bikhchandani and Ostroy (2002) show that there is an equivalence between the core of the coalitional game and the competitive equilibrium for single-sided auctions.

Definition 4 (Competitive Equilibrium, CE Parkes (2006))**.** *Prices $\mathcal{P}$, and allocation X^* are in competitive equilibrium if allocation X^* maximizes the payoff of every bidder and the auctioneer revenue given prices $\mathcal{P}$. The allocation X^* is said to be supported by prices $\mathcal{P}$ in CE.*

Bikhchandani and Ostroy (2002) show that X^* is supported in CE by some set of prices $\mathcal{P}$ if and only if X^* is an efficient allocation. Although CE always exist, they possibly require non-linear and non-anonymous prices. Prices are *linear* if the price of a bundle is equal to the sum of prices of its items, and prices are *anonymous* if prices are equal for every bidder. Non-anonymous ask prices are also called *personalized* prices. Following three types of ask prices are usually discussed:

1. a set of linear anonymous prices $\mathcal{P} = \{p(k)\}$
2. a set of non-linear anonymous prices $\mathcal{P} = \{p(S)\}$
3. a set of non-linear personalized prices $\mathcal{P} = \{p_i(S)\}$

Generating minimal CE prices is desirable, since it usually imposes incentive compatibility of the auction design.

Definition 5 (Minimal CE Prices Parkes (2006))**.** *Minimal CE Prices minimize the auctioneer revenue $\Pi(X^*, \mathcal{P})$ on an efficient allocation X^* across all CE prices which support it.*

Minimal CE prices are not necessarily unique. One way to derive minimal CE prices could be to use the duals of the WDP. Unfortunately, WDP is a binary program, and the solution will not be integral in most cases if solved as an LP relaxation. Therefore, the duals will not be minimal and over-estimate the value of the items.

2.2.3 Non-Linear Personalized Price Auctions

By adding constraints for each set partition of items and each bidder to the WDP the formulation can be strengthened, so that the integrality constraints on all variables can be omitted but the solution is still always integral. Such a formulation describes every feasible solution to an integer problem, and is solvable with linear programming. Personalized non-linear CE prices can be derived from the dual strengthened problem as described in Bikhchandani and Ostroy (2002).We will refer to this formulation as NLPPA WDP.

From duality theory follows that the *complementary slackness conditions* must hold in the case of optimality. It has been shown that these are equal to the CE conditions, stating that every buyer receives a bundle out of his demand set $D_i(\mathcal{P})$ and the auctioneer selects the revenue maximizing allocation.

Definition 6 (Demand Set)**.** *The demand set $D_i(\mathcal{P})$ of a bidder i includes all bundles which maximize a bidder's payoff π_i at the given prices $\mathcal{P}$:*

$$D_i(\mathcal{P}) = \{S : \pi_i(S,\mathcal{P}) \geq \max_{T \subseteq \mathcal{K}} \pi_i(T,\mathcal{P}), \pi_i(S,\mathcal{P}) \geq 0, S \subseteq \mathcal{K}\}$$

Complementary slackness provides us with an optimality condition, which also serves as a termination rule for NLPPAs. If bidders follow the straightforward strategy then terminating the auction when each active bidder receives a bundle in a revenue-maximizing allocation will result in the efficient outcome. Note that a demand set can include the empty bundle. Additionally, the starting prices must represent a feasible dual solution de Vries et al. (2007). A trivial solution is to use zero prices for all bundles.

Individual NLPPA formats have different rules of selecting bundles and bidders whose prices are increased. The Ascending Proxy Auction has been suggested in the context of the US FCC spectrum auction design Ausubel et al. (2006); Ausubel and Milgrom (2006a). It is similar to iBundle(3), but the use of proxy agents is mandatory, which essentially lead to a sealed-bid auction format. Since this is the main difference compared to iBundle(3), we will only discuss iBundle in the following.

The ***iBundle auction*** calculates a provisional revenue maximizing allocation at the end of every round and increases the prices based on the bids of non-winning (unhappy) bidders. Parkes and Ungar (2000) suggest three different versions of iBundle: iBundle(2) with anonymous prices, iBundle(3) with personalized prices, and iBundle(d) which starts with anonymous prices and switches to personalized prices for agents which submit bids for disjoint bundles. We will restrict our analysis to *iBundle(2)* and *iBundle(3)* for the questions of this paper.

The ***dVSV auction*** de Vries et al. (2007) design differs from iBundle in that it does not compute a provisional allocation in every round but increases prices for a *minimally undersupplied set of bidders.*

Definition 7 (Minimally undersupplied set of bidders)**.** *A set of bidders is minimally undersupplied if in no efficient allocation each bidders receive a bundle from his demand set, and removing one of the bidders forfeits this property.*

Similar to iBundle(3), it maintains non-linear personalized prices and increases the prices for all agents in a minimally undersupplied set based on their bids of the last round.

2.2.4 Ascending Vickrey Auctions

iBundle(3) and dVSV result in minimal CE prices. Minimal CE prices and VCG payments typically differ. Bikhchandani and Ostroy (2002) show that the bidders are substitutes condition (BSC) is necessary and sufficient to support VCG payments in competitive equilibrium.

Definition 8 (Bidders are Substitutes Condition, BSC)**.** *The BSC condition requires*

$$w(N) - w(N \setminus M) \geq \sum_{i \in M} [w(N) - w(N \setminus i)], \forall M \subseteq N$$

If BSC fails, the VCG payments are not supported in any price equilibrium and truthful bidding is not an equilibrium strategy. A bidder's payment in the VCG mechanism is always less than or equal to the payment by a bidder at any CE price. Even though the BSC condition is sufficient for VCG prices to be supported in CE, Ausubel and Milgrom (2006a) show that a slightly stronger bidder submodularity condition (BSM) is required for an ascending auction to implement VCG payments.

Definition 9 (Bidder Submodularity Condition, BSM)**.** *BSM requires that for all $M \subseteq M' \subseteq N$ and all $i \in N$ there is*

$$w(M \cup \{i\}) - w(M) \geq w(M' \cup \{i\}) - w(M')$$

de Vries et al. (2007) show that under BSM their primal-dual auction yields VCG payments. When the BSM condition does not hold, the property brakes down and a straightforward strategy is likely to lead a bidder to pay more than the VCG price for the winning bundle Dunford et al. (2007). de Vries et al. (2007) also show that when at least one bidder has a non-substitutes valuation an ascending CA cannot implement the VCG outcome. BSM is often not given in realistic value models as the ones provided by CATS Leyton-Brown et al. (2000).

The ***CreditDebit Auction*** by Mishra and Parkes (2007) is an extension to the dVSV design which achieves the VCG outcome for general valuations. It introduces the concept of universal competitive equilibrium (UCE) prices, which are CE prices for the main economy as well as for every marginal economy, where a single buyer is excluded. Their auction terminates as soon as UCE prices are reached and VCG payments are determined either as one-time discounts dynamically during the auction. The authors show that truthful bidding is an ex post Nash equilibrium in these auctions. This is not a contradiction with the previous paragraph, since the bidders do not always pay their bids but can receive discounts. The auction, however, shares the central problem of the VCG auction: if buyer submodularity is not given, the outcomes might not be in the core.

2.2.5 Linear Price Auctions

Though we have seen that linear prices do not support the efficient allocation in all cases, they are the widespread in practical use, and therefore we include two of them here.

The ***Combinatorial Clock Auction (CC auction)*** described in Porter et al. (2003) utilizes anonymous linear prices called *item clock prices*. In each round bidders express the quantities desired on the bundles at the current prices. As long as demand exceeds supply for at least one item (each item is counted only once for each bidder) the price clock "ticks" upwards for those items (the item prices are increased by a fixed price increment), and the auction moves on to the next round. If there is no excess demand and no excess supply, the items are allocated corresponding to the last round bids and the auction terminates. If there is no excess demand but there is excess supply (all active bidders on some item did not resubmit their bids in the last round), the auctioneer solves the winner determination problem considering all bids submitted during the auction runtime. If the computed allocation does not displace any bids from the last round, the auction terminates with this allocation, otherwise the prices of the respective items are increased and the auction continues.

The ***ALPS*** (Approximate Linear PriceS) design Bichler et al. (2009) is based on the *Resource Allocation Design (RAD)* proposed by Kwasnica et al. (2005) and uses anonymous linear ask prices which are derived from the restricted dual of the LP relaxation of the WDP Rassenti et al. (1982). The termination rule of the RAD auction has been improved in ALPS to prevent premature auction termination. Furthermore, the ask price calculation better minimizes and balances the prices. In a modified version ***ALPSm***, all bids submitted in an auction remain active throughout the auction. This rule had a significant positive impact on efficiency in experimental evaluations. In the following, we will only consider the results of ALPSm and the Combinatorial Clock auction to provide a comparison to NLPPAs.

2.2.6 Problems

Clearly, NLPPAs can be considered a fundamental contribution to the combinatorial auction theory, as they describe iterative auction designs that are fully efficient. However, they are based on a number of assumptions. In particular, straightforward bidding might not hold in practical settings where bidders have bounded rationality, given that bidders don't know, whether the submodularity condition holds, and there is a huge number of bundles a bidder has to deal with. Recent experimental work has actually shown that bidders did not follow a pure best-response strategy, even in simple settings with only a few items Scheffel et al. (2010). It is also important to note that, if one of the bidders deviates from his best-response strategy in a Nash equilibrium, a Nash equilibrium strategy is also not necessarily a best response for other bidders any more. For the reasons outlined above, it is likely that not all bidders will follow a straightforward bidding strategy.

Therefore, it is important to understand their performance in case of non-straightforward bidding strategies, when bidders either cannot follow such a strategy for computational or cognitive reasons, or deliberately choose another strategy.

2.2.7 Evaluation

Laboratory experiments are very costly and typically restricted to a small number of treatments and sessions. Therefore, the robustness of NLPPAs against different types of bidding behavior can best be analyzed in computational experiments.

2.2.7.1 Setup of Numerical Simulations

Our simulation environment consists of three main components. A *value model* defines valuations of all bundles for each bidder. A *bidding agent* implements a bidding strategy adhering to the given value model and to the restrictions of the specific auction design. An *auction processor* implements the auction logic, enforces auction protocol rules, and calculates allocations and ask prices.

Different implementations of value models, bidding agents (i.e., strategies) and auction processors can be combined, which allows performing sensitivity analysis by running a set of simulations while changing only one component and preserving all other parameters.

For the comparison of auction formats, we use a set of *focus variables*, such as allocative efficiency, revenue distribution, and speed of convergence measured by number of auction rounds. We believe that the number of rounds is the most relevant number to represent the auction duration, since the absolute time will heavily depend on computational capacities of bidders and speed of communication. We use allocative efficiency (or simply efficiency) as a primary measure to benchmark auction designs. It is measured as the ratio of the total valuation of the resulting allocation X to the total valuation of an efficient allocation X^* Kwasnica et al. (2005):

$$E(X) = \frac{\sum_{i \in \mathcal{I}} v_i(\bigcup_{S \subseteq \mathcal{K}: x_i(S)=1} S)}{\sum_{i \in \mathcal{I}} v_i(\bigcup_{S \subseteq \mathcal{K}: x_i^*(S)=1} S)}$$

The term $\sum_{i\in\mathcal{I}} v_i(\bigcup_{S\subseteq\mathcal{K}:x_i(S)=1} S)$ can be simplified to $\sum\limits_{i\in\mathcal{I}}\sum\limits_{S\subseteq\mathcal{K}} x_i(S)v_i(S)$ in case of the XOR bidding language, since at most one bundle per bidder can be allocated.

Similarly, we measure the revenue distribution which shows how the overall economic gain is distributed between the auctioneer and bidders. Given the resulting allocation X and the bid prices $\{b_i(S)\}$, the *auctioneer's revenue share* is measured as the ratio of the auctioneer's income to the total sum of valuations of an efficient allocation X^*:

$$R(X) = \frac{\sum\limits_{S\subseteq\mathcal{K}}\sum\limits_{i\in\mathcal{I}} x_i(S)b_i(S)}{\sum\limits_{i\in\mathcal{I}} v_i(\bigcup_{S\subseteq\mathcal{K}:x_i^*(S)=1} S)}$$

In order to compare different settings, we will sometimes also plot auctioneer revenue as % of the revenue in the VCG outcome. The cumulative bidders' revenue share is $E(X)-R(X)$. Note that it is possible for two auction outcomes with equal efficiency to have significantly different auctioneer revenues. In the following, we will briefly discuss the value models and behavioral assumptions in our bidding agents.

2.2.7.2 Value Models

Since there are hardly any real-world CA data sets available, we have based our research on the Combinatorial Auctions Test Suite (CATS) value models that have been widely used for the evaluation of WDP algorithms Leyton-Brown et al. (2000). In the following, we will describe a value model as a function that generates realistic, economically motivated combinatorial valuations on all possible bundles for every bidder.

In the ***Real Estate 3x3*** value model, the items sold in the auction are the real estate lots k, which have valuations v_k drawn from the same normal distribution for each bidder. Adjacency relationships between two pieces of land l and m (e_{lm}) are created randomly for all bidders. Edge weights $r_{lm} \in [0,1]$ are generated for each bidder and used to determine bundle valuations of adjacent pieces of land:

$$v(S) = (1 + \sum_{e_{lm}:l,m\in S} r_{lm}) \sum_{k\in S} v_k$$

We use the *Real Estate 3x3* value model with 9 lots and normally distributed individual item valuations with a mean of 10 and a variance of 2. There is a 90% probability of a vertical or horizontal edge, and an 80% probability of a diagonal edge. Edge weights have a mean of 0.5 and a variance of 0.3. Unless explicitly stated (Section 2.2.8.5), all auction experiments with the Real Estate value model are conducted with 5 bidders.

The ***Transportation*** value model models a nearly planar transportation graph in Cartesian coordinates, where each bidder is interested in securing a path between two randomly selected vertices (cities). The items traded are edges (routes) of the graph. Parameters for the Transportation value model are the number of items (edges) m and graph density ρ, which defines an average number of edges per city, and is used to calculate the number of vertices as $(m * 2)/\rho$. The bidder's valuation for a path is defined by the Euclidean distance between two nodes multiplied by a random number, drawn from a uniform distribution. We consider the Transportation value model with 25 edges, 15 vertices, and 15 bidders. Every bidder had interest in 16 different bundles on average.

The ***Pairwise Synergy*** value model from An et al. (2005) is defined by a set of valuations of individual items $\{v_k\}$ with $k \in \mathcal{K}$ and a matrix of pairwise item synergies $\{syn_{k,l} : k, l \in \mathcal{K}, syn_{k,l} = syn_{l,k}, syn_{k,k} = 0\}$. The valuation of a bundle S is then calculated as

$$v(S) = \sum_{k=1}^{|S|} v_k + \frac{1}{|S|-1} \sum_{k=1}^{|S|} \sum_{l=k+1}^{|S|} syn_{k,l}(v_k + v_l)$$

A synergy value of 0 corresponds to completely independent items, and the synergy value of 1 means that the bundle valuation is twice as high as the sum of the individual item valuations. The model is very generic, as it allows different types of synergistic valuations, but it was also used to model certain types of transportation auctions An et al. (2005). In this paper, we use the Pairwise Synergy value model with 7 items, where item valuations are drawn for each auction independently from a uniform distribution between 4 and 12. The synergy values are drawn from a uniform distribution between 1.5 and 2.0. We tested lower synergy values, but found little difference in the results. All auctions with Pairwise Synergy value model have 5 bidders.

In the *Real Estate* and *Pairwise Synergy* value models bidders were interested in a maximum bundle size of 3, because in these value models large bundles are always valued over small ones. This is also motivated by real-world observations An et al. (2005). Without this limitation, the auction usually degenerates into a scenario with a single winner for the bundle containing all items. In the *Transportation* value model bidders were not restricted in bundle size.

The value models describe very different processes for the generation of valuations. Some have small bundles only, some have small and large bundles, and also the way how synergies are determined is different.

For the numerical experiments all valuations for all bidders need to be generated and in each round all bundles need to be sorted by payoff. This leads to substantial computation time and puts a practical limit on the size of the value model, which can be simulated. For example, a single CreditDebit auction with the Real Estate value model took up to two hours due to the high number of auction rounds.

2.2.7.3 Behavioral Assumptions and Bidding Agents

Theoretical models provide arguments for straightforward bidding in NLPPAs. In an auction with private valuations, however, when the bidders do not know whether the bidder submodularity holds, and therefore can decide to shade their bids. There are also cognitive barriers for the straightforward strategy, since bidders need to determine their demand set for an exponential number of possible bundle bids in each round.

It is useful to look at empirical observations and the behavioral literature to derive hypotheses on bidding strategies in NLPPAs. Unfortunately, so far, there have only been a few lab experiments using NLPPAs and we do not know of any applications in the field. Chen and Takeuchi (2009) compared the VCG auction and iBEA in experiments where humans competed against artificial bidders. Bidders were significantly more likely to bid on packages with a high temporary profit, but did not follow the pure straightforward strategy. More recently, Scheffel et al. (2010) conducted experiments comparing iBundle, ALPS, Combinatorial Clock, and the VCG auction. Again, bidders in the iBundle auction did not follow the straightforward strategy, even though they were provided with a decision support tool that helped them select their demand set. There was, however, a high likelihood to bid on one of the best 10 bundles based on their payoff in the current round. Bidder idiosyncrasies and mixed strategies such as in trembling-hand perfect equilibria Selten (1975) are possible explanations of these findings. Unfortunately, behavioral models of bidding in multi-item auctions are rare Plott and Salmon (2002).

We have developed a number of bidding agents that are motivated by different conjectures about the behavior of bidders in NLPPAs. These agents implement a bidding strategy adhering to the given value model and to the restrictions of a specific auction format.

The ***straightforward*** bidder is motivated by the theory. He always bids his demand set which maximizes his surplus if it were to win one of its bids at the current prices. Obviously, the results of auctions with straightforward bidders shall achieve the outcomes predicted by the theory.

From lab experiments, we know that bidders do not always follow the straightforward strategy due to cognitive reasons and simple errors. The ***forgetful*** agent follows the straightforward strategy, but "forgets" to submit 10% of his bids in each round. These 10% are determined randomly and independently for every round. Similarly we also modeled a ***heuristic*** bidder. This agent randomly selects 5 out of his 20 best bundles based on his payoff in a round.

Sometimes bidders in the lab try to increase their chances of winning by bidding on many bundles, which can be explained by risk aversion. The ***powerset*** bidder evaluates all possible bundles in each round, and submits bids for bundles which are profitable given current prices. We have limitedthis bidder to 10 bundles with the highest payoff in each round, which is based on the observation that bidders typically do not bid on many bundle bids in an auction round Scheffel et al. (2010).

Typically, in our settings these 10 bundles will have up to 20 % difference in payoff at the beginning of the auction given equal start prices.

The NLPPAs, which we study in this work, terminate as soon as the new provisional allocation includes bids from all active bidders. This termination rule may motivate collusive bidders to submit more then just the demand set in the first round in the hope that a suitable allocation is found early and the auction terminates before the prices rise. The ***level*** bidder models a dishonest strategy that tries to exploit this idea. We modify the straightforward bidder by lowering valuations of his best l bundles and setting all of them equal to the valuation of the l^{th} best bundle. In our simulations, we have used $l = 10$. Note that proxy bidding agents cannot prevent bidders from adopting this strategy.

If a bidder is restricted in time during the auction, he might select his most valuable bundles a priori, and stick to this selection throughout the auction.

This ***preselect*** agent selects his 20 most valuable bundles and follows the straightforward strategy, only using the preselected bundles. Again, a proxy cannot detect or prevent this strategy.

To ensure comparability between auction formats, we use a fixed minimum increment of 1 and integer valuations for all valuations. Under these conditions, NLPPAs always terminate with an efficient solution given straightforward bidding de Vries et al. (2007).

2.2.8 Results

This section describes the results of simulations in terms of efficiency, revenue distribution and the number of auction rounds. Unless explicitly stated otherwise, each auction setup was repeated 50 times with different random seeds for value models and, where appropriate, bidding agents. Overall, we provide the results of an extensive experimental design with more than 25,000 auctions.

Bidding Agent	ICA Format	Credit-Debit	dVSV	iBundle (3)	iBundle (2)	ALPSm	Clock	VCG truth
Straightforward	Efficiency in %	100.00	100.00	100.00	99.90	98.64	95.16	100.00
	Rev. Auctioneer in %	83.45	84.57	84.61	83.88	84.09	87.05	83.17
	Rev. all bidders in %	16.55	15.43	15.39	16.01	14.55	8.10	16.83
	Rounds	139.96	137.90	151.28	149.26	72.56	28.76	1.00
Forgetful 10% chance to forget each bid	Efficiency in %	99.92	99.78	100.00	99.92	98.64	96.35	
	Rev. Auctioneer in %	81.73	85.28	84.73	83.88	84.11	88.02	
	Rev. all bidders in %	18.19	14.51	15.27	16.03	14.54	8.30	
	Rounds	550.54	545.30	329.68	237.20	72.26	29.52	
Level10	Efficiency in %	90.00	90.05	89.71	90.12	91.12	86.67	
	Rev. Auctioneer in %	72.10	72.29	72.34	71.55	72.25	76.46	
	Rev. all bidders in %	18.12	17.97	17.58	18.72	19.01	10.30	
	Rounds	82.38	82.00	133.22	130.96	128.88	26.06	
Powerset10 10 best bundles selected in each round	Efficiency in %	90.50	89.67	98.48	99.20	99.57	97.27	
	Rev. Auctioneer in %	23.10	71.53	82.93	82.14	87.65	94.09	
	Rev. all bidders in %	67.33	18.22	15.55	17.02	11.91	3.20	
	Rounds	1525.44	1519.58	979.18	283.46	24.94	25.30	
Preselect20 20 best bundles	Efficiency in %	98.24	98.24	98.24	97.56	91.51	92.07	
	Rev. Auctioneer in %	76.26	79.55	79.27	78.69	76.35	84.65	
	Rev. all bidders in %	21.94	18.64	18.92	18.79	15.07	7.39	
	Rounds	147.18	141.32	146.34	145.62	61.76	28.32	
Heuristic 5 random bundles out of 20 best	Efficiency in %	76.09	73.68	98.94	97.51	99.29	98.18	
	Rev. Auctioneer in %	22.11	52.16	82.03	83.55	87.88	94.19	
	Rev. all bidders in %	54.07	21.56	16.88	14.02	11.40	4.00	
	Rounds	3183.12	3058.88	1860.88	524.04	27.44	25.60	

TABLE 2.1: Performance of iterative combinatorial auctions with differing Bidding Agents for the Real Estate value model

2.2.8.1 Analysis of Different Bidding Strategies

First we assume that all bidders in the auction use the same strategy and analyze the results for different value models and bidding strategies. In particular, we want to investigate how different NLPPAs behave when the bidders deviate from the straightforward strategy.

The average performance metrics are summarized in Table 2.1 for the Real Estate, in Table 2.2 for the Pairwise Synergy, and in Table 2.3 for the Transportation value models.

Interestingly, we found a similar pattern in the results for all value models. We have also repeated the same tests with different instances of the Pairwise Synergy value model, where the synergy level was lower and in some cases negative (subadditive valuations), and with the Real Estate value model with a maximum bundle size of four instead of three. These modifications led to similar results.

Bidding Agent	ICA Format	Credit-Debit	dVSV	iBundle (3)	iBundle (2)	ALPSm	Clock	VCG truth
Straightforward	Efficiency in %	100.00	100.00	100.00	100.00	99.65	99.53	100.00
	Rev. Auctioneer in %	89.63	90.53	90.46	90.47	89.81	92.83	89.60
	Rev. all bidders in %	10.37	9.47	9.54	9.53	9.85	6.69	10.40
	Rounds	204.44	202.96	154.96	154.82	68.70	31.68	1.00
Forgetful 10% chance to forget each bid	Efficiency in %	99.81	99.65	100.00	99.99	99.65	99.56	
	Rev. Auctioneer in %	88.31	91.04	90.41	90.54	89.92	92.97	
	Rev. all bidders in %	11.50	8.63	9.59	9.45	9.73	6.58	
	Rounds	715.40	712.26	333.48	243.32	68.60	31.86	
Level10	Efficiency in %	98.40	98.41	98.48	98.47	97.21	94.46	
	Rev. Auctioneer in %	88.38	88.78	88.82	88.80	88.08	86.90	
	Rev. all bidders in %	10.03	9.64	9.67	9.68	9.14	7.59	
	Rounds	150.16	149.44	150.28	150.10	117.74	33.66	
Powerset10 10 best bundles selected in each round	Efficiency in %	96.01	95.91	99.16	99.55	99.75	99.51	
	Rev. Auctioneer in %	35.82	87.06	89.01	89.98	92.28	98.19	
	Rev. all bidders in %	60.18	8.85	10.15	9.57	7.47	1.31	
	Rounds	1353.50	1352.36	650.24	192.60	29.34	31.24	
Preselect20 20 best bundles preselected before the auction	Efficiency in %	85.80	85.80	85.80	85.80	82.75	85.21	
	Rev. Auctioneer in %	79.47	79.91	79.97	79.98	77.07	82.30	
	Rev. all bidders in %	6.35	5.90	5.84	5.84	5.70	2.91	
	Rounds	230.90	230.06	138.54	138.46	48.14	30.56	
Heuristic 5 random bundles out of 20 best	Efficiency in %	85.33	85.40	98.70	98.15	99.40	99.35	
	Rev. Auctioneer in %	25.97	60.64	88.86	88.85	92.87	97.92	
	Rev. all bidders in %	59.37	24.78	9.84	9.30	6.53	1.41	
	Rounds	2551.94	2544.98	1261.50	338.50	31.86	31.34	

TABLE 2.2: Performance of iterative combinatorial auctions with differing Bidding Agents for the Pairwise Synergy value model

Only the Transportation value model was different with respect to its high robustness against preselect bidding. The main reason is the low number of

Bidding Agent	ICA Format	Credit-Debit	dVSV	iBundle (3)	iBundle (2)	ALPSm	Clock	VCG truth
Straightforward	Efficiency in %	100.00	100.00	100.00	99.99	99.55	99.48	100.00
	Rev. Auctioneer in %	56.13	66.93	65.43	65.36	65.92	77.10	55.49
	Rev. all bidders in %	43.87	33.07	34.57	34.63	33.60	22.41	44.51
	Rounds	216.78	205.18	78.66	78.02	32.24	29.64	1.00
Forgetful 10% chance to forget each bid	Efficiency in %	99.47	99.60	99.93	99.97	99.55	99.58	
	Rev. Auctioneer in %	51.38	67.88	65.55	65.81	65.92	77.10	
	Rev. all bidders in %	48.10	31.76	34.35	34.15	33.60	22.48	
	Rounds	434.96	414.70	124.84	106.90	32.24	29.74	
Level10	Efficiency in %	84.95	85.06	83.64	84.01	83.96	84.56	
	Rev. Auctioneer in %	23.13	26.56	26.36	26.30	27.65	37.05	
	Rev. all bidders in %	61.92	58.46	57.31	57.70	56.19	47.36	
	Rounds	60.56	56.06	40.14	39.38	19.62	14.02	
Powerset10 10 best bundles selected in each round	Efficiency in %	91.33	91.73	97.28	97.26	99.78	97.39	
	Rev. Auctioneer in %	2.47	56.09	58.94	59.95	72.56	88.60	
	Rev. all bidders in %	88.80	35.52	38.19	37.20	27.18	8.82	
	Rounds	312.56	311.36	154.06	93.36	19.80	25.00	
Preselect20 20 best bundles preselected before the auction	Efficiency in %	99.80	99.80	99.80	99.80	99.52	99.24	
	Rev. Auctioneer in %	55.40	66.51	65.24	65.24	66.44	76.77	
	Rev. all bidders in %	44.39	33.28	34.56	34.56	33.06	22.50	
	Rounds	217.24	205.62	78.64	78.12	32.56	29.88	
Heuristic 5 random bundles out of 20 best	Efficiency in %	84.14	84.50	96.24	96.10	99.75	97.73	
	Rev. Auctioneer in %	6.06	54.98	59.04	59.59	73.82	88.35	
	Rev. all bidders in %	78.26	29.75	37.10	36.43	25.90	9.33	
	Rounds	789.24	788.24	268.62	158.12	21.20	25.08	

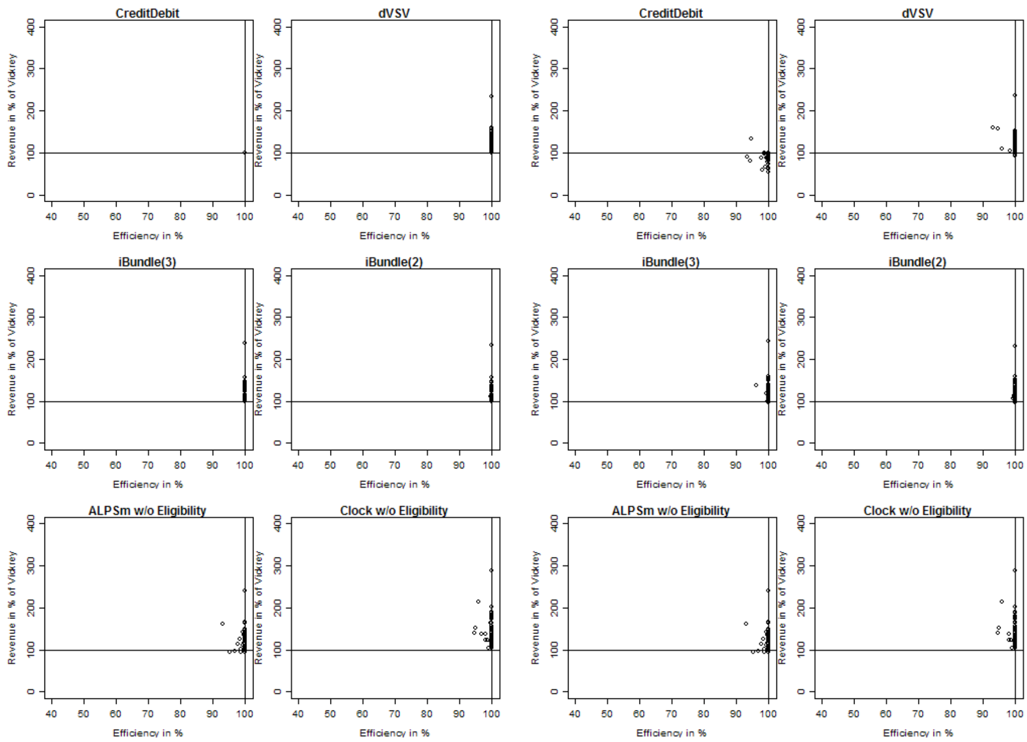
Table 2.3: Performance of iterative combinatorial auctions with differing Bidding Agents for the Transportation value model

bundles with significant competition, which is due to the underlying topology of transportation networks and the fact that only a few bundles are of interest to every bidder. For the same reason, the level bidders could successfully collude and significantly increase their payoff.

Below we describe the main findings for every type of the bidding strategy, for the case when all bidders in the auction follow it.

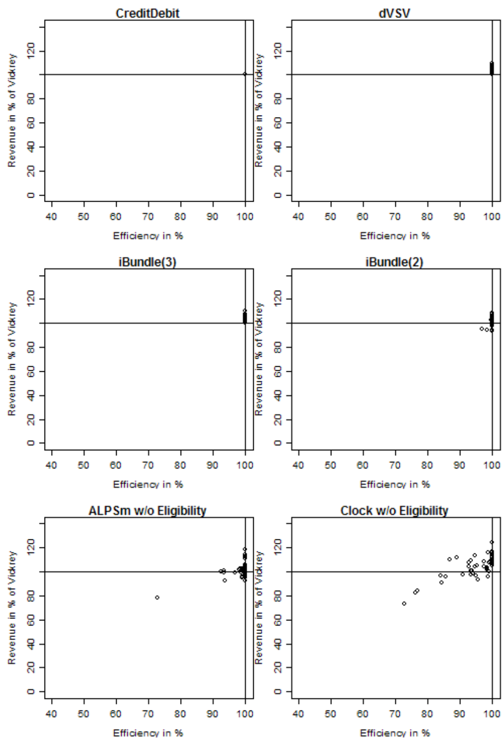
2.2.8.1.1 Straightforward Bidder Our computational experiments with straightforward bidders and NLPPAs yielded outcomes in line with the theory. All NLPPAs were efficient. iBundle(3) and dVSV achieved VCG outcomes only when the BSC condition was satisfied. When BSC was not satisfied, iBundle(3) and dVSV resulted in higher prices. In the Transportation value model we could observe cases where the prices in NLPPAs were up to 250% higher than in the VCG auction (see Figure 2.1), whereas in the Real Estate and Pairwise Synergy value models, the price increase was low (see Figures 2.2 and 2.3).

The CreditDebit auction always resulted in VCG payoffs, as theory predicts. iBundle(2) did not result in an efficient outcome for some instances, but these

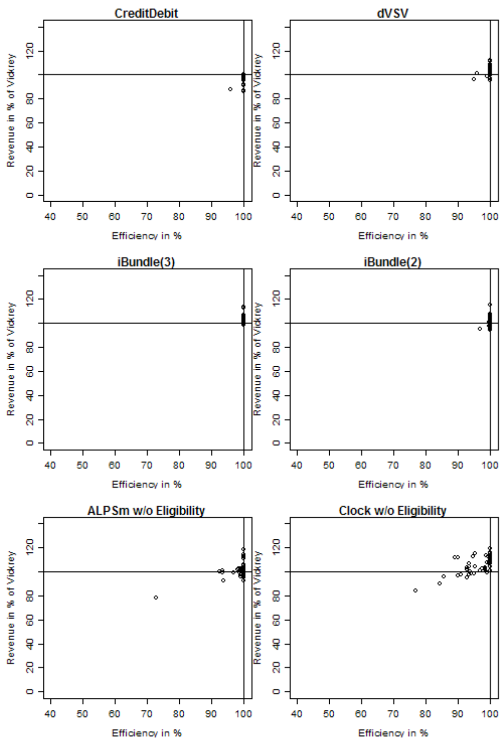


(a) Straightforward (b) Forgetful

FIGURE 2.1: Efficiency and revenue of iterative combinatorial auctions for the Transportation value model

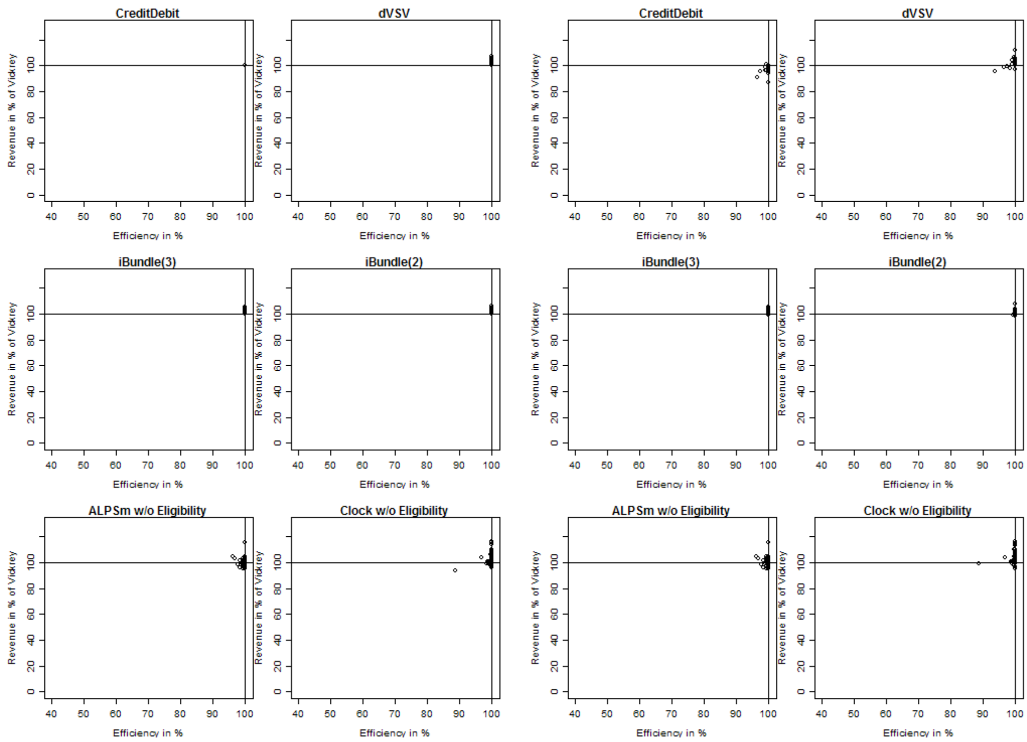


(a) Straightforward



(b) Forgetful

FIGURE 2.2: Efficiency and revenue of iterative combinatorial auctions for the Real Estate 3x3 value model



(a) Straightforward (b) Forgetful

FIGURE 2.3: Efficiency and revenue of iterative combinatorial auctions for the Pairwise Synergy value model

occasions were rare and the loss of efficiency generally very low. One reason is, that iBundle(2) is efficient for superadditive valuations Parkes and Ungar (2000), which is mostly the case in our value models.

In the linear-price auctions (ALPS and CC) the straightforward bidder was less efficient than NLPPAs. Still, their efficiency was 95.16% and 98.64% on average for the Real Estate value model, and even more than 99% for the Pairwise Synergy and the Transportation value model. It is important to note that with the straightforward bidder, we observed cases in ALPSm and CC auctions, where efficiency was as low as 70%. If bidders follow the straightforward strategy in ALPS and the CC auction, it can happen that they do not reveal certain valuations that would be part of the efficient solution Bichler et al. (2009). In the presence of activity rules, bidders are forced to bid on more than just their demand set. This will have a positive effect on the robustness of ALPSm format as we will see when we discuss powerset bidders.

2.2.8.1.2 Forgetful Bidder NLPPAs were fairly robust against forgetful bidders, and the efficiency losses were low. Only the number of auction rounds increased significantly across all value models. For example, the CreditDebit auction took on average 139.96 auction rounds in the Real Estate value model with straightforward bidders and 550.54 rounds with forgetful bidders. Interestingly, the linear price auctions were hardly impacted at all compared to the straightforward bidding strategy. Also the average number of auction rounds remained almost the same.

2.2.8.1.3 Level Bidder As expected, the collusive level bidder causes an efficiency loss. In the Real Estate value model efficiency dropped to around 90% in all auction formats and also the auctioneer revenue was not significantly different (t-test, $\alpha = 0.05$). In the Transportation value model, this strategy was very successful. Here, the level bidder achieved a significantly higher revenue than with a straightforward strategy, however, at the expense of efficiency, which dropped to around 84% on average. For the Transportation value model, the competition is focused on a small number of items or legs in the transportation network and it is more likely that a valid allocation is found earlier when all bidders follow the level bidding strategy.

In the CreditDebit auction the high bidder revenue caused by overestimated price discounts due to the bid shading in the level strategy. In all auction formats, however, there are also instances in which the auctioneer gained more and the bidders gained less compared to the straightforward bidding strategy. This strategy works only in an expected sense if all bidders adhere to it. It does not represent a stable equilibrium.

2.2.8.1.4 Powerset Bidder For the iBundle(2), iBundle(3), dVSV, and the CreditDebit auction, this strategy led to a significant decrease in efficiency (t-test, $\alpha = 0.05$). For iBundle auctions the efficiency loss was lower than for dVSV and the CreditDebit auctions. Apparently, the price calculation algorithm using minimally undersupplied set is less robust against non-straightforward bidding.

In contrast, the efficiency and auctioneer revenue share of ALPS auctions was equal or higher compared to the straightforward strategy in all value models. Simultaneously the number of rounds was significantly reduced. The CC auction performed well in homogenous markets, modeled by Real Estate and Pairwise Synergy value models. Typically, these linear-price auctions are used with strong activity rules to encourage the revelation of many bundle preferences already in the early rounds of an auction, which might lead to a similar strategy with bidders in the field.

2.2.8.1.5 Preselect Bidder The efficient solution cannot be found if it includes the bundles which are omitted by the preselect bidder. In the Transportation value model this had little effect on efficiency compared to straightforward bidding, since there is only a small number of interesting bundles for every bidder. In other value models we could see a significant decrease in all measurements.

2.2.8.1.6 Heuristic Bidder Heuristic bidder, who bids on ramdom 5 of his 20 best bundles, causes significant efficiency losses in all NLPPAs. We observed the highest efficiency losses for dVSV and the CreditDebit auction. The reason for the low revenue of the CreditDebit auction is that discounts are miscalculated if not all bundle bids are available at the end. In addition, the more complex price update rule is less robust against non-straightforward bidding.

2.2.8.2 Sensitivity wrt. Straightforward Bidding

We have conducted another set of auctions using Real Estate and Transportation value models to measure the effect of one single bidder deviating from the straightforward strategy, while the rest adheres to it. For each setting, we run 50 auctions using iBundle(3), iBundle(2) and ALPSm formats.

The results follow the same pattern over all three ICA formats and both value models. The allocative efficiency was not impacted, except that already a single level bidder reduced the efficiency significantly. The level bidder also suffered highest revenue loss of 46% of his revenue, followed by the preselect bider, who had just a minor loss of less then 5%. All other bidder types did not change the auction outcome significantly. This indicates that the equilibrium, which brings significant increase in revenue to level bidders when all bidders follow this strategy, is not stable.

2.2.8.3 Threshold problem

The threshold problem is characterized by a situation where several local bidders are competing against a global bidder. The local bidders are interested in individual lots or small bundles, and the global bidder tries to win a larger set of lots. A successful auction design shall help the local bidders to coordinate their bids in order to win against the global bidder in case such an allocation is efficient.

We analyzed different ICA designs with respect to the threshold problem using the Real Estate value model with dedicated local bidder for each of the nine lots and two global bidders. The valuations were selected such that they impose a high competition between local and global bidders. In 20 selected instances the efficient allocation included the nine local bidders and not the global bidders and the allocation with global bidders was within 10% from the optimal solution. We focused on the straightforward, the forgetful, the heuristic, and the powerset10 bidder.

ICA Format / Bidding Agent	CreditDebit	dVSV	iBundle(3)	iBundle(2)	ALPSm	Clock	VCG
Straightforward	20	20	20	18	19	20	20
Forgetful	1	1	6	5	19	18	
Powerset10	1	1	4	2	19	19	
Heuristic	0	0	4	4	20	20	

Table 2.4: Number of instances of the threshold problem in iterative combinatorial auctions, where small bidders actually won

As Table 2.4 illustrates, both linear and non-linear auction formats can solve the threshold problem with straightforward bidding. If the bidders deviated from straightforward bidding, the NLPPAs failed to solve the threshold problem in most cases. The reason is that the NLPPAs do not preserve information about bids submitted in previous rounds. Only straightforward bidders always include old bids with updated prices since the demand set can only grow throughout the auction given the NLPPA price update rule. Strong activity rules, as suggested by Mishra and Parkes (2007), can be used to force bidders to conform to the straightforward bidding strategy and improve performance of NLPPAs in this situation. However such strong rules require the bidder to evaluate and rank all bundles in advance and virtually transform the auction in to a sealed-bid format, thus eliminating the advantages of an iterative preference elicitation process.

2.2.8.4 Speed of Convergence

The CC auction had the lowest number of auction rounds in all treatments. On average, NLPPAs took three times as many rounds as linear-price based auctions (Figure 2.4). In contrast to the theory that expects a lower number of auction rounds in dVSV compared to iBundle(3), we observed a higher number of rounds in dVSV. This happens because the minimally undersupplied set is not unique and we used the smallest possible minimally undersupplied set we found, which resulted in small price steps. For the same reason, non-straightforward strategies caused the highest increase in rounds for dVSV and CreditDebit auctions, compared to other formats. The speed of convergence of these two formats can be improved by increasing prices on several disjunct minimally undersupplied sets in every round.

2.2.8.5 Impact of Increasing Competition

Auctions are expected to yield more revenue if there is more competition. Table 2.5 presents results of different auction formats using the Real Estate value model and a varying number of bidders. Each number represents an average of 10 auctions with same setting and different random seeds for the value model. We observed the expected behavior in NLPPAs and in the ALPSm design. The average revenue share in the CC auctions decreased from 5 to 7 bidders. Linear price-based auctions and the iBundle design showed a lower number of rounds with an increasing number of bidders. In contrast, the

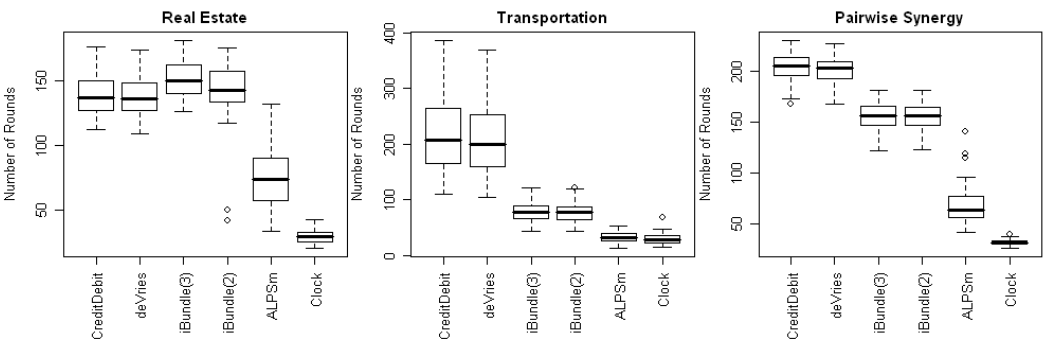


FIGURE 2.4: Rounds needed by iterative combinatorial auctions

dVSV and CreditDebit auctions showed a massive increase of rounds. This is explained by different price update mechanisms. The iBundle design, which increases prices for all unhappy bidders, will generally increase more prices when the competition is higher. The dVSV and CreditDebit auctions, which increase prices for a minimally undersupplied set of bidders, will be able to find a smaller minimally undersupplied set (typically with only one bidder) when the competition increases, and therefore increase prices for less bundles in each round.

Bidding Agent	ICA Format	Credit-Debit	dVSV	iBundle (3)	iBundle (2)	ALPSm	Clock	VCG truth
4 bidders	Efficiency in %	99.74	100.00	100.00	100.00	96.69	95.88	100.00
BSC fulfilled	Rev. Auctioneer in %	78.49	80.34	80.34	79.96	84.17	73.82	79.96
100 %	Rev. all bidders in %	21.25	19.66	19.66	20.04	12.52	22.06	20.04
	Rounds	195.84	201.46	83.40	83.40	35.66	103.06	1.00
5 bidders	Efficiency in %	99.94	100.00	100.00	100.00	96.52	95.06	100.00
BSC fulfilled	Rev. Auctioneer in %	84.48	84.94	84.94	83.16	88.09	77.27	83.16
90 %	Rev. all bidders in %	15.46	15.06	15.06	16.84	8.43	17.79	16.84
	Rounds	148.96	150.10	143.72	146.28	31.14	68.92	1.00
6 bidders	Efficiency in %	99.91	100.00	100.00	100.00	94.74	97.06	100.00
BSC fulfilled	Rev. Auctioneer in %	87.00	87.20	87.39	85.42	88.04	82.44	85.42
50 %	Rev. all bidders in %	12.91	12.80	12.61	14.58	6.69	14.62	14.58
	Rounds	132.58	133.42	207.10	209.62	30.68	61.86	1.00
7 bidders	Efficiency in %	99.89	100.00	100.00	100.00	94.35	96.98	100.00
BSC fulfilled	Rev. Auctioneer in %	88.29	88.61	88.79	86.38	87.94	84.45	86.38
40 %	Rev. all bidders in %	11.59	11.39	11.21	13.62	6.41	12.54	13.62
	Rounds	122.06	122.82	271.46	274.54	29.92	52.58	1.00

TABLE 2.5: Comparison of ICAs with differing competition levels

2.2.8.6 Impact of BSC

We have discussed that if BSM is satisfied, NLPPAs will lead to Vickrey prices. Due to computational reasons, we have restricted ourselves to analyze the somewhat weaker BSC condition only. The results based on the Real Estate value model, where the BSC condition was fulfilled in approximately half of the randomly generated instances, and all agents were following the straightforward strategy are summarized in Tables 2.6 and 2.7, as well as Figure 2.5. As expected, prices and consequently revenue were higher than the VCG outcome in NLPPAs. For linear-price auctions, the impact was not significantly different.

Revenue \ ICA Format	iBundle(2)	iBundle(3)	dVSV	CreditDebit	Clock	ALPSm
Min	98.86	100.00	100.00	100.00	102.40	86.68
Mean	99.87	100.00	100.00	100.00	107.51	97.25
Max	101.72	100.00	100.00	100.00	120.81	116.51

Table 2.6: Revenue in % to the VCG outcome, in the Real Estate value model with BSC fulfilled

Revenue \ ICA Format	iBundle(2)	iBundle(3)	dVSV	CreditDebit	Clock	ALPSm
Min	96.55	100.52	100.52	100.00	95.71	84.56
Mean	102.48	103.31	103.30	100.00	111.39	98.61
Max	112.34	111.69	111.69	100.00	127.99	119.25

Table 2.7: Revenue in % to the VCG outcome, in the Real Estate value model with BSC not fulfilled

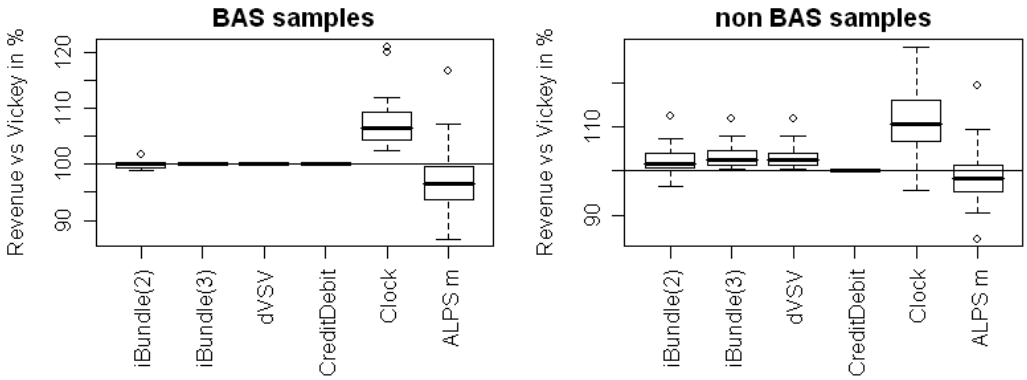


Figure 2.5: Impact of BSC on prices for straightforward bidding with the Real Estate value model

2.2.8.7 Summary

Interestingly, we had similar results regarding efficiency, auctioneer's revenue share, and auction rounds across all value models. Only the Transportation value model was different wrt. the high level of ask prices compared to the VCG auction, and also its stability against preselect bidding. The main reason was the low number of bundles with significant competition that is due to the underlying topology of transportation networks. For the same reason, the level bidders could successfully collude and significantly increase their payoff.

Linear price auctions were robust against all strategies, except the level strategy, which assumes collusive behaviour and makes it difficult for any auctioneer to select an efficient solution in general. NLPPAs were robust against forgetful bidders, but at the expense of a high number of bidding rounds. There were significant efficiency losses in NLPPAs with heuristic bidders, powerset, level, and preselect bidders.

dVSV and CreditDebit auctions have a significantly lower efficiency than iBundle(3) with heuristic and powerset bidders. The main difference between these formats is the set of bidders, for which the ask prices are increased. Increasing the ask prices on a minimally undersupplied set of bidders is less robust against these strategies. Note that Proxy agents, which can be used to enforce straightforward bidding Mishra and Parkes (2004), cannot detect and prevent level and preselect bidding strategies.

2.2.8.8 Conclusion

NLPPAs such as iBundle(3), dVSV, iBEA, and CreditDebit auctions have greatly advanced our understanding for the design of efficient auction mechanisms in the realm of private valuations. These formats are modeled after well-known optimization algorithms that lead to efficient solutions, provided that bidders follow the straightforward strategy.

Since these are exact algorithms, the auction generally requires many rounds, where all 2^m valuations of all losing bidders are elicited. Both the high number of auction rounds and the necessity of the straightforward bidding strategy motivate the use of proxy agents, which need to be hosted by a trusted third party, which essentially reduces the auction to a sealed-bid event for the bidders. While there are incentives for bidders to follow the straightforward strategy in these auctions, it is not always acceptable to use proxy agents, nor can they prevent certain strategies, such as level bidding.

In contrast, linear price combinatorial auctions follow a more heuristic approach to find the optimal solution. While our results achieve high efficiency values on average, one can easily construct examples, where linear price CAs cannot be efficient Bichler et al. (2009). There are a few remedies, such as the proxy phase in the Clock-Proxy auction Ausubel et al. (2006) that addresses these inefficiencies, but these designs have not yet been thoroughly analyzed Blumrosen and Nisan (2005).

Linear-price designs suffer from the fact that they cannot be 100% efficient, but they have shown to be more robust against many strategies and bear a few advantages. The main advantage is the a linear number of prices needed. This reduces the communication from the auctioneer to the bidders and presents a guideline to help bidders find profitable items and bundles Kwon et al. (2005). They also exhibit a low number of auction rounds compared to NLPPAs.

Overall we have seen that if bidders do not express their valuations correctely, the performance of all auction formats suffers. Suboptimal bidding (with respect to efficiency) may happen because of strategic considerations in some cases but even if the bidders want to express their valuations correctely they might fail because of the complexity to do so.

2.2.9 Applicability to Procurement Auctions

Multi-unit combinatorial auctions, will likely lead to economic inefficiencies for procurement settings, as bidders can realistically only express a very small fraction of their bids of interest. For a combinatorial auction with $\mathcal{I}$ items, a bidder can already submit $2^{\mathcal{I}}-1$ possible bids. If you also allow for k quantities of each item, the number of possible bids grows even faster ($\phi(\mathcal{I},k) = \sum_{j=0}^{\mathcal{I}-1} k^{\mathcal{I}-j}\binom{n}{j}$). For example, with 10 items it would be possible to specify 1023 bids in a single-unit combinatorial auction. However, with 6 allowable quantities for each item the supplier can possibly specify more than 282 million bids. Clearly, bidders will only be able to submit a small proportion of the possible bids of interest, and the auctioneer will most likely not find the efficient, maybe not even a feasible allocation.

There are two possible remedies: Either the auctioneer simplifies the auction and reduces the number of possibilities a bidder can bid on through prebundling or other forms of simplification Milgrom (2009), or he provides a more powerful bidding language that allows to describe his preferences in a concise way.

2.3 Volume Discount Auctions

Volume discount bids allow for a compact representation of bids in markets with economies of scale. These bid types define unit prices for specific volume intervals as stepwise linear functions.

2.3.1 Bidding Languages

The earlier literature on supplier selection and volume discounts includes studies of various discounting schemes, such as unit discounts Silverson and Peterson (1979), inventory models with demand uncertainty and incremental quantity discounts and carload quantity discounts Jucker and Rosenblatt (1985); Lee and Rosenblatt (1986). Munson and Rosneblatt (1998) provide a perspective on discounts used in practice, while Chaudhry et al. (1993) discuss a vendor selection model in the presence of price breaks.

Davenport and Kalagnanam (2000) were among the first authors to focus on auctions with incremental volume discount bids. An application thereof has been described in Hohner et al. (2003). Their bidding language requires suppliers to specify continuous supply curves for each item. Eso et al. (2001) further advance the ideas described in Davenport and Kalagnanam (2000) and allow for discontinuities and decreasing slopes in the bids.

There has also been some work on total quantity discounts, where the unit price starting at a particular quantity is charged for the entire quantity purchased, not only for the units above the threshold quantity. Katz et al. (1994) discuss a procurement decision support system and a respective mathematical program with total quantity discounts. Crama et al. (2004) investigate a problem, where a chemical company needs to purchase a number of ingredients from one or more suppliers with a total quantity discount. Here, only one discount rate is used for all ingredients. Crama et al. (2004) also need to decide how to use the purchased ingredients to manufacture the desired quantities of the end products, where there are alternative recipes, which is different to the problem analyzed in this thesis.

2.3.2 Winner Determination

While the academic literature is still in its infancy, a number of companies such as CombineNet, Emptoris, Iasta, and TradeExtension Gartner (2008) provide decision support systems allowing for various types of discounts and complex bid types. These tools enable purchasing managers to explore different award scenarios based on various operational or strategic side constraints. Respective software vendors offer a wide variety of constraints among several dozens or even hundreds of constraint classes Bichler et al. (2006).

The general problem has been discussed in Goossens et al. (2007). They provide an interesting contribution to the supplier selection problem with total quantity discounts and a proof, showing that no polynomial-time approximation scheme with constant worst-case ratio exists for this supplier selection problem and that the decision version is strongly $\mathcal{NP}$-complete.

A complexity analysis of a similar allocation problem with linear demand curve bids can be found in Sandholm and Suri (2001). These types of bids describe piecewise linear unit-price functions of the demand and the authors argue that they can approximate any function arbitrarily close. Note that also step unit-price functionsm, as the ones that will be discussed in this paper, become piecewise linear total cost functions. Dang and Jennings (2003) discussed market clearing in multi-unit combinatorial auctions with supply function bids, where suppliers specify quantities and prices for packages of items.

2.3.3 Open Issues

We have worked with a number of procurement managers in the past few years to evalute why there is very little application. While the above academic work covers important requirements, many real-world cases demand for a more comprehensive bidding language for practical applicability. In many applications, some suppliers provide incremental volume discount bids, others total quantity discount bids, or overall lump-sum discounts on total spend.

As outlined, in this work, we considerably extend the expressiveness of the bidding languages discussed in the literature and propose a mixed integer programming formulation to solve practically relevant problem sizes. In contrast to previous work, we suggest a parametric multi-product cost function to generate realistic bids and analyze different discount policies based on cost and description length.

Chapter 3

The L_{ESS} Bidding Langue

3.1 Bidding Language

In this section, we will introduce a bidding language allowing to describe a supplier's cost function. We focus on markets with economies of scale and scope, where bidders typically express discounts in order to reflect these economic characteristics. A language in computer science and logic assigns a semantic to a syntax. Bidding languages have been studied in the context of combinatorial auctions Abrache et al. (2004); Boutilier and Hoos (2001); Nisan (2006). However, this research typically focuses on multi-item, but single-unit negotiations.

3.1.1 Description Length of Bidding Languages

There is a vast literature in Microeconomics and Econometrics on modeling and estimation of multi-product cost functions. Such multi-product cost functions describe the cost of production $c_s : \mathbb{R}^{\mathcal{I}} \rightarrow \mathbb{R}$ as a function of the total quantity produced in all products or items $i \in \mathcal{I}$ for a supplier $s \in \mathcal{S}$. Depending on industry and company specifics, different types and parametric shapes of such cost functions can be found, sometimes explicitly given based on data from cost accounting. A firm has economies of scale (i.e., is operating in a downward sloping region of the average cost curve) if and only if it has increasing returns to scale. Likewise, it has diseconomies of scale (is operating in an upward sloping, convex region of the average cost curve) if and only if it has decreasing returns to scale.

Bidding languages should be expressive enough to allow for the description of different shapes that multi-product cost functions can assume. At the same time, the bidding language should allow to describe bids in a compact way with only a few parameters. In general, expressivity of a bid language should increase efficiency of economic mechanisms, since bidders are able to better describe their preferences (Benisch et al., 2008; Sandholm, 2008). Combinatorial auctions allow bidders to express all types of synergies across items, but the number of possible bids in large-scale combinatorial auctions is typically beyond what human bidders can express. There is a number of articles on the communication complexity of such auctions Nisan and Segal (2001). As we have discussed in Section 2.2.9, this phenomenon becomes even worse with multi-unit combinatorial auctions. Compact bidding languages, which allow bidders to describe their preferences on multiple items and quantities as a function, can alleviate this problem.

Research in Artificial Intelligence has long dealt with questions of adequate knowledge representation and reasoning for a particular application domain. The most important decision to be made, is the *expressivity* of the knowledge representation. Typically, the more expressive languages are, the harder it is to automatically derive inferences. In a general way, this has been shown for formal languages, where regular languages (type-3) can be decided in linear time, whereas the general class of recursively enumerable languages (type-0) require a Turing machine. In logic, the boolean satisfiability problem (SAT) is a well-known NP-complete decision problem, whereas reasoning in propositional Horn logic (HORNSAT) is P-complete. In contrast, satisfiability of first-order Horn clauses is undecidable Sowa (2000). So, there is also a trade-off between the expressivity of a bidding language and the time to solve respective allocation problems using different inference mechanisms.

The pricing rules introduced in the $\mathcal{L}_{ESS}$ bidding language could be described as Horn clauses. In this paper, we will solve the allocation problem as an MILP and not as a logic program, as MILPs are geared to optimization with discrete and continuous variables, which is required in our case, while logic programming has its strength when reasoning on discrete logic variables. There is a substantial literature on the relationship between mixed integer linear programs (MILPs) and logic programs. For example, (Chandru and Hooker, 1999) showed that many logical inference problems can be solved as a relaxed linear program even though they are not Horn clauses.

We will address the hardness of the winner determination problem given an $\mathcal{L}_{ESS}$ bidding language in the next section. For now, we want to concentrate on the expressiveness of a bidding language. Benisch et al. (2008) try to define general notions of expressiveness of economic mechanisms. This is an important new strand in the literature, as many of the strategic complexities and the inefficiencies of popular auction mechanisms such as split-award auctions or the simultaneous multi round auction (SMR) can be attributed to the limited expressiveness of the bidding language used. A good example is the exposure problem arising in SMR that can be solved by allowing for package bids in combinatorial auctions (Cramton et al., 2006).

Milgrom (2009) also shows that in certain settings additional expressiveness can give rise to additional unwanted equilibria and poor efficiency. Therefore, he argues for simplified mechanisms with restricted message spaces and shows that, if carefully designed, simplification can avoid unwanted equilibria, without hurting efficiency. Finding good simplifications, however, is difficult without a good understanding of the bidders' valuations or cost functions.

Definition 10. *A* compact bidding language *allows to define the bid price as a function* $p_s : \mathbb{R}^{\mathcal{I}} \rightarrow \mathbb{R}$ *of quantity for one or more items* $i \in \mathcal{I}$*.*

We will also use the term *bid function* for p_s in this paper. Such a bid function has a particular parametric form. Clearly, the most compact format would be to reveal the parameters and the specification of the true total cost function in a direct revelation mechanism to an auctioneer. For example, we will use a multi-product cost function with eight parameters for the experimental evaluation in section 5.1. The parametric shape of such cost functions might be non-linear and different among suppliers and industries, which is just one of the reasons, why the true specification of the function is typically not revealed in practice. It is rather common to specify volume discounts or markups for economies or diseconomies of scale. Such volume discounts can be formulated as piecewise-defined linear functions, which approximate the underlying cost curve. Therefore, such bids are typically an approximation of the underlying cost function, even if bidders are willing to reveal their costs truthfully. Let us now introduce *description length* as a measure for how compact bidders can describe information about their preferences or their underlying cost function.

Definition 11. *The* description length of a bid *consists of the bits to describe the parameters of the bid function* p_s *for a given maximum approximation error,* ϵ^{max}*, to the underlying utility or cost function.*

Bidding languages should allow for a close approximation of wide-spread types of cost functions, but the same time have a low description length. A bad approximation will make it difficult for the auctioneer to find an efficient allocation. Multi-unit bundle bids in combinatorial auctions allow for close approximations, as they only specify discrete points but at the expense of a huge number of bids required to describe a bidder's costs.

3.1.2 The $\mathcal{L}_{ESS}$ Bidding Language

Existing bid types in the literature support specific types of either incremental volume discount or total quantity discounts, such as in Davenport and Kalagnanam (2000) and Goossens et al. (2007). In contrast, bid languages we observed in practice exhibit substantial structural variation across bidders. Offers from suppliers can come in any combination of incremental or total quantity discounts, depending on multiple conditions. To accommodate the richness observed in practice one needs a language that allows for different discount types, denoted R_d.

In case of diseconomies of scale, for example, when the volume awarded is beyond the production capacity of a supplier, he might want to charge respective *markups* R_m to cover his increased per-unit costs. While markups are conceptually equivalent to volume discounts, we will use a separate notation R_m in the next section, as they need to be modeled differently in our MIP.

In addition, we regularly observed *lump sum discounts* R_l, which describe refunds of part of the total price. For example, if the volume purchased exceeds a threshold, a supplier might be willing to reduce the overall payment by a fixed amount $R_l = \$10,000$ on the total price. These lump sum discounts R_l can also be defined on spend S_l or quantity Q_l and are often used to describe economies of scope.

In $\mathcal{L}_{ESS}$, bidders should be able to express such different types of discounts. Every supplier $s \in \mathcal{S}$ submits a base price $P_{i,s}$ for every item $i \in \mathcal{I}$ and the maximum quantity $E_{i,s}$ he is willing to supply. In addition, he specifies volume discounts $d \in \mathcal{D}$, lump sum discounts $l \in \mathcal{L}$, and markups $m \in \mathcal{M}$ to modify the base price based on certain *spend conditions*. The total bid price function $p_s : \mathbb{R}^{\mathcal{I}} \to \mathbb{R}$ of a set of items $\mathcal{I}$ can be written as

$$p_s(x_1, ..., x_{\mathcal{I}}) = \sum_{i \in \mathcal{I}} P_{i,s} x_{i,s} - \sum_{d \in \mathcal{D}} R_d y_d 1_{\{C_d\}} + \sum_{m \in \mathcal{M}} R_m y_m 1_{\{C_m\}} - \sum_{l \in \mathcal{L}} R_l 1_{\{C_l\}}$$

where R_d describes a *volume discount* per unit that is awarded on a quantity y_d if a *spend condition* C_d is true, e.g., after the quantity exceeds a certain lower bound on quantity, Q_d, or spend, S_d. Note that Q_d and S_d can be defined on a particular item provided by the supplier or also a set of items by this supplier. We will use the term "discount interval" and refer to spend conditions, which define a unit price for a particular quantity interval. Volume discounts of a specific bidder can be valid for the total quantity purchased (*total quantity discounts*) or for the amount exceeding a pre-specified threshold (*incremental (volume) discounts*). This also holds for markups R_m. Lump sum discounts R_l are defined on overall spend or quantity, not on per unit.

Spend conditions (C) are an important language feature, which allow for much flexibility. By allowing conditional discounts and markups with possibly multiple conditions, we are able to formalize all features of bids that have been considered in the literature or we have encountered in practice. Spend conditions can be defined on a set of items and be based on spend (S) or volume (Q) purchased. For example, if spend on the items A and B is more than \$100,000, a supplier offers an lump sum discount of $R_l = \$4,000$. An elementary spend conditions C is treated as a literal in propositional logic such as $S_{A,B} > 100,000\$$ or $Q_A > 2000$. Composite spend conditions take the form of a conjunction of m elementary conditions. So in general, a discount rule takes the form of a Horn clause, with $C_1 \wedge C_2 ... \wedge C_k \implies R$. We limited discount rules to Horn clauses, in order to keep the corresponding supplier quantity selection problem of the auctioneer as concise as possible (see section 4.1). Modeling disjunctions and conjunctions in the condition of such a discount rule is possible as well, but rarely necessary as we found.

Definition 12 (Discount rule)**.** *A discount rule F is a Horn clause of the form $C_1 \wedge C_2 ... \wedge C_k \implies R$, where*

- *C_k is a literal defined on spend levels or quantity levels for a set of items, with $k = \{1, .., \mathcal{K}\}$ being the set of respective spend conditions.*
- *R is a discount, i.e., a lump sum discount, an incremental volume discount, a total quantity discount, or a respective markup.*

The supplier is also able to specify disjunctive discount rules, i.e., two or more rules cannot be active at the same time. For example, in case the supplier purchases more than \$10,000 from item A and B, there is a discount of \$1.77. Alternatively, if the supplier buys more than 5,000 units from item A, the discount is \$1.02. Only one of these two discounts is eligible, and the auctioneer will choose the discount that minimizes his total cost.

Definition 13. *An $\mathcal{L}_{ESS}$ bid is a tuple ($\mathcal{P}$, $\mathcal{E}$, $\mathcal{H}$, $\mathcal{K}$), where*

- *$\mathcal{P}$ is a set of base unit prices for each item $i \in \mathcal{I}$,*
- *$\mathcal{E}$ is a set of maximum quantities $E_{i,s}$ that a supplier s can provide for each item $i \in \mathcal{I}$,*
- *$\mathcal{H}$ is a set of discount rules $F \in \mathcal{H}$, and*
- *$\mathcal{K}$ is a set of disjunctions specified on the set of rules in $\mathcal{H}$.*

$\mathcal{L}_{ESS}$ provides expressiveness at low description length by allowing to express step functions to describe the average unit costs of a supplier, or the respective piecewise linear functions describing the total cost function $c_s(x)$.

Volume discounts of this sort are also referred to as second degree price discrimination according to Pigou (1946). In $\mathcal{L}_{ESS}$ we ignore first or third degree price differentiation, as these pricing policies differentiate among individual buyers or buyer segments, and we focus on bids that are sent to a particular buyer as a response to a request for quotation (RfQ) or in a reverse auction. We will also ignore product differentiation, where suppliers submit information on multiple qualitative attributes such as product attributes, terms of delivery and payment that might differentiate one supplier from another. Often, these attributes are standardized in an RfQ in order to avoid having to trade-off multiple qualitative attributes and price. Bidding languages for configurable and multi-attribute auctions have been discussed by Bichler and Kalagnanam (2005). According to Bichler et al. (2002), who distinguish among three dimensions in a business-to-business procurement negotiation, $\mathcal{L}_{ESS}$ is suitable for *multi-item, multi-unit negotiations*.

Chapter 4

The Supplier and Quantity Selection

In the following, we will investigate a buyer's problem, who needs to select quantities from each supplier providing bids in $\mathcal{L}_{\mathcal{ESS}}$ such that his costs are minimized and his demand is satisfied. We will refer to this problem as *Supplier Quantity Selection ($\mathcal{SQS}$)* problem and introduce a respective mixed integer program in the following.

4.1 The Supplier Quantity Selection Problem Formulation

We will first introduce some necessary notation. We will use uppercase letters for parameters (table 4.2), lowercase letters for decision variables (table 4.1), and calligraphic fonts for sets (table 4.3). Sets indexed by a member of another set represent the subset of all elements that are relevant to the index.

$$
\begin{array}{llll}
\min \sum_{i \in \mathcal{I}} \sum_{s \in \mathcal{S}} P_{i,s} x_{i,s} - \sum_{d \in \mathcal{D}} R_d y_d + \sum_{m \in \mathcal{M}} R_m y_m - \sum_{l \in \mathcal{L}} R_l c_l + \sum_{k \in \mathcal{K}} R_k c_k & & & \\
s.t. & & & \\
\sum_{S \in \mathcal{S}} x_{i,s} \geq W_i & \forall i \in \mathcal{I} & & (1) \\
x_{i,s} \leq E_{i,s} & \forall i \in \mathcal{I}, \forall s \in \mathcal{S} & & (2) \\
\sum_{i \in \mathcal{I}_d} x_{i,s_d} - D_d c_d \geq y_d & \forall d \in \mathcal{D} & & (3d) \\
\sum_{i \in \mathcal{I}_m} x_{i,s_m} + Bc_m - D_m \leq y_m + B & \forall m \in \mathcal{M} & & (3m) \\
Bc_d \geq y_d & \forall d \in \mathcal{D} & & (4d) \\
\sum_{n \in \mathcal{N}_d} j_n - \sum_{e \in \bar{\mathcal{E}}_d} c_e \geq |N_d| \, c_d & \forall d \in \mathcal{D} & & (5d) \\
\sum_{n \in \mathcal{N}_l} j_n - \sum_{e \in \bar{\mathcal{E}}_l} c_e \geq |N_l| \, c_l & \forall l \in \mathcal{L} & & (5l) \\
|N_m|^{-1} (\sum_{n \in \mathcal{N}_m} j_n + 1) - \sum_{e \in \bar{\mathcal{E}}_m} c_e \leq c_m + 1 & \forall m \in \mathcal{M} & & (5m) \\
|N_k|^{-1} (\sum_{n \in \mathcal{N}_k} j_n + 1) - \sum_{e \in \bar{\mathcal{E}}_k} c_e \leq c_k + 1 & \forall k \in \mathcal{K} & & (5k) \\
\sum_{i \in \mathcal{I}_n} P_{i,s_n} x_{i,s_n} - \sum_{d \in \mathcal{D}_n} R_d y_d + \sum_{m \in \mathcal{M}_n} R_m y_m \geq S_n j_n & \forall n \in \mathcal{N} & & (6l, d) \\
\sum_{i \in \mathcal{I}} x_{i,s_n} \geq Q_n j_n & \forall n \in \mathcal{N} & & (7l, d) \\
\sum_{i \in \mathcal{I}} x_{i,s_n} - Q_n < B j_n & \forall n \in \mathcal{N} & & (8m, k) \\
x_{i,s}, y_d, y_m \geq 0 & \forall i \in \mathcal{I}, \forall s \in \mathcal{S}, \forall d \in \mathcal{D}, \forall m \in \mathcal{M} & & \\
c_d, c_m, c_l, c_k, j_n \in \{0, 1\} & \forall d \in \mathcal{D}, \forall l \in \mathcal{L}, \forall n \in \mathcal{N}, \forall m \in \mathcal{M}, \forall k \in \mathcal{K} & &
\end{array}
$$

The objective function minimizes the product of all base prices $P_{i,s}$ and quantities $x_{i,s}$ of item i purchased of supplier s, subtracts the sum of all discounts R_d and lump sum discounts R_l, and adds the markups R_m and lump sum markups k.

The first constraints (1) ensure that the demand W_i is fulfilled, and the second set of constraints (2) that the amount purchased from a product i does not

Decision Variables			
Varaible	Description	Range	Occurrence
$x_{i,s}$	Amount of item i bought from supplier s	$\in \mathbb{N}_0$	$\|\mathcal{S}\| * \|\mathcal{I}\|$
y_d	Amount active in discount d	$\in \mathbb{N}_0$	$\|\mathcal{D}\|$
y_m	Amount active in markup m	$\in \mathbb{N}_0$	$\|\mathcal{M}\|$
c_d	Indicator for discount d	$\in \{0,1\}$	$\|\mathcal{D}\|$
c_m	Indicator for markup m	$\in \{0,1\}$	$\|\mathcal{M}\|$
c_l	Indicator for overall dicount l	$\in \{0,1\}$	$\|\mathcal{L}\|$
j_n	Indicator for condition n	$\in \{0,1\}$	$\|\mathcal{N}\|$

TABLE 4.1: List of Variables in $\mathcal{SQS}$

exceed the maximum quantity $F_{i,s}$ provided by each supplier of each quantity. The constraint sets (3d) and (3m) determine the relevant volume y_d or y_m, for which the discount or markup resp. is defined, and B is a sufficiently large number. For example, if $D_d = 0$, then (3d) defines a total quantity discount, where $y_d = x_{i,s}$, otherwise, D_d is set to the threshold, after which the volume discount is valid, as such describing an incremental volume $y_d = x_{i,s} - D_d$. Typically, the discount intervals and markups hold for a single item, but they can also be defined on multiple items $i \in \mathcal{I}_d$.

Parameters	
Parameter	Description
$P_{i,s}$	Base price for item i form supplier s
R_d	Amount (price decrease) of discount d
R_m	Amount (price increase) of markup m
R_l	Amount (lump sum payment) of lump sum discount l
R_k	Amount (lump sum payment) of lump sum markup k
W_i	Demand (want) for item i
$F_{i,s}$	Quantitiy that supplier s can provide of item i
D_d	Displacement for discount d
D_m	Displacement for markup m
S_n	Minimal spend for spend condition n
Q_n	Minimal quantity for spend condition n
B	Big (enough) number

TABLE 4.2: List of Parameters in $\mathcal{SQS}$

For each discount rule, we introduce a binary variable c_d, c_l, and c_m. Such decision variables are determined based on spend conditions, which we define in constraint sets (6-8). Constraint (4) makes sure that a discount is only provided ($y_{i,s} > 0$) if the respective binary variable for this discount, c_d, is true. Constraint sets (5d), (5l), and (5m) make sure that if a particular set of spend conditions is given ($j_n = 1$), which are a precondition for a discount, markup, or lump sum discount, then also the respective binary variable c_d, c_l, or c_m is true. $|N_d|$, $|N_m|$, and $|N_l|$ describe the number of conditions that need to be true for the respective binary variable to become true. These constraints also allow to specify sets of rules $\bar{\mathcal{E}} \subset \mathcal{D} \cup \mathcal{M} \cup \mathcal{L} \cup \mathcal{K}$, which cannot be active at the same time as the respective rule.

Sets	
Set	Description
$\mathcal{S}$	Set of all suppliers
$\mathcal{I}$	Set of all items
$\mathcal{D}$	Set of all discounts
$\mathcal{M}$	Set of all markups
$\mathcal{L}$	Set of all lump sum discounts
$\mathcal{K}$	Set of all lump sum markups
$\mathcal{I}_n$	Set off all items included in spend condition n
$\mathcal{N}$	Set off all spend conditions
$\mathcal{N}_d$	Set off all spend conditions necessary for discount d
$\mathcal{N}_m$	Set off all spend conditions necessary for markup m
$\mathcal{N}_l$	Set off all spend conditions necessary for lump sum discount l
$\mathcal{N}_k$	Set off all spend conditions necessary for lump sum markup+ k
$\bar{\mathcal{E}}_d$	Set off all pricing rules that disable discount d
$\bar{\mathcal{E}}_m$	Set off all pricing rules that disable markup m
$\bar{\mathcal{E}}_l$	Set off all pricing rules that disable lump sum discount l
$\bar{\mathcal{E}}_k$	Set off all pricing rules that disable lump sum markup k

TABLE 4.3: List of Sets in $\mathcal{SQS}$

The final sets of constraints (6 - 8) models individual conditions on spend or quantity that need to be fulfilled for a particular discount rule in constraint sets (5). For example, constraint set (6 l,d) specifies a minimum spend condition for volume discounts and lump sum discounts. In words, if the total cost including markups and discounts (not considering other lump sum discounts) exceeds S_n, then an additional lump sum discount will be granted. Constraint set (7 l,d) determines a minimum quantity condition used in volume and lump sum discounts. Constraint set (8 m) defines a minimum quantity condition for a markup rule.

Goossens et al. (2007) describe the problem of selecting a set of suppliers that offer a variety of items using total quantity discounts. The problem is referred to as TQD. They have provided a polynomial reduction of 3-dimensional matching, a well-known strongly $\mathcal{NP}$-complete problem, to TQD. Showing that $\mathcal{SQS}$ is $\mathcal{NP}$-complete is straightforward, as it contains TQD as a special case.

Theorem 1. *The decision version of the $\mathcal{SQS}$ problem is strongly $\mathcal{NP}$-complete.*

Here, we refer to TQD' as the decision version of the more-for-less variant of TQD, and $\mathcal{SQS}$' as the decision version of $\mathcal{SQS}$. Any input for TQD' can be solved with $\mathcal{SQS}$', and a solution to $\mathcal{SQS}$' would solve TQD'. $\mathcal{SQS}$' is obviously in $\mathcal{NP}$ since given a solution it suffices to check the constraints and the value of the solution.

While this shows that $\mathcal{SQS}$ is at least as hard as TQD to solve, it is interesting to understand how the increased expressiveness of $\mathcal{L}_{\mathcal{ESS}}$ impacts the empirical hardness of the problem. As is often the case, the formulation of the problem matters, and there are significant differences in runtime depending on the model formulation (Hooker, 2009).

4.2 Scenario Analysis

We will focus on scenario analysis as a typical use case. During scenario analysis, procurement managers typically use additional side constraints to explore different award scenarios. For example, a purchasing manager might be interested in an optimal allocation with a maximum of 5 winners, or the optimal allocation, where the spend on an individual supplier is limited to 1 million dollars due to certain risk considerations. An ex-post analysis based on already submitted bids also allows to analyze the cost of a particular constraint by comparing the objective value of respective scenarios.

We will now discuss a number of side constraints that are important for procurement managers and used during the scenario analysis. Purchasing managers want to set a lower and an upper bound for the quantity a supplier can win overall (V_s^l, V_s^u) in (9), or on a particular item ($V_{i,s}^l$, $V_{i,s}^u$) in (10). Also, they want to limit the overall spend per winner (T_s^l, T_s^u) in (11). In constraint sets (12 & 13) the maximum number of winners L is restricted.

$$
\begin{aligned}
V_s^l \leq \sum_{l \in \mathcal{I}} x_{i,s} &\leq V_s^u && \forall s \in \mathcal{S} && (9)\\
V_{i,s}^l \leq x_{i,s} &\leq V_{i,s}^u && \forall i \in \mathcal{I}, && (10)\\
& && \forall s \in \mathcal{S} &&\\
T_s^l \leq \sum_{i \in \mathcal{I}_s} P_{i,s} x_{i,s} - \sum_{d \in \mathcal{D}_s} R_d y_{i,d} + \sum_{m \in \mathcal{M}_s} R_m y_{i,m} &\leq T_s^u && \forall s \in \mathcal{S} && (11)\\
\textstyle\sum_{i \in \mathcal{I}} x_{i,s} &\leq B a_s && \forall s \in \mathcal{S} && (12)\\
\sum_{s \in \mathcal{S}} a_s &\leq L && && (13)\\
a_s &\in \{0,1\} && \forall s \in \mathcal{S} &&
\end{aligned}
$$

The number of scenarios can be huge, and it would be interesting for procurement managers to know, which side constraints have the biggest impact on total cost. For linear programs such information is provided by the dual variables of respective side constraints. Such dual information is not readily available for mixed integer programs and IP duality is a notoriously difficult topic Guzelsoy and Ralphs (2007); Williams (1996). One can, however, fix the binary variables to their optimal values and resolve the MIP as a linear program. The resulting duals can then provide useful shadow prices for constraints such as (1), (9), (10), and (11). It can also be helpful to include the right-hand sides of certain constraints as variables, and ask whether there is an award scenario that improves the total cost by a certain percentage. This can be achieved by constraining the objective function value accordingly. Overall, sensitivity analysis can provide valuable feedback for procurement managers during scenario analysis.

Scenario analysis can be challenging for the procurement manager, as there are many different side constraints one can explore. It would be very valuable to have dual information available that indicates, which constraints are most binding, i.e., have the largest impact on total cost. Unfortunately, $\mathcal{SQS}$ is a mixed integer linear program. The integrality constraints transform the problem from a convex to a non-convex optimization problem and the linear dual problem might have an objective function value that is strictly smaller than the objective function of the primal problem. In this case, the dual variables will not provide useful information.

The OR literature has addressed the problem of finding dual price interpretations to integer programs and MIPs (Williams, 1996). The classic work in this area is Gomory and Baumol (1960). They add additional constraints to the LP relaxation of the MIP, which define linear combinations of existing constraints, until the solution results in an integer solution. The resulting dual values can be imputed back to the original constraints from which they were derived. However, these prices are not unique as they depend on the sequence of cuts introduced to the formulation and difficult to interpret economically. Guzelsoy and Ralphs (2007) provide an up-to-date survey of the literature in this field. They also describe a practical approach to sensitivity analysis on right-hand side variables, using warm-starting functionality in modern MIP solvers. The solvers return lower bounds for the problem with a modified right hand side using the information gathered from the branching tree of the original problem solved . We ran experiments with and without warm starting in the open-source solver Symphony (https://projects.coin-or.org/SYMPHONY/), which provides such functionality. However, the results were not stable enough to report.

Apart from dual information on individual constraints, it can be helpful to include the right-hand sides of certain constraints as variables, and ask whether there is an award scenario that improves the total cost by a certain percentage. This can be achieved by constraining the objective function value accordingly.

4.2.1 Price Feedback in $\mathcal{L}_{ESS}$

There is a huge literature on competitive equilibrium prices in iterative combinatorial auctions (Parkes, 2006). Linear as well as non-linear pricing concepts in this literature have been developed only for simple bundle bids and a winner determination without additional side constraints. With more complex bidding languages and multiple allocation constraints in $\mathcal{SQS}$ it is unclear, how such ask prices could be derived in $\mathcal{L}_{ESS}$. However, feedback information is important in every indirect mechanism to help bidders improve their bids and become winning in the next round.

Mechanism design questions or any game-theoretical analysis of iterative mechanisms with $\mathcal{L}_{ESS}$ is beyond the scope of this work. Nevertheless, price feedback can be derived based on the WDP. We suggest a single-dimensional ask price by calculating the height of the lump sum discount, a bidder would have had to provide, in order to become winning. For this, a bidder s can specify a certain vector of desired quantities $x_{1,s}, ..., x_{\mathcal{I},s}$ and the auctioneer responds by a respective lump sum discount R_l^{ask}. This lump sum discount can be calculated by resolving the MILP with a $x_{i,s}$ fixed to the amount that supplier s wants to win on each item $i \in \mathcal{I}$. This is similar in spirit to the winning levels described in Adomavicius and Gupta (2005) for combinatorial auctions, although their calculation is different. Whether it is possible to perform scenario analysis or derive respective ask prices in an interactive manner depends on the time to solve realistic problem sizes. In the following, we will report on experiments to analyze the empirical hardness of these problems.

Chapter 5

Experimental Setup

In this chapter we will introduce a valuation model called *"CoP - Cost of Production"* that generates instances that we will use in our experiments and describe how we set the experiments up. We focused on the two main performance meters we want to examine: the computation time dependent on the problem size and the savings in spend versus simpler mechanisms.

5.1 Value Model "Cost of Production"

The key component in an experiment with computational simulations is the quality of the generated test data. As described by Leyton-Brown et al. (2000) in their work about a test suite for combinatorial auctions "the lack of standardized, realistic test cases does not make it impossible to evaluate or compare algorithms, it does make it difficult to know what magnitude of real-world problems each algorithm is capable of solving, or what features of real-world problems each algorithm is capable of exploiting".

In their environment a test case consists of a set of bids for which the revenue maximizing allocation shall be determined. In order to generate them they use a model, called value model from here on, that provides a numerical value for every possible bundle describing the value of it. This so called valuation can then, depending on the assumed strategy of the bidders, translated into bids. As them we will "assume a sealed-bid incentive-compatible mechanism, where the price offered [...] is equal to the bidders valuation." In a combinatorial auctions setting, for every bundle and bidder a bid is placed with the bid price set to the valuation of the bidder. As we have seen in section 2.2.9 this would lead to an unmanageable huge number of bids in our setting, and we want to interpolate the cost function with fewer bids. Naturally the interpolation method and quality have a big impact on the performance of the supplier and quantity selection algorithm.

In order to make our value model realistic and close to business reality, we derived it from the theory of cost of production in economics, as described for example in Pindyck and Rubinfeld (2005). The cost of production is not the only factor impacting the price of a good, but according to Kalecki (1991) there are industries where the price is cost-determined.

5.1.1 Composition of Cost

The total cost describes the total economic cost of producing a given quantity of an object. The graph of the cost of producing a given quantity of an object is called a cost curve. There are several types of cost curves, which are all related to each other. The most basic ones are total cost curves, plotting the cost to produce a quantity of an object against the quantity produced. Average total cost curves capture the relationship between the cost of producing one copy of the object and the quantity that is produced. Finally marginal cost curves are the first derivative of a total cost curve and represent the relation between marginal (i.e., incremental) cost incurred and the quantity of output produced.

The total cost is usually considered to be composed of fixed and variable cost. In economics it is usually also differed between long and short run cost, but for us only the short run cost is of interest, as it influences the prices directly.

5.1.1.1 Fixed Cost

Fixed costs differ from variable costs, as they are independent of the quantity produced. They are not permanently fixed as sunk costs, but in relation to the quantity of production for the relevant period. Consider the example of a small logistics company where the cost of leasing a truck to deliver goods is present independently of the amount of delivery done with it. As one can see in figure 5.1 fixed costs lead to flat total costs and marginal costs that have only a peak at the first unit but decreasing average total cost. Fixed costs that

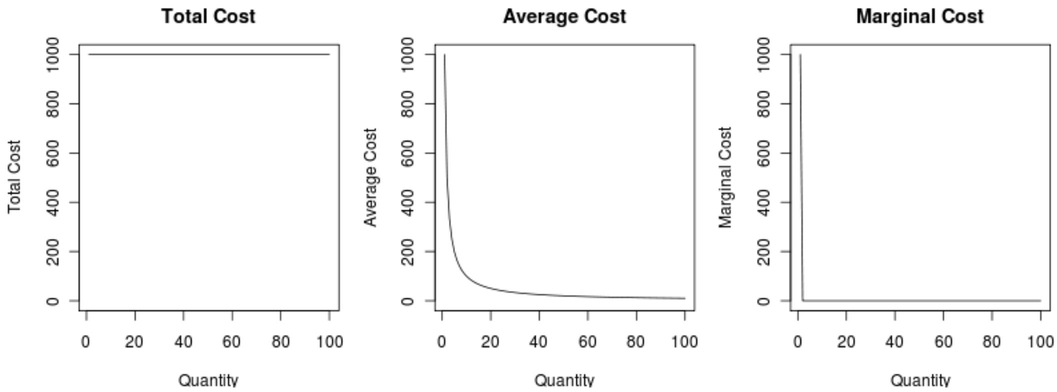


FIGURE 5.1: Total, average and marginal costs dependent on the number of units purchased if only fixed costs of 1000 $ are present

can be spread against a bigger quantity are the primary sources of economies of scale.

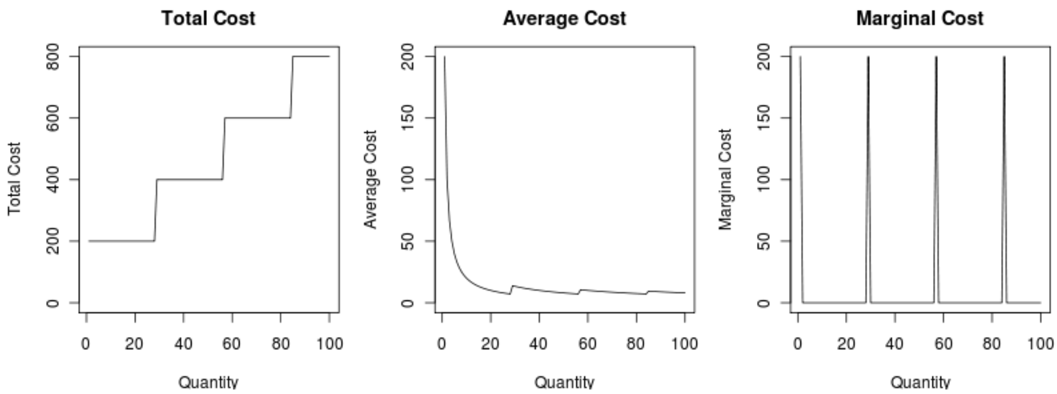


FIGURE 5.2: Total, average and marginal costs dependent on the number of units purchased if only stepwise fixed costs of 200 $ are present

The assumption of costs that are fixed for all possible amounts is very narrow. Therefore we included step costs, also called semi-fixed costs, that remain fixed only over a range of units produced and increase to a higher level once a critical level of output has been reached. In figure 5.2 it can be seen that the steps are diminishing in the average cost perspective but not in the marginal cost perspective. In our example, a second truck might be needed as the capacity of the first one is fully utilized.

5.1.1.2 Variable Cost

Variable costs on the other hand change, in proportion to the quantity that is produced. In figure 5.3 we can see that linear variable costs on its own do produce constant average and marginal cost. This can be interpolated easily with a single price, both with incremental and total quantity discounts. In our example this could be the cost for the fuel used, which is linear in the weight of the truck.

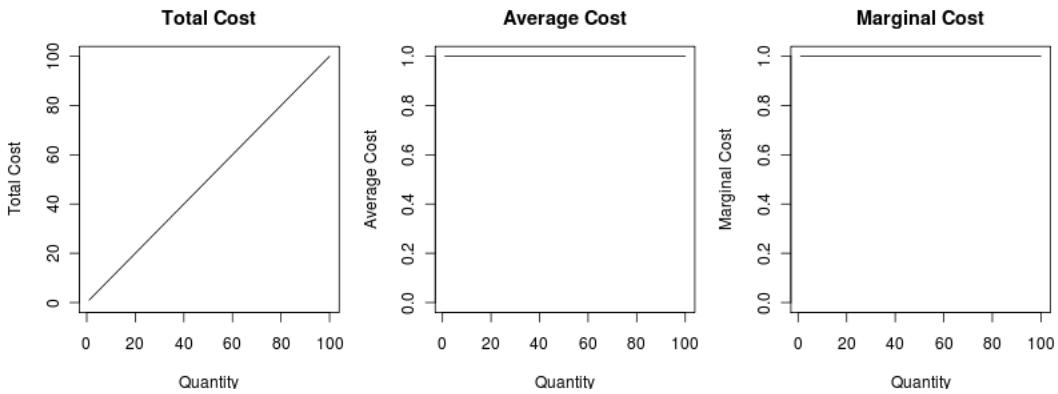


FIGURE 5.3: Total, average and marginal costs dependent on the number of units purchased if only variable costs of 1 $ are present

In economics literature Pindyck and Rubinfeld (2005) average cost is generally thought to be U-shaped, because for low quantities the economies of scale dominate until an optimal point from where the diseconomies of scale become dominant.

In our example variable costs could relate to the costs of the labor needed for handling the goods transported. For amounts that can be handled by regular staff in time it is decreasing, but when the capacity is fully utilized the cost is increasing because of the overtime work. Variable costs therefore have a point from whereon they grow faster than linear with the amount purchased. Together with fixed costs they combine to a U-shaped average cost function as seen in figure 5.4. If there are not only superlinear variable cost present but also stepwise fixed costs the average total cost gets "W"-shaped as seen in figure 5.5.

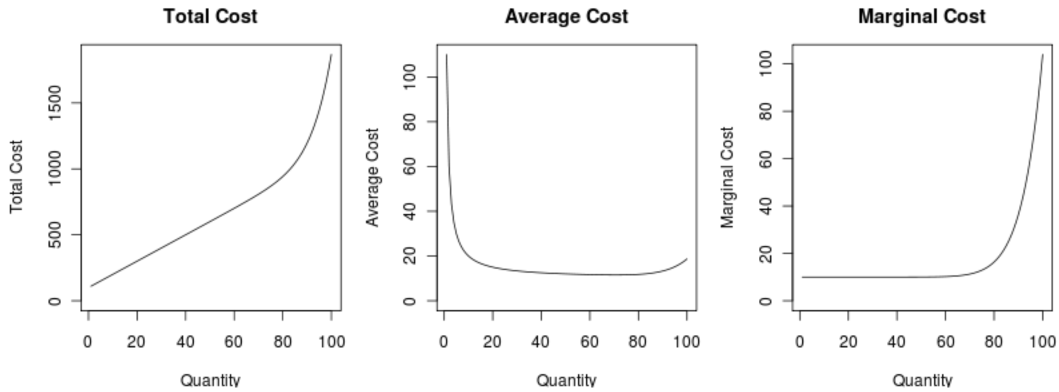


FIGURE 5.4: Total, average and marginal costs dependent on the number of units purchased if fixed costs of 1000 $ and superlinear variable costs are present

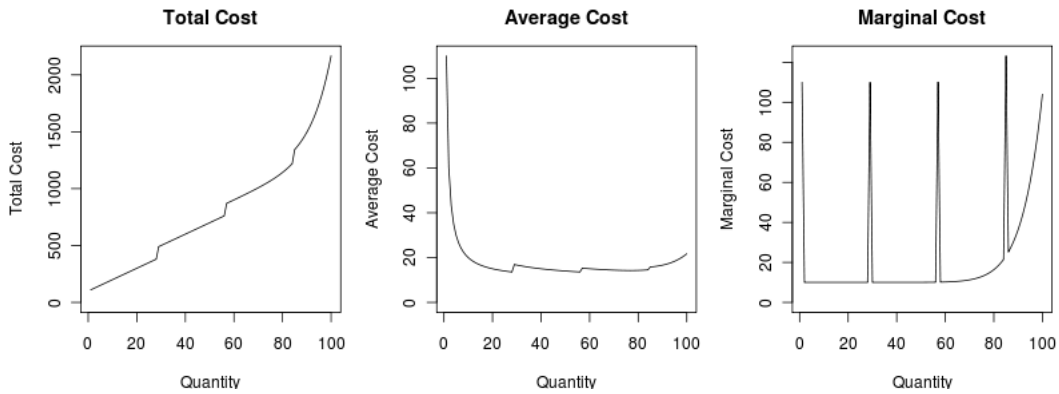


FIGURE 5.5: Total, average and marginal costs dependent on the number of units purchased if stepwise fixed costs of 200 $ and superlinear variable costs are present

This makes the second derivative non-continuous and the approximation with volume discounts hard. A possible way to avoid this, if the jumps in the cost function are known, is to use lump sum markups.

5.1.2 Parametrizable Cost Function

In order to generate simulation instances we need to define a function that can on the one hand represent all the characteristics we have seen in the previous section, and on the other hand it must be possible to parameterize and repeatably randomize it.

5.1.2.1 Parametrizable Single Item Cost Function

We propose the following function to generate our instances with all available parameters are summarized in table 5.1. Similar cost functions have been used to estimate cost parameters of companies with multiple products or outputs (Baumol, 1987; Evans and Heckman, 1984; Stewart, 2009) in economics.

$$c_{i,s}(x_{i,s}) = B_{i,s}\lceil x_{i,s}/z_{i,s}\rceil + \alpha_{i,s}x_{i,s} + \beta_{i,s}(x_{i,s}/\gamma_{i,s})^{\rho_{i,s}}$$

The first phenomenon we've seen are, possibly stepwise, fixed costs, which can be distributed over an increasing number of goods for increasing demand, and lead to decreasing average costs. This is modeled by the first element and therefore parameter of the cost function $B_{i,s}$. This describes the item specific stepwise fixed cost for item i,where z models the capacity bound, after which an additional $B_{i,s}$ of costs arise. Note, that with stepwise fixed costs present, the cost function is not continuous any more.

Cost function Parameters	
Parameter	Description
$B_{i,s}$	Item stepwise fixed cost of supplier s for item i
$z_{i,s}$	Capacity of production line for item i from supplier s
$\alpha_{i,s}$	Linear variable cost of supplier s for item i
$\beta_{i,s}$	Slope of the nonlinear variable cost of supplier s for item i
$\gamma_{i,s}$	Delay of the nonlinear variable cost of supplier s for item i
$\rho_{i,s}$	Exponent of the nonlinear variable cost of supplier s for item i

TABLE 5.1: Single item cost function parameters

The second phenomenon is the, possibly non-linear, variable cost. We have included $\alpha_{i,s}$ describing the linear part of the variable costs, though this parameter is not by itself necessary in order to represent all the characteristics, because together with the fixed costs they allow different suppliers to be the cheapest at different demand levels.

$\beta_{i,s}$ describes the slope of the nonlinear part of the variable costs for product i and is also not indispensable but makes balancing between the fixed and variable costs easier. $\rho_{i,s}$ is the exponent of the nonlinear element in the cost function, representing diseconomies of scale. The shape of the cost function is very sensitive to this parameter and for values ≥ 1.3 it becomes dominant even for moderate demand levels.

In practice we've seen the cost curves to be relatively flat and diseconomies of scale to arise only for high demand levels but there quite sharp. In our cost function this is made possible by $\gamma_{i,s}$ which delays the effect of the nonlinear part of the cost function. Problems instances with only a small flat proportion of near minimal average costs are not only unrealistic but also atypically easy to solve because large parts of the cost curves do not need to be considered.

5.1.2.2 Parametrizable Multi Item Cost Function

One of our key contributions is the possibility to include economies of scope and therefore we also wanted to incorporate them into our experiments. The cost function thereby must become a "number of different good types"-dimensional function, giving a cost for every possible combination of amounts of the goods. There are two main sources of economies of scope, shared fixed costs, and synergies(similarities) between items. Both can be added easily to the single item cost function:

$$c_s(x_1, ..., x_I) =$$
$$A_s\lceil \sum_{i\in\mathcal{I}} x_i/W_i\rceil + \sum_{i\in\mathcal{I}} B_{i,s}\lceil x_i/z_i\rceil + \sum_{i\in\mathcal{I}} \beta_{i,s}(x_i/\gamma_{i,s})^{\rho} + \sum_{i\in\mathcal{I}} \sqrt{x_i \sum_{i\neq j} \chi_{i,j,s} x_j}$$

A_s describes the fixed overhead cost of supplier s and $\chi_{i,j,s}$ is used to model synergies between two products i and j. The term representing the synergies is square rooted to make it a linear part of the cost function and can model both economies and diseconomies of scope with negative and positives values of $\chi_{i,j,s}$. Table 5.2 gives a summary of all available parameters in the multi item cost function.

Each of these parameters has to be chosen carefully in order not to degenerate the resulting cost function.

Cost function Parameters	
Parameter	Description
$A_{i,s}$	Overhead cost of supplier s
$B_{i,s}$	Item stepwise fixed cost of supplier s for item i
$z_{i,s}$	Capacity of production line for item i from supplier s
$\alpha_{i,s}$	Linear variable cost of supplier s for item i
$\beta_{i,s}$	Factor of the nonlinear variable cost of supplier s for item i
$\gamma_{i,s}$	Delay of the nonlinear variable cost of supplier s for item i
$\rho_{i,s}$	Exponent of the nonlinear variable cost of supplier s for item i
$\chi_{i,j,s}$	Cross item cost synergy of supplier s for item i and j

TABLE 5.2: Multi item cost function parameters

5.1.2.3 Proposed Parametrization

Even the best parameterizable cost function will not generate meaningful test instances if the parameters are not chosen carefully. For the related family of knapsack problems it has been shown by Pisinger (2005) that the correlation between the weight and the profit of an item has a big impact on the hardness of the problem.

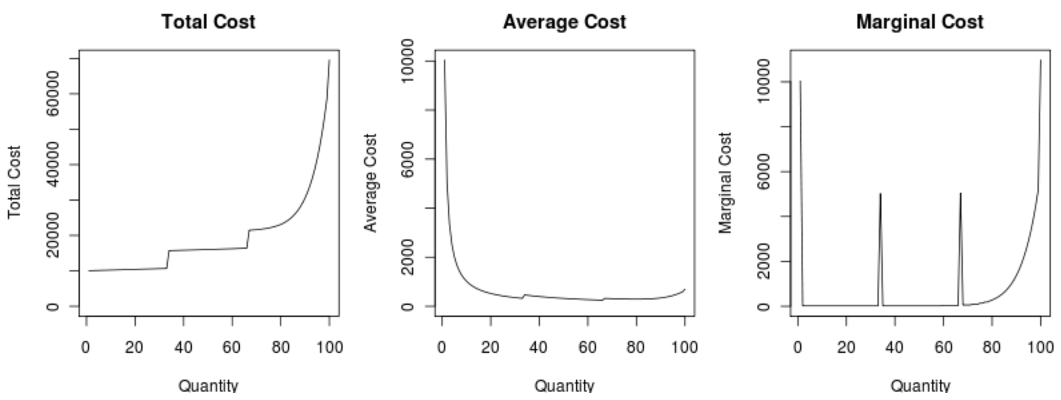


FIGURE 5.6: Cost function base case (not randomized)

The equivalence of this correlation in our setting would be the correlation between the amount purchased and the costs for it. This maps to our observation, that instances where there is a large relatively flat area in the average costs curves are harder. In figure 5.6 you can see the shape of the cost curve we used as a base case for our simulations. It includes all the properties we've discussed in section 5.1.1 and features a large relatively flat area in the average cost. In table 5.3 you can see which parameters were used to generate this cost curve.

Variable	Default value
$B_{i,s}$	5000.0
z_i	33%
$\alpha_{i,s}$	20.0
$\beta_{i,s}$	20.0
$\gamma_{i,s}$	60.0
ρ	15

TABLE 5.3: Parameters of the cost funtion in the proposed base case

On the other extreme instances that incorporate constant average costs are trivial too, because then there are no payment modifiers needed to interpolate the cost curve and the formulation becomes a linear program without integrality constraints.

5.1.2.4 Randomization

In order to generate random problem instances out of this parameterizable cost function, the parameters need to be disturbed randomly. Here we need to be careful to balance between generating instances with sufficient variation and keeping the fundamental characteristics of the cost function.

The parameters for a randomized instance usually are drawn from a probability distribution. By the central limit theorem, the sum of a number of random variables with finite means and variances approaches a normal distribution as the number of variables increases.

The prices for a product are affected by a large number of factors that clearly have finite means and variances. Therefore we chose to draw all parameters for the cost function of a specific instance from a normal distribution with the mean being the base parameter as in table 5.3. The standard deviation was set to values that gave a wide spectrum of shapes, but in average still yield a "W"-shaped cost function and are summarized in table 5.4.

For every parameter there was a plausibility check added, which prevented inconsistent values and abolished the respecting instances. In figure 5.7 you can see three examples of randomly parameterized cost functions.

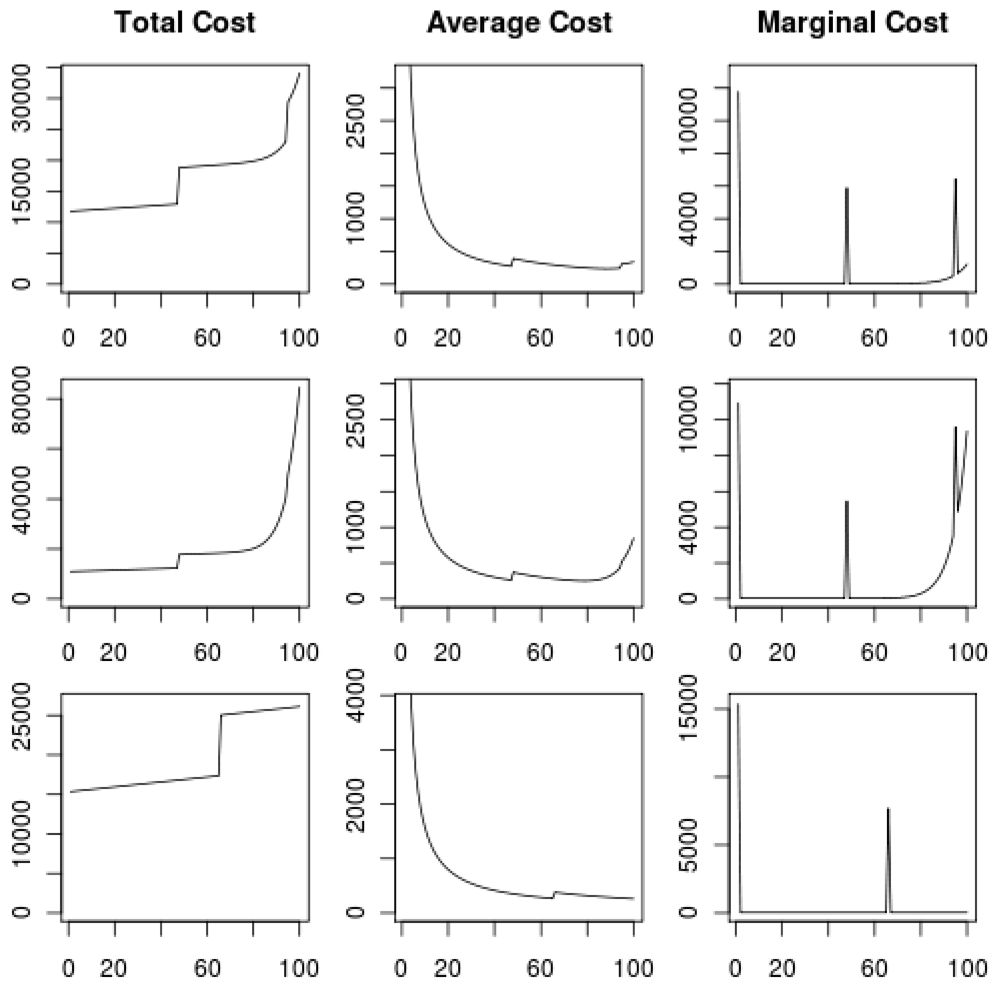


FIGURE 5.7: Random instaces of the cost function for seeds 2,3 and 9

The similarity of the prices for equal amounts purchased from different suppliers is another important factor that needs to be considered. For real world products the prices from different suppliers are clearly correlated through similar production standards and the market pressure nivellates disparities even further.

In Leyton-Brown et al. (2002) the authors have shown that the number of dominated bids is the key characteristic for the computational hardness of a

Variable	μ	σ_{item}	$\sigma_{supplier}$
$B_{i,s}$	5000.0	2500.0	500.0
z_i	33%	25%	15%
$\alpha_{i,s}$	20.0	10.0	5.0
$\beta_{i,s}$	20.0	10.0	5.0
$\gamma_{i,s}$	60.0	20.0	10.0
ρ	15	2.0	0.5

TABLE 5.4: Parameters of the cost funtion in the base (singe item) case

combinatorial auction instance, and many distributions proposed in the literature fail to generate a sufficient number of them. This leads to a misjudgment of the problem size and interferes predictions of the running time. Analogously to the dominated bids in Leyton-Brown et al. (2002) cost curves that are dominated by other cost curves contain many payment modifiers that do not need to be considered.

As seen in figure 5.7 the shapes of the cost functions for different seeds are quite varying by design and a lot of them would be dominated if assigned to the same item. Therefore we choose a two step process where we have drawn the respective parameters for a cost curve per item. And in a second step added additional variation for each supplier taking the realization of the initial random variable as the mean of a new supplier-specific random variable. The respective standard deviation values are shown in table 5.4 and three resulting supplier specific cost curves are shown in figure 5.8.

The resulting cost curves together with the amount demanded already define the optimal, meaning cost minimizing, solution of the problem. Translated into a combinatorial auction instance, every possible combination of item amounts requested would translate to a bid. The same would be possible in $\mathcal{L}_{\mathcal{ESS}}$ by specifying a payment modifier for every possible combination of item amounts requested. This extreme approach clearly is a contradiction to our goal of reducing the representation complexity for the bidders. Therefore the next step will be to derive rules how simulated bidders could interpolate their cost function with the possibilities $\mathcal{L}_{\mathcal{ESS}}$ offers.

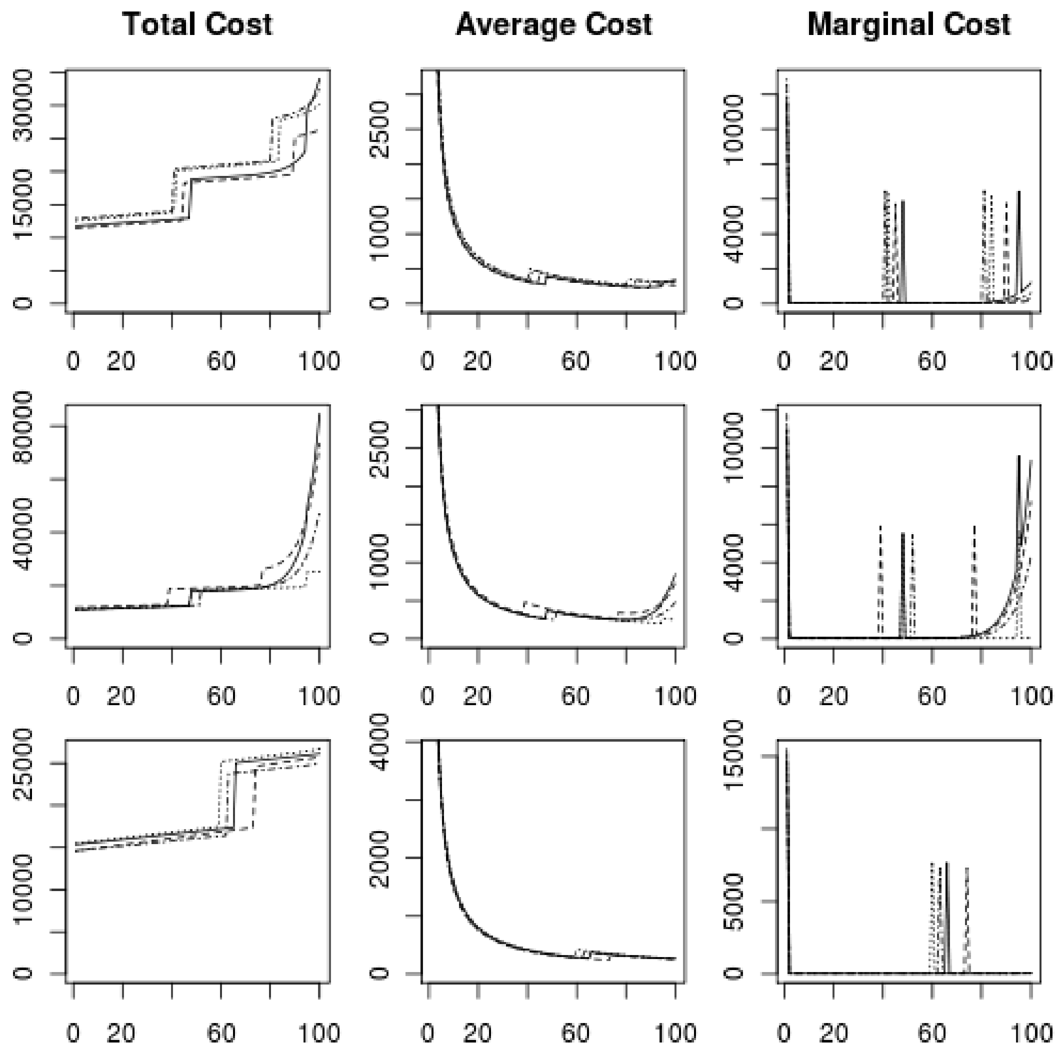


FIGURE 5.8: Random instances of the cost function for seeds 2,3 and 9 and different supplier realizations in dotted lines

5.2 Bid Generation

The bid generation describes the process of approximating the cost function with $\mathcal{L}_{\mathcal{ESS}}$, similar to what has been described in ??. The resulting function will be called bid interpolation function from here on.

Note, that in a real procurement auction suppliers might not want to reveal their true cost functions if it is not induced by the auction mechanism. For combinatorial auctions this can be reached by a so called generalized vickey auction for example as described in 2.2.2.1. Note further, that if the approximation error gets large and the bid language does not allow to describe the underlying cost function arbitrarily close, this might have an impact on the bidding strategy and bidders might have an incentive to speculate.

The applicability or transformability of this mechanism to our problem would be an interesting further work and is not examined in this thesis. A game theoretical analysis only makes sense if we already have a working bidding language and supplier selection process, and therefore we assume well behaving bidders to evaluate the possible performance of our approach.

Accordingly we assume a direct revelation mechanism, where bidders try to reveal bids reflecting the underlying total cost function truthfully and as close as possible without underbidding their costs. We analyze two treatments: Either the number of discounts per item and the corresponding quantity intervals are fixed, or the suppliers can supply an arbitrary number of intervals determined by a predefined approximation error ϵ_s^{max}.

5.2.1 Fixed Interval Bidding

Often in practical settings the boundaries of the price intervals are predefined by the auction, as this only requires the bidders to submit a single number per interval indicating the price valid in that interval. This also can be easily implemented in a web form for example.

Predefining the intervals naturally limits the expressiveness of the bidding language. In figure 5.9 you can see the bid interpolation functions of our standard cost function without stepwise fixed costs using five equidistant intervals for incremental and total quantity bids.

For total quantity bids, where the resulting interpolation function is piecewise linear in the intervals, one can see, that the bid interpolation function overlaps with the cost function at the point of maximal average costs. As total quantity bids describe a piecewise constant function in average costs this point determines the minimal price at which the interpolation function never lies below the cost function. As for total quantity discounts the price is valid for the whole amount purchased, the resulting interpolation function intersects the origin of the coordinate system in total costs, resulting in the typical saw

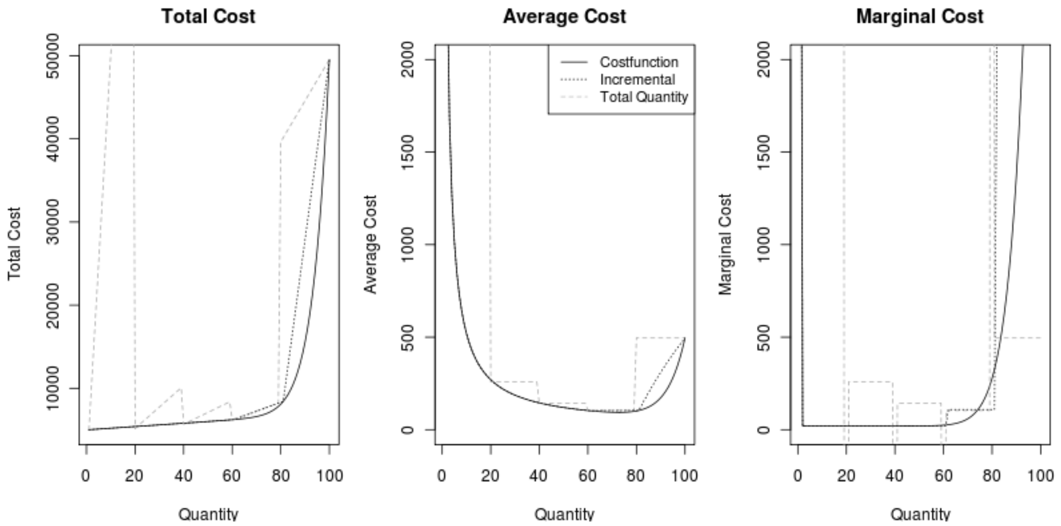


FIGURE 5.9: Comparison of fixed interval bidding with 5 intervals

tooth shape. The approximation error increases with an increasing gradient of the average cost curve.

With incremental bids on the other hand the price is valid only for the incremental units and the total cost curve of the bid interpolation function dos not incorporate any jumps. The resulting bid interpolation function is constant on marginal costs and the approximation error depends on the gradient of the marginal cost curve. As discussed in **??** the approximation error of incremental bids is never greater than the approximation error of total quantity bids for convex cost functions.

If the cost function itself does incorporate jumps, as generated by stepwise fixed costs for example, this does no longer hold for all cases. The jumps in total cost effects that the bid interpolation function for incremental bids does no longer overlap with the cost function at the interval boundaries. Therefore the approximation error can get arbitrarily big if a jump occurs close to an interval boundary. Total quantity bids do not suffer from this effect but the interpolation error also tends to increase with the presence of stepwise fixed costs on a smaller scale.

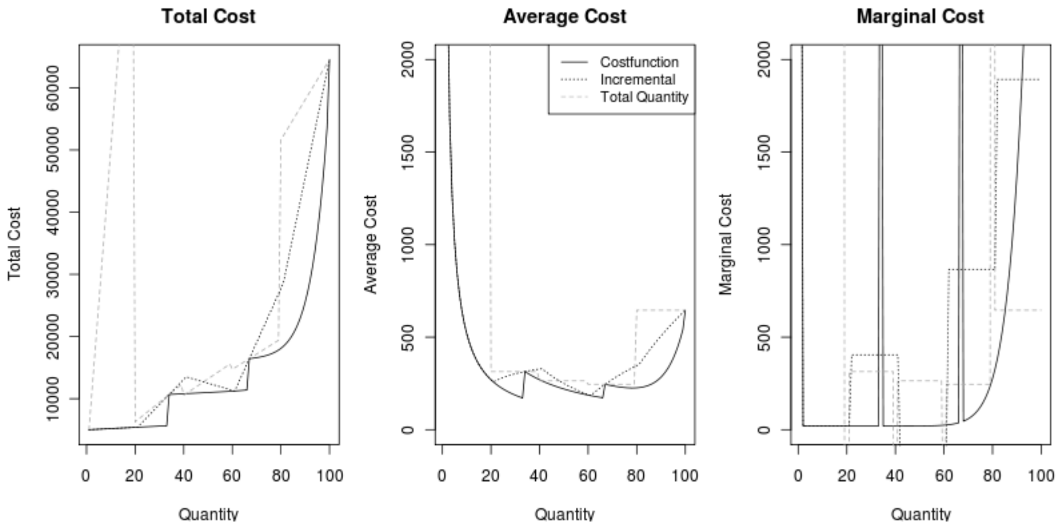


FIGURE 5.10: Comparison of fixed interval bidding with 5 intervals and stepwise fixed costs

5.2.2 Fixed Approximation Error Bidding

The intention of $\mathcal{L}_{ESS}$ was to make it handle able for bidders to express their cost function by reducing the number of priced needed. Therefore we investigated the question how many priced are needed to express a cost function with an bounded approximation error and which impact has this on computation time and total spend.

To answer this question we implemented a simulated bidding agent that interpolates a given cost function with a given approximation error using as little tiers as possible. It features greedy approach that generates maximally wide intervals regarding the tolerable approximation error relative to the current total costs, starting at a single item. In 5.11 you can see the resulting bid approximation functions.

The large approximation errors for incremental bids with fixed intervals are caused by disadvantageous interval boundaries. If bidders can define the price breaks freely, as in $\mathcal{L}_{ESS}$, a jump in the cost function can be interpolated with an incremental markup followed by an incremental discount. The jump in total cost translates to a spike in marginal costs, which can be interpolated with an incremental interval of length 1.

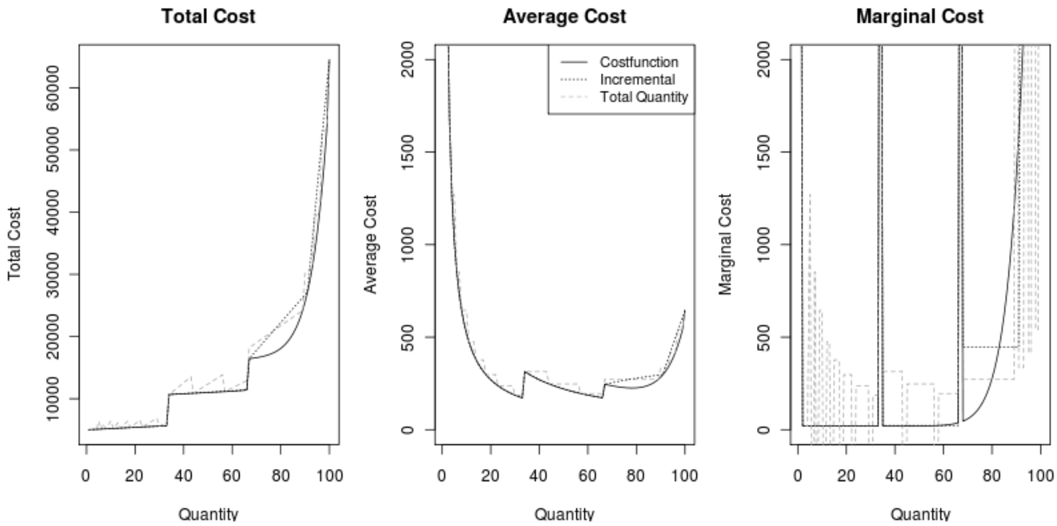


FIGURE 5.11: Comparison of fixed approximation error bidding with a maximum error of 25%

For total quantity bidding the advantage of having freely settable price ranges is not as outstanding but it allows the bidders to express their costs functions as close as desired. For both discount types having freely settable price ranges allows to interpolate any given cost function exactly by defining a price individually for every possible quantity.

5.2.2.1 Fixed Approximation Error Bidding with Lump Sum Modifiers

As pointed out in 5.1.1 jumps in the cost function cannot only be expressed as a pair of an incremental discount and markup in $\mathcal{L}_{ESS}$, but also by so called lump sum markups. If lump sum markups are used to express jumps the net cost function is convex again and incremental bidding reduces to piecewise linear interpolation as described in **??**.

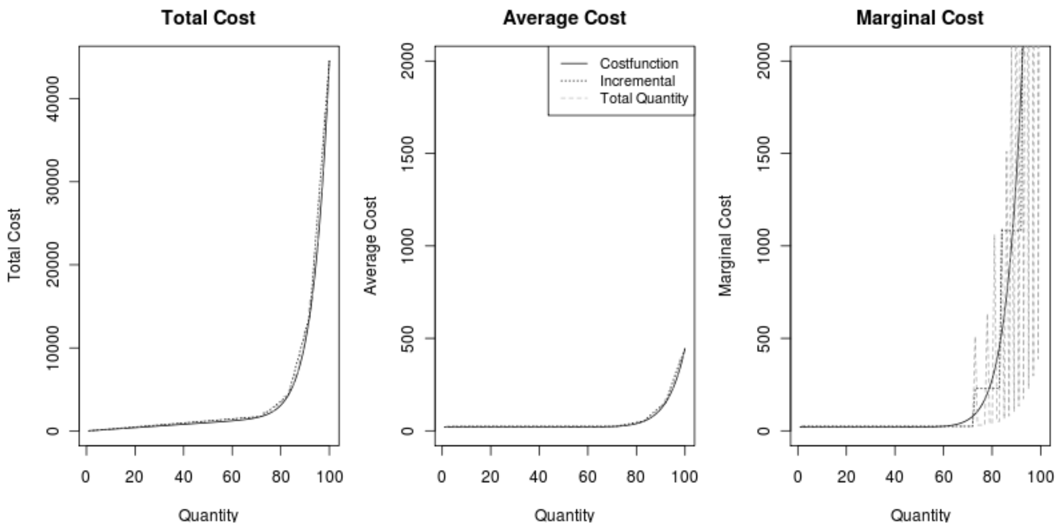


FIGURE 5.12: Comparison of fixed approximation error bidding with a maximum error of 25% without fixed costs

In figure 5.12 you can see the approximation of our standard cost function where the fixed costs set to zero. This reflects the situation a bidder faces if the fixed costs are already covered using lump sum markups. With no fixed costs present the average costs are strictly increasing. Therefore there are for incremental bidding only markups needed to describe the nonlinearity at high amounts. For total quantity bidding the situation even improves more as the apparent absence of fixed costs makes the total cost function go through the origin of the coordinate system.

Chapter 6

Experimental Results

In this chapter we will summarize the results of our simulation experiments.

6.1 Cost of Flexibility

Goossens et al. (2007) provided a tailor-made formulation for total quantity discount bids, as well as computational results on randomly generated test instances. In contrast, $\mathcal{L}_{ESS}$ provides bidders with more flexibility and allows for other types of volume discounts and various spend conditions. Typically, more flexibility and expressiveness comes at the cost of computational complexity. It is interesting to see, if the expressiveness of $\mathcal{L}_{ESS}$ comes at a large computational cost.

Therefore, in a first set of experiments, we used the synthetic bids, which were kindly provided by Goossens et al. (2007) for a comparison. These experiments are limited to bids with total quantity discounts only.

In their instances with 40 items, Goossens et al. (2007) generated an upperbound-increase from one interval to the next, which was a random number between 10,000 and 50,000, while for instances with 100 items, the upperbound increase was a random number between 10,000 and 100,000. The results of the formulation in Goossens et al. (2007) with a branch-and-cut approach and with the $\mathcal{SQS}$ formulation can be found in Tables 6.1 and 6.2. We focus on the branch-and-cut results, since those provided the best results for larger instances. Note that we have used CPLEX version 12.1, while Goossens

Instances	$\mathcal{SQS}$ b&c comp. time	$\mathcal{SQS}$ b&c #nodes	Goossens b&c comp. time	Goossens b&c #nodes
S 10 40 3	0.08	9.3	0.09	0.3
S 10 40 5	0.17	16.4	0.15	10.9
S 10 100 3	0.13	5	0.12	0.2
S 10 100 5	1.13	31.1	0.55	3.2
S 20 40 3	0.16	10.9	0.12	0.5
S 20 40 5	0.87	36.7	0.50	4.7
S 20 100 3	0.49	17.1	0.34	2.1
S 20 100 5	2.40	32.1	1.17	2.1
S 50 40 3	0.79	34.6	0.51	2.7
S 50 40 5	5.87	102.9	2.99	16.5
S 50 100 3	2.45	28.0	1.45	2.1
S 50 100 5	21.67	94.8	10.45	14.7
R 10 40 3	0.07	6.7	0.09	2.1
R 10 40 5	1.10	85.5	0.59	30.5
R 10 100 3	0.22	7.2	0.14	2.6
R 10 100 5	3.814	70.8	1.50	31.7
R 20 40 3	0.29	18.7	0.29	9.5
R 20 40 5	3.76	138.5	1.81	68.5
R 20 100 3	1.41	25.9	0.83	8.1
R 20 100 5	29.75	274.9	6.81	70.1
R 50 40 3	2.01	71.3	1.67	81.6
R 50 40 5	26.90	346.5	14.18	140.9
R 50 100 3	12.75	91.9	8.84	43.8
R 50 100 5	451.84	996.0	61.71	216.8

TABLE 6.1: Comparison of $\mathcal{SQS}$ against the results by Goossens et al. (2007) (base case)

et al. (2007) used version 8.1. Also, they used a Pentium IV 2 GHz computer with 512 Mb RAM. Nevertheless, the comparison helps understand possible penalties for a more expressive bidding language.

	$\mathcal{SQS}$ b&c		Goossens b&c	
Instances	comp. time	#nodes	comp. time	#nodes
S 10 40 3	0.05	6.2	0.14	0.4
S 10 40 5	0.16	12.6	0.49	23.2
S 10 100 3	0.09	2.1	0.29	6.8
S 10 100 5	1.05	26.4	2.48	66.5
S 20 40 3	0.12	8.9	0.45	5.2
S 20 40 5	0.87	31.9	4.42	148.7
S 20 100 3	0.44	11.1	2.08	43.4
S 20 100 5	2.08	28.8	11.47	164.2
S 50 40 3	0.61	19.4	8.90	261.9
S 50 40 5	6.02	100.6	147.56	2500.0
S 50 100 3	2.17	20.4	28.42	523.7
S 50 100 5	21.02	84.7	271.80	3006.0
R 10 40 3	0.07	6.3	0.10	0.1
R 10 40 5	0.71	40.7	0.67	18.5
R 10 100 3	0.19	7.5	0.17	0.2
R 10 100 5	3.19	51.5	2.49	42.6
R 20 40 3	0.24	12.8	0.60	15.9
R 20 40 5	2.64	80.7	3.55	89.8
R 20 100 3	1.07	21	1.85	17.6
R 20 100 5	31.38	322	25.09	434.8
R 50 40 3	1.65	49.1	35.16	1196.8
R 50 40 5	21.82	298.7	169.82	1511.1
R 50 100 3	10.78	68.6	274.82	6035.4
R 50 100 5	400.68	1064.3	2036.07	17577.4

Table 6.2: Comparison of $\mathcal{SQS}$ against the results in Goossens et al. (2007) (more for less)

Interestingly, much larger instances could be solved to optimality with this set of bids in seconds, and the differences in runtime between the results reported by Goossens et al. (2007) and the results of $\mathcal{SQS}$ were small. The results for the base case in Table 6.1 where a bit slower, while the results for the more-for-less scenario in Table 6.2 were actually faster. The more-for-less scenario aimed for optimal solutions with free disposal of additional quantity, while the base case did not.

Obviously, the structure of the bids has a significant impact on the runtime. The bids generated based on our multi-product cost function were considerably harder to solve. We assume that smooth cost functions generate a lot of solutions with similar objective function value, a type of symmetry problem. The structured instances in Goossens et al. (2007) include economies of scale, where intervals with more quantity have lower prices than intervals with less quantity, whereas the cost functions in our paper also include diseconomies of scale. Also the demand can make a difference. If the demand is increased, runtimes can increase, because there are more possible quantities to purchase.

In summary, the predictive quality of such runtime experiments depends on the structure of the bids and the respective scale economies in a market. Overall, our analysis reveals that surprisingly large instances can be solved to optimality with these types of bids.

6.2 Value Model Cost of Production

In our main experiments we used the value model $\mathcal{COP}$ to generate instances as described in chapter 5.1. We used these instances to evaluate $\mathcal{L}_{ess}$ and $\mathcal{SQS}$ with respect to the computational and communicational burden as well as the possible savings in spend relative to a split award auction. Computation time is central if $\mathcal{SQS}$ is to be solved in an interactive analysis of different award scenarios during the scenario analysis. The bid evaluation should not take more than a minute, as this allows exploring different side constraints in an interactive manner. In this section, we have therefore set a time limit of 60 seconds and reported the current best solution for not proven optimal or suboptimal instances.

6.2.1 Single Item Instances

In the first set of experiments we generated instances as outlined in section 5.1.2.4 with a single item in order to evaluate the impact of scale economies in isolation. In table 6.3 the parameters of the runs are summarized.

Variable	Values
Number of suppliers	2, 8, 32, 64
Number of items	1
Number of price intervals	4, 8
Acceptable approximation error	1%, 25%
Seed	1-30

Table 6.3: Parameters of the single item runs

6.2.1.1 DescriptionLength

We have discussed in section 3.1.1 that apart from the expressiveness of a bid language, the description length matters. 6.1 shows how the number of intervals in the $\mathcal{SQS}$ formulation increases with a decreasing maximal absolute approximation error ϵ_s^{max}. The required number of discount intervals is much higher for total quantity bids.

The large number of intervals needed to describe a cost function with a low approximation error can make bidding impractical. Note that in an incentive

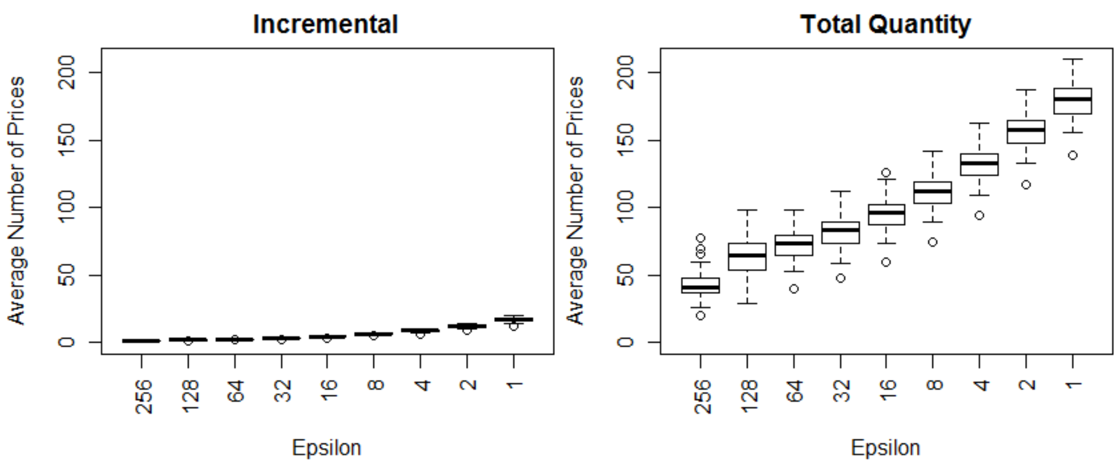


FIGURE 6.1: Description length comparison of incremental vs. total quantity bids in $\mathcal{L}_{ESS}$ for 32 supplier instances of the base case

compatible mechanism such as the Vickrey-Clarke-Groves mechanism, bidding truthfully might not be a dominant strategy, unless the bidding language allows describing the underlying cost function close enough with a realistic number of bids.

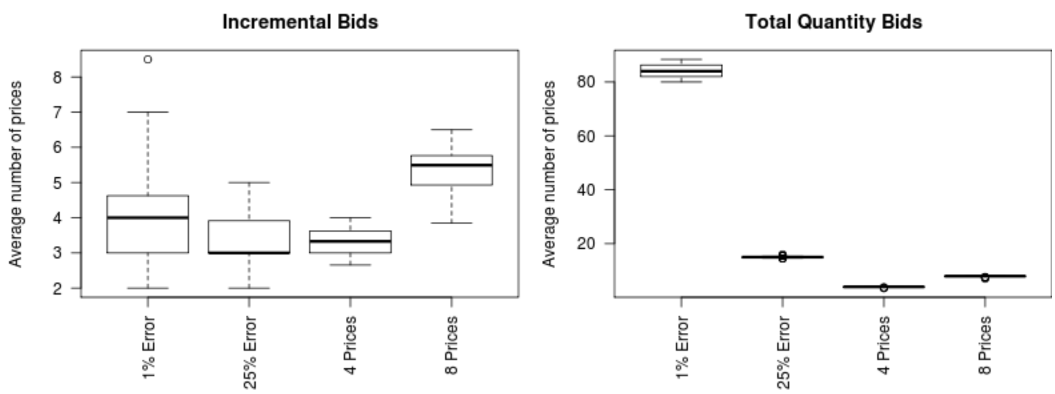


FIGURE 6.2: Comparison of number of price intervals for different incremental and total quantity bidding policies for 30 single item CoP instances

In figure 6.2 we have plotted the resulting number of price intervals for the parameters used in our simulations. The boxes show the number of intervals that were generated as described in section 5.2.

For incremental bidding the necessary number of price intervals was always below ten even with an approximation error of 1 %. For total quantity bidding on the other hand, even an approximation error of 25 % demanded around 20 price intervals. This shows, the burden for the bidders to communicate their cost function nearly exactly is only manageable using incremental bids in these settings.

6.2.1.2 Computational Burden

In figures 6.3, 6.4 and 6.5 one can see a comparison of the computation time needed to solve $\mathcal{SQS}$ for instances with 2, 8 and 64 suppliers and for different bidding policies.

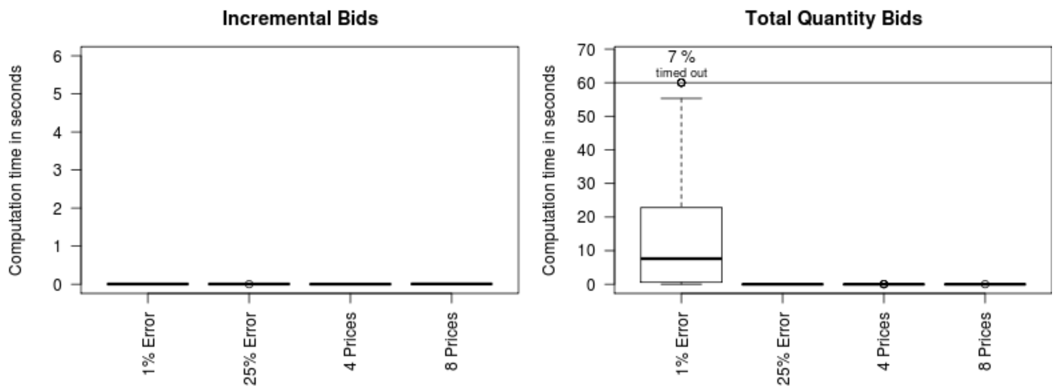


FIGURE 6.3: Comparison of computation time for different incremental and total quantity bidding policies for 30 single item CoP instances with 2 suppliers

The number of intervals greatly influences the expected computation time for $\mathcal{SQS}$, which can be clearly seen in our results. All instances except the ones with total quantity bids and an approximation error limiting bidding policy could be solved to proven optimality within 60 seconds. With an allowed approximation error of 25 % only a few instances timed out, whereas with an allowed approximation error of 1% and at least 8 suppliers more than half of the instances timed out.

The instances with a fixed number of price intervals could all be solved to optimality in less than 20 seconds. At four price levels total quantity instances

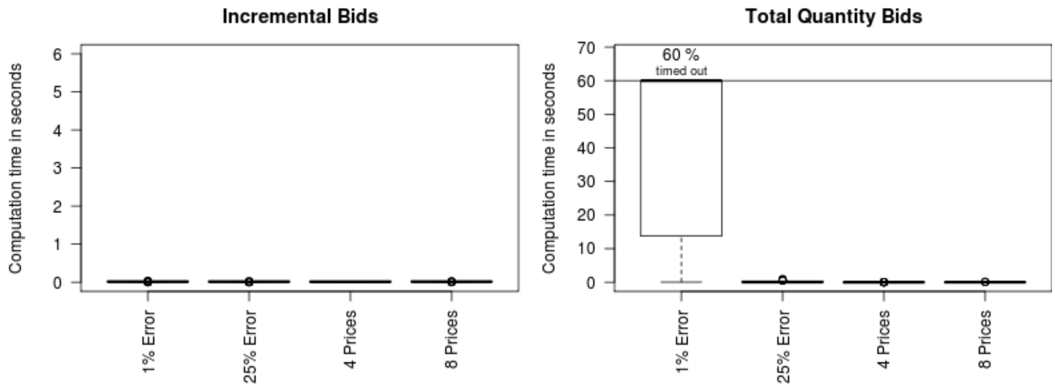


Figure 6.4: Comparison of computation time for different incremental and total quantity bidding policies for 30 single item CoP instances with 8 suppliers

took less time on average but at 8 price intervals some total quantity instance took more than 10 times as long as the respective incremental instances.

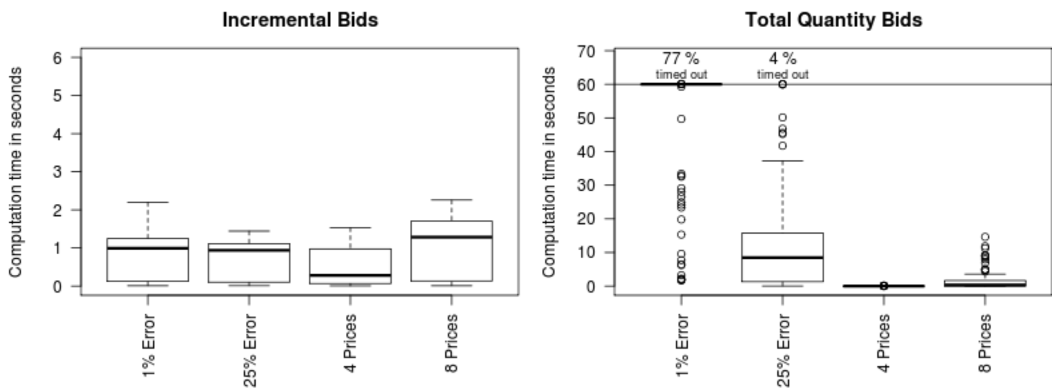


Figure 6.5: Comparison of computation time for different incremental and total quantity bidding policies for 30 single item CoP instances with 64 suppliers

Overall one can say that the usage of incremental bids, of predefining a fixed number of price intervals, makes scenario navigation possible for auctions where big quantities of only one good are auctioned.

6.2.1.3 Spend Comparison

The optimality that is mentioned in the previous section relates to the MIP formulation for $\mathcal{SQS}$, and even an 100 % optimal solution can only be as good as the bids that were submitted. For real world uses the spend, the total amount that the auctioneer has to pay in order to get the demanded quantities, is the most important benchmark. In real world cases the most common approach is to use an split award auction. Therefore we used simulated suppliers submitting two price quotes for each item at 30% and 70% of the volume in a split award auction as the point of reference.

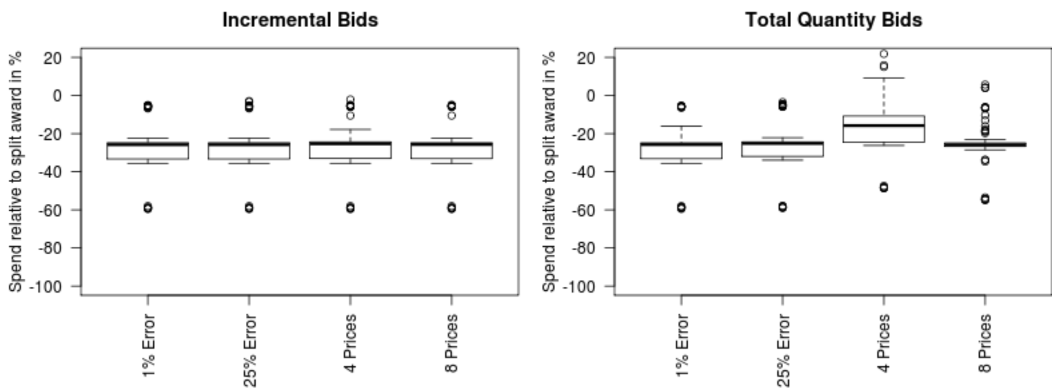


FIGURE 6.6: Comparison of spend relative to an split award auction for different incremental and total quantity bidding policies for 30 single item CoP instances with 2 suppliers

In figures 6.6 and 6.7 the spend that resulted from using $\mathcal{L}_{ess}$ and $\mathcal{SQS}$ with the different bidding policies against a split award auction.

It is very outstanding, that the spend of all incremental auctions was nearly equal. The average spend of the instances with an allowed approximation error 25 % was 0.027 % higher than the one resulting from an allowed approximation error of 1%. Even if only 4 fixes intervals were used the spend was only 0.45 percent higher. Total quantity bids showed similar results for the bidding policies with an approximation error bound. For fixed price intervals the results have been not so good. In the worst case and only 4 intervals available the spend was even higher than with a simple split award auction. This did only happen for a low number of suppliers as there are few alternative solutions, and the optimal solution can be blocked by a bad cost approximation.

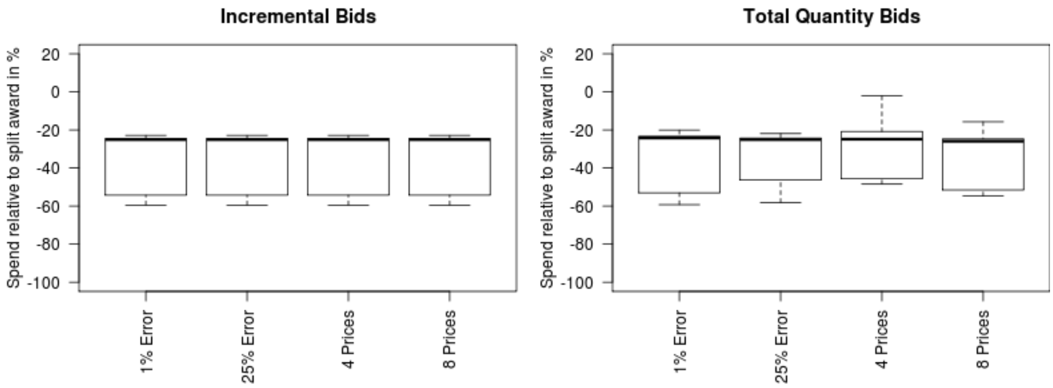


FIGURE 6.7: Comparison of spend relative to an split award auction for different incremental and total quantity bidding policies for 30 single item CoP instances with 64 suppliers

6.2.2 Single Item Instances with Lump Sum Discounts and Markups

In $\mathcal{L}_{ess}$ suppliers have the option to define so called lump sum discounts and markups, that are a one time payment. These lump sum discounts and markups can be interpreted as a direct way to communicate jumps in the cost function, as illustrated in section 5.2.2.1. In practice the bidders will only very seldom be able and or willing to communicate this information so this is more of theoretical value.

In figure 6.8 one can see, that the use of lump sum markups to indicate the jumps in the cost function makes computing $\mathcal{SQS}$ faster, and all instances could be solved in less than a second. Figure 6.9 shows the spend if bidders use lump sum markups. For all error bound bidding policies the spend was within less than 0.1 % of the optimum. With incremental bids 4 price intervals were also sufficient to get close to the optimal spend, and also total quantity bids profited a lot.

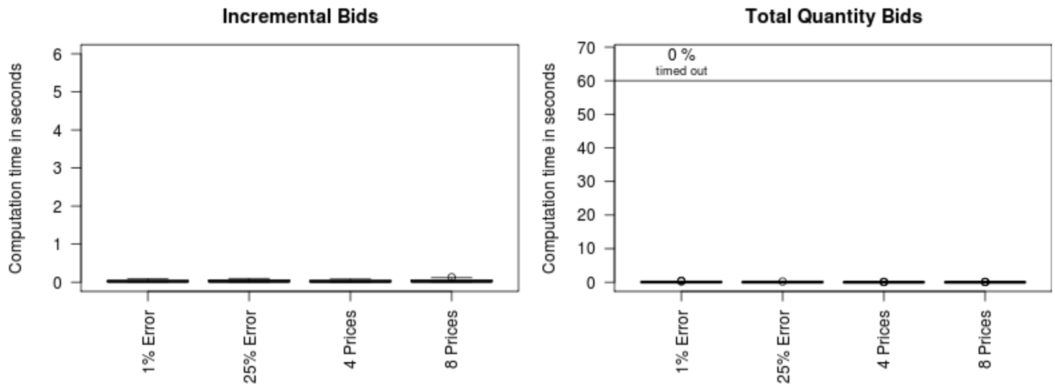


FIGURE 6.8: Comparison of computation time for different incremental and total quantity bidding policies for 30 single item CoP instances with 64 suppliers

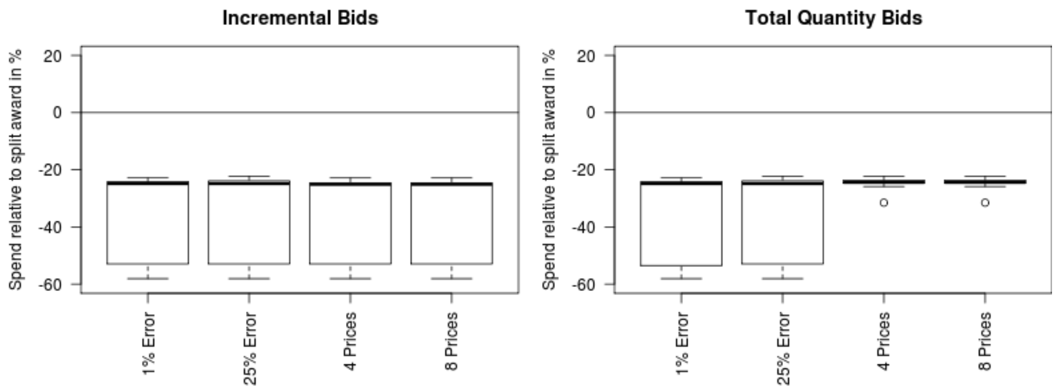


FIGURE 6.9: Comparison of spend relative to a split award auction for different incremental and total quantity bidding policies using lump sum discounts and markups for 30 single item CoP instances

6.2.3 Multi Item Instances - Realistic Problems

Finally, we will discuss the impact of the different discount policies on the total cost of the purchasing organization when auctioning multiple items at once. This is the most realistic but also most challenging scenario for the bidding language and supplier selection. In contrast to the single item cases where we focused on elaborating effects scale economies on bidding and supplier selection, we tried here to explore the applicability of $\mathcal{L}_{ess}$ and $\mathcal{SQS}$ for realistic multi item instances.

Again, we assume a direct revelation mechanism, and bidders submit their bid in a way that approximates their true costs as close as possible, either restricted by the number of intervals or a predefined approximation error. We simulated suppliers submitting two price quotes for each item at 30% and 70% of the volume in a split-award auction using the value model cost of production with parameters outlined as 5.1.2.4. The same suppliers also submit bids in an auction with only total quantity discount bids, and in an auction with only incremental volume discount bids. For the latter two auction types, we also distinguished settings with a fixed set of 5 intervals, or an approximation error ϵ of 1.0 %.

The results are summarized in Table 6.4. Each row summarizes the result of a particular volume discount auction relative to the cost of the split-award auction in percentage values after 5, 10, 20, 60, 120, and 500 seconds. All reported numbers are average numbers for 30 instances solved.

A spend greater than the one that could be realized by an split award auction can be a result of two factors, either the interpolation error is too big or the current integer solution is too far away from the optimum. Therefore we have then analyzed the proven optimality of these instances, and the MIP gap after 5, 10, 20, 60, 120 and 500 seconds can be found in 6.5. A value of -1.000 means that the best bound found so far is still ≤ 0, and there is no meaningful MIP gap reported by the solver.

As one can see the incremental instances with 5 intervals were not solved to optimality, while the instances with total quantity discounts could be solved to optimality even for larger instances with 30 suppliers and 10 items in 2 minutes. So the MIP gap of the incremental volume discount bids was always higher, when the number of intervals was limited to five. With a limit on the approximation error ϵ to 1%, the number of intervals grows up to more than 80 intervals per item in case of total quantity discount bids, but only up to around 30 intervals in case of volume discount bids. Interestingly, the instances with only incremental volume discount bids are still harder to solve and the MIP gap is typically larger. So, the structure of the problems with total quantity discounts allows for larger problems to be solved.

We will now take a look at the spend with a fixed number of 5 intervals. If the MIP gap is high, then also the spend with total quantity discount bids and incremental volume discount bids can be much worse than with simple split

		Spend in % of split awards with time limit					
Suppliers	Items	5s	10s	20s	60s	120s	500s
Total quantity 5 intervals							
10	10	101.13	100.94	100.94	100.02	100.94	100.94
10	30	101.22	101.22	101.22	101.58	101.22	101.22
10	50	102.28	102.20	102.19	102.45	102.19	102.19
30	10	101.44	101.44	101.44	101.44	101.44	101.44
30	30	102.75	102.59	102.57	103.80	102.56	102.56
30	50	102.79	102.69	102.48	102.74	102.27	102.27
Incremental 5 intervals							
10	10	93.65	93.47	93.49	93.27	93.45	93.46
10	30	97.31	95.50	94.67	94.84	94.48	94.40
10	50	103.54	100.72	96.70	95.87	95.62	95.46
30	10	96.76	94.86	94.37	92.63	94.28	94.21
30	30	109.45	109.09	105.02	97.81	95.97	95.62
30	50	115.53	113.37	107.40	103.07	97.98	95.85
Total quantity $\epsilon = 1.0$							
10	10	93.59	93.60	93.60	93.52	93.25	87.89
10	30	29311.83	95.37	95.37	95.77	95.36	95.17
10	50	31571.88	29585.34	96.65	97.44	96.64	96.64
30	10	117653.66	97.36	97.34	96.49	97.36	96.96
30	30	92869.85	92869.85	92869.85	102.60	102.30	102.29
30	50	82807.65	83589.52	83589.52	92582.93	103.08	103.08
Incremental $\epsilon = 1.0$							
10	10	87.72	87.73	86.35	86.09	86.27	86.15
10	30	374.71	132.40	94.47	90.68	88.08	87.69
10	50	450.39	335.86	123.02	95.08	89.94	89.19
30	10	930.91	184.22	96.96	89.46	87.88	87.29
30	30	1449.13	1277.17	1350.74	162.00	103.73	90.97
30	50	1675.84	1433.83	1333.95	696.10	150.84	94.08

Table 6.4: Comparison of spend relative to a split award auciton

award auctions. Therefore, we will mainly look at the column with 500s, where the MIP gap was low. Interestingly, in the setting with total quantity bids and 5 intervals, the spend achieved was higher than that of split award auctions, although the smaller instances could be solved to optimality. The reason for this is the bad approximation. In contrast, incremental volume discount bids

			Optimality with time limit					
Suppliers	Items	Prices	5s	10s	20s	60s	120s	500s
Total quantity, 5 intervals								
10	10	5	1.000	1.000	1.000	1.000	1.000	1.000
10	30	5	1.000	1.000	1.000	1.000	1.000	1.000
10	50	5	0.997	0.998	0.999	0.999	1.000	1.000
30	10	5	1.000	1.000	1.000	1.000	1.000	1.000
30	30	5	0.993	0.996	0.997	0.998	0.998	0.998
30	50	5	0.989	0.991	0.993	0.996	0.996	0.997
Incremental, 5 intervals								
10	10	5	0.937	0.944	0.947	0.963	0.958	0.965
10	30	5	0.890	0.909	0.919	0.927	0.923	0.926
10	50	5	0.844	0.867	0.904	0.918	0.916	0.920
30	10	5	0.887	0.909	0.916	0.925	0.921	0.927
30	30	5	0.438	0.765	0.802	0.877	0.881	0.889
30	50	5	0.319	0.512	0.765	0.804	0.840	0.866
Total quantity $\epsilon < 1.0$								
10	10	80.41	0.636	0.680	0.725	0.775	0.812	0.868
10	30	77.36	-1.000	0.509	0.558	0.638	0.681	0.758
10	50	76.65	-1.000	-1.000	0.496	0.582	0.615	0.699
30	10	80.41	-1.000	0.515	0.584	0.646	0.693	0.760
30	30	77.36	-1.000	-1.000	-1.000	0.493	0.529	0.621
30	50	76.92	-1.000	-1.000	-1.000	0.000	0.441	0.549
Incremental $\epsilon < 1.0$								
10	10	28.20	0.595	0.811	0.880	0.937	0.911	0.946
10	30	25.28	-1.000	0.219	0.375	0.412	0.538	0.597
10	50	24.58	-1.000	-1.000	0.328	0.438	0.499	0.576
30	10	28.20	-1.000	-1.000	-1.000	-1.000	0.096	0.168
30	30	25.22	-1.000	-1.000	-1.000	-1.000	-1.000	0.108
30	50	24.52	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000

TABLE 6.5: Proven optimality of the solutions in table 6.4

achieved a lower total cost compared to split award auctions for all problem sizes already after 120 seconds. This was the case, even though the problems could not be solved to optimality within 500 seconds.

In situations, where the approximation error ϵ was limited to 1%, total quantity discount bids led to cost savings after 500 seconds compared to split-award auctions. Larger instances with 30 suppliers and 30 items led to a large MIP gap even after 500 seconds and the results were worse than those of a split award auction. Also in this setting, the incremental volume discounts led to lower total cost compared to split-award auctions for all problem sizes, but also compared to total quantity discount bids after 500 seconds.

Chapter 7

Conclusion and Outlook

7.1 Conclusion

We have suggested a bidding language for markets with economies of scale and scope and a respective MIP to solve the resulting supplier quantity selection (SQS) problem. We try to easy the problematic bidding complexity, induced by the huge number of possible bid combinations, for the bidders in a combintorial auction. This problem we've shown, by numerical simulations, to be a limiting factor of the performance of combinatorial auctions in general. This gets even worse in procurement settings.

The proposed bidding language is considerably more expressive than what has been discussed in the literature so far and includes incremental volume discounts, total quantity discounts, lump sum discounts, and a variety of spend conditions defined on spend and quantity of selected items. While both, incremental volume discount bids and total quantity discount bids, have been described in the literature and are used in procurement practice, there has not been a thorough comparison among those discount policies as of yet. This is the first paper, to use different types of cost functions to generate bids, which allowed us to analyze not only computation times, but also total spend of different discount policies.

Our results show that that realistic problem sizes can be solved in a matter of minutes, but that big multi item problems with only incremental volume discount bids are harder to solve than those with only total quantity discount bids. On the other hand if a supplier wants to approximate his true cost function with total quantity discount bids closely, this leads to a much larger and number of discount intervals.

There were several situations, where the results of simple split-award auctions with simple price quotes lead to lower total cost than those of more advanced bidding languages. The reason is either a bad approximation of the costs or the inability to find the cost-minimal solution within an acceptable response time. For example, if suppliers only used a few total quantity discount intervals, this led to much higher total cost for the buyer. However, we have also shown that significant savings can be achieved with compact bidding languages compared to split award auctions, if bidders are able to approximate their cost functions well. Even though big instances with few incremental price intervals could not be proven to be optimal, the resulting costs were always lower than the ones resulting of a split award auction.

In summary, a procurement manager needs to take care that the bidding language provides enough flexibility so that bidders can describe their cost structures arbitrarily close. At the same time bids should have low description length, such that suppliers are only forced to specify a few parameters and not hundreds of numbers. Also, a procurement manager should make sure that the number of bids in an application is such that he can expect to solve the problem instances to optimality. The results of our analysis should provide a better understanding under which circumstances compact bidding languages should be used in procurement practice. Mechanism design questions have been outside the scope of this paper, and remain fruitful questions for future research in this area.

While the bid language leads to much flexibility for the suppliers in specifying discounts in a compact format, it also incurs computational complexity on the buyer's side. Practical applications span a wide range of possible computational time requirements ranging from demanding interactive uses such as scenario analysis, and dynamic reverse auctions where acceptable computational times may be in the scale of minutes, to less demanding cases such as multiple round negotiations where runtimes in hours, often days, may be tolerable. We have analyzed the empirical hardness of SQS and the time to solve various problem sizes in order to understand the impact of different types of constraints and discounts on the runtime with a focus on interactive applications such as scenario analysis.

The results provide an understanding under which circumstances such expressive bidding languages can be used in procurement practice. We have also analyzed the potential impact of such bidding languages on the total spend of the purchasing manager, if bidders reveal their costs truthfully, and found that the cost savings depend on the parameters of the cost function and the bidding language used.

7.2 Outlook

Our main goal was to improve the performance of procurement auctions by giving the suppliers a better possiblity to express their costs, and the therby lowering the costs for the buyer, together with provinding a better understanding of the scenario. With our computational experiments we haven shown the feasibility of our aproach in priciple, but real events always incorporate human bidders, and therefore a verification of our results by lab experiments remains an important open issue.

Mechanism design questions have been out of the scope of this work and we have assumed extremely well behaving bidders, which clearly is a very strong assumption. A game theoretical analysis of the proposed mechanism could therefore provide valueable insights how bidders might behave in such a procurement event.

The capability of the bidders to express their cost, is crucial as we have seen, and could greatly benefit from a visual representation of their bids. The use of total quantity bidding is still dominating in real procment events, as they seem to appear easier to grasp in textual form. Already a very simple plot of the resulting costs, as for example as in section 5.2, could help in promoting the use of incremental discounts in our opinion, as they would demonstrate the resulting jumps in the cost function.

In iterative combinatorial auctions the bidders gain feedback in between rounds by the form of ask prices and can improve their bids accordingly. In practical settings further guidance of the bidders is necessary as Scheffel et al. (2010) have shown in their analysis of linear price auctions and therein also importance of bidder decision support. For combinatorial auctions we have seen that the simpler linear price auctions can perform very good, in contrast to their theoretical limitations.

In section 4.2.1 we have suggested a simple price feedback based on a lump sum discount request. It remains an interesting question if such a very simple feedback, that has the advantage of a very easy and fast comprehensiveness, could guide the buyers accordingly in an iterative action based on $\mathcal{L}_{ess}$ and $\mathcal{SQS}$.

Bibliography

Abrache, Jawad, Benoît Bourbeau, Teodor Gabriel Crainic, Michel Gendreau. 2004. A new bidding framework for combinatorial e-auctions. *Computers & Operations Research* **31**(8) 1177–1203.

Adomavicius, G., A. Gupta. 2005. Towards comprehensive real-time bidder support in iterative combinatorial auctions. *Information Systems Research (ISR)* **16** 169–185.

An, N., W. Elmaghraby, P. Keskinocak. 2005. Bidding strategies and their impact on revenues in combinatorial auctions. *Journal of Revenue and Pricing Management* .

Ausubel, L., P. Cramton, P. Milgrom. 2006. The clock-proxy auction: A practical combinatorial auction design. P. Cramton, Y. Shoham, R. Steinberg, eds., *Combinatorial Auctions*. MIT Press, Cambridge, MA.

Ausubel, L., P. Milgrom. 2006a. Ascending proxy auctions. P. Cramton, Y. Shoham, R. Steinberg, eds., *Combinatorial Auctions*. MIT Press, Cambridge, MA.

Ausubel, L., P. Milgrom. 2006b. The lovely but lonely vickrey auction. P. Cramton, Y. Shoham, R. Steinberg, eds., *Combinatorial Auctions*. MIT Press, Cambridge, MA.

Baumol, W. J. 1987. *Microtheory: Applications and Origins*. 1st ed. MIT Press, Cambridge, MA, USA.

Benisch, M., N. Sadeh, T. Sandholm. 2008. A theory of expressiveness in mechanisms. *23rd AAAI Conference on Artificial Intelligence*.

Bichler, M., A. Davenport, G. Hohner, J. Kalagnanam. 2006. Industrial procurement auctions. P. Cramton, Y. Shoham, R. Steinberg, eds., *Combinatorial Auctions*. MIT Press.

Bichler, M., J. Kalagnanam. 2005. Configurable offers and winner determination in multi-attribute auctions. *European Journal of Operational Research* **160**(2) 380–394.

Bichler, M., J. Kalagnanam, K. Katircioglu, A. King, R. Lawrence, H. Lee, G. Lin, Y. Lu. 2002. Applications of flexible pricing in business-to-business electronic commerce. *IBM Systems Journal* **41** 287–302.

Bichler, M., P. Shabalin, A. Pikovsky. 2009. A computational analysis of linear-price iterative combinatorial auctions. *Information Systems Research* **20**(1) 33–59.

Bikhchandani, S., J. M. Ostroy. 2002. The package assignment model. *Journal of Economic Theory* **107**(2) 377–406.

Blumrosen, L., N. Nisan. 2005. On the computational power of iterative auctions. *ACM Conference on Electronic Commerce (EC 05)*. Vancouver, Canada.

Boutilier, C., H. H. Hoos. 2001. Bidding languages for combinatorial auctions. *17th International Joint Conference on Artificial Intelligence (IJCAI) 2001*. Washington, USA, 1211–1217.

Chandru, V., J. Hooker, eds. 1999. *Optimization Methods for Logical Inference*. John Wiley & Sons.

Chaudhry, S., F. Forst, J. L. Zydiak. 1993. Vendor selection with price breaks. *European Journal of Operaitonal Research* **70** 52–66.

Chen, Yan, Kan Takeuchi. 2009. Multi-object auctions with package bidding: An experimental comparison of vickrey and ibea. *Games and Economic Behavior* **to appear** –. doi:10.1016/j.geb.2009.10.007.

Crama, Y., R. Pascual, A. Torres. 2004. Optimal procurement decisions in the presence of total quantity discounts and alternative product recipes. *European Journal of Operational Research* **159** 364–378.

Cramton, P., Y. Shoham, R. Steinberg. 2006. Introduction to combinatorial auctions. P. Cramton, Y. Shoham, R. Steinberg, eds., *Combinatorial Auctions*. MIT Press, Cambridge, MA.

Dang, V., N. Jennings. 2003. Optimal clearing algorithms for multi-unit single-item and multi-unit combinatorial auctions with demand-supply function bidding. Technical report, Dept. of Electronics and Computer Science, Univ. of Southampton, UK.

Davenport, A., J. Kalagnanam. 2000. Price negotiations for procurement of direct inputs. *IMA Ḧot TopicsẄorkshop: Mathematics of the Internet: E-Auction and Markets*, vol. 127. Minneapolis, USA, 27–44.

de Vries, S., J. Schummer, R. Vohra. 2007. On ascending vickrey auctions for heterogeneous objects. *Journal of Economic Theory* **132** 95–118.

Dunford, M., K. Hoffman, D. Menon, R. Sultana, T. Wilson. 2007. Testing linear pricing algorithms for use in ascending combinatorial auctions. Tech. rep., George Mason University.

Eso, M., S. Ghosh, J. Kalagnanam, L. Ladanyi. 2001. Bid evaluation in procurement auctions with piece-wise linear supplycurves. Technical report rc22219, IBM T.J. Watson Research Center.

Evans, D., J. Heckman. 1984. A test for subadditivity of the cost function with an application to the bell system. *American Economic Review* **74**(4) 615–623.

Gartner. 2008. Magic quadrant for sourcing application suites. Tech. rep., Gartner Consulting.

Gomory, R. E., W. J. Baumol. 1960. Integer programming and pricing. *Econometrica* **28** 521–550.

Goossens, D. R., A. J. T. Maas, F. Spieksma, J. J. van de Klundert. 2007. Exact algorithms for procurement problems under a total quantity discount structure. *European Journal of Operational Research* **178** 603–626.

Guzelsoy, M., T. Ralphs. 2007. Duality for mixed-integer linear programs. *The International Journal of Operations Research* **4** 118–137.

Hohner, G., J. Rich, E. Ng, G. Reid, A. Davenport, J. Kalagnanam, H. Lee, C. An. 2003. Combinatorial and quantity discount procurement auctions with mutual benefits at mars, incorporated. *Interfaces* **33**(1) 23–35.

Hooker, J. N. 2009. A principled approach to mixed integer/linear problem formulation. *INFORMS Computing Society Conference*. 79–100.

Jucker, J. V., M. J. Rosenblatt. 1985. Single period inventory models with demand uncertainty and quantity discounts: Behavioral implications and a new solution procedure. *Naval Research Logistics Quarterly* **32**(4) 537–550.

Kalecki, Michal. 1991. *Collected Works of Michal Kalecki*, vol. Volume II: Capitalism: Economic Dynamics. Oxford Univ P.

Katz, P., A. Sadrian, P. Tendrick. 1994. Telephone companies analyze price quotations with bellcore's pdss software. *Interfaces* **24** 50–63.

Kwasnica, T., J. O. Ledyard, D. Porter, C. DeMartini. 2005. A new and improved design for multi-objective iterative auctions. *Management Science* **51**(3) 419–434.

Kwon, R. H., G. Anandalingam, L. H. Ungar. 2005. Iterative combinatorial auctions with bidder-determined combinations. *Management Science* **51**(3) 407–418.

Lee, H. L., M. J. Rosenblatt. 1986. A generalized quantity discount pricing model to increase supplier's profit. *Management Science* **32** 1177–1188.

Leyton-Brown, K., E. Nudelman, Y. Shoham. 2009. Empirical hardness models: Methodology and a case study on combinatorial auctions. *Journal of the ACM* **56** 1–52.

Leyton-Brown, K., M. Pearson, Y. Shoham. 2000. *Towards a universal test suite for combinatorial auction algorithms*.

Leyton-Brown, Kevin, Eugene Nudelman, Yoav Shoham. 2002. Learning the empirical hardness of optimization problems: The case of combinatorial auctions. *Constraint Programming (CP)*. 556–572.

Milgrom, P. 2009. Simplified mechanisms with an application to sponsored-search auctions. *Games and Economic Behavior* **to appear**.

Mishra, D., D. Parkes. 2004. Ascending price vickrey auctions using primal-dual algorithms. Discussion paper, Harvard University.

Mishra, D., D. Parkes. 2007. Ascending price vickrey auctions for general valuations. *Journal of Economic Theory* **132** 335–366.

Munson, C. L., M. J. Rosneblatt. 1998. Theories and realities of quantity discounts: an exploratory study. *Production and Operations Management* **7**(4) 352–369.

Nisan, N. 2006. Bidding languages. P. Cramton, Y. Shoham, R. Steinberg, eds., *Combinatorial Auctions*. MIT Press, Cambridge, MA.

Nisan, N., I. Segal. 2001. The communication complexity of efficient allocation problems. *DIMACS workshop on Computational Issues in Game Theory and Mechanism Design*. Minneapolis, MI.

Park, S., M. H. Rothkopf. 2005. Auctions with bidder-determined allowable combinations. *European Journal on Operational Research* **161** 399–415.

Parkes, D. 2006. Iterative combinatorial auctions. P. Cramton, Y. Shoham, R. Steinberg, eds., *Combinatorial Auctions*. MIT Press, Cambridge, MA.

Parkes, D., L. H. Ungar. 2000. Iterative combinatorial auctions: Theory and practice. *17th National Conference on Artificial Intelligence (AAAI-00)*.

Pigou, A. C., ed. 1946. *The Economics of Welfare*. Macmillan & Co. Ltd., London.

Pindyck, RS, DL Rubinfeld. 2005. Microeconomics (6th edn).

Pisinger, D. 2005. Where are the hard knapsack problems? *Computers & Operations Research* **32**(9) 2271–2284.

Plott, C., T. Salmon. 2002. The simultaneous acending auction: Dynamics of price adjustment in experiments and in the u.k. 3g spetrum auction. Social Science Working Paper 1155, California Institute of Technology.

Porter, D., S. Rassenti, A. Roopnarine, V. Smith. 2003. Combinatorial auction design. *Proceedings of the National Academy of Sciences of the United States of America (PNAS)* **100** 11153–11157.

Rassenti, S., V. L. Smith, R. L. Bulfin. 1982. A combinatorial auction mechanism for airport time slot allocations. *Bell Journal of Economics* **13** 402–417.

Rothkopf, M. H. 2007. Thirteen reasons why the vickrey-clarke-groves process is not practical. *Operations Research* **55** 191–197.

Rothkopf, M. H., A. Pekec, R. M. Harstad. 1998. Computationally manageable combinatorial auctions. *Management Science* **44** 1131–1147.

Sandholm, T. 2008. Expressiveness in mechanisms and its relation to efficiency: Our experience from $40 billion of combinatorial multi-attribute auctions, and recent theory. *Proceedings of the 2nd International Workshop on Computational Social Choice*.

Sandholm, T., S. Suri. 2001. Market clearability. *Proceedings of the International Joint Conference on Artificial Intelligence*. IJCAI.

Sandholm, Tuomas. 1999. Approaches to winner determination in combinatorial auctions. *Decision Support Systems* **28**(1) 165–176.

Scheffel, T., A. Pikovsky, M. Bichler, K. Guler. 2010. An experimental comparison of linear and non-linear price combinatorial auctions. *Information Systems Research* **to appear**.

Schneider, S., P. Shabalin, M. Bichler. 2010. On the robustness of non-linear personalized price combinatorial auctions. *European Journal of Operational Research* **206**(1) 248–259.

Selten, R. 1975. A reexamination of the perfectness concept for equilibrium points in extensive games. *International Journal of Game Theory* **4** 25–55.

Silverson, R., E. A. Peterson. 1979. *Decision systems for inventory management and production planning*. John Wiler and Sons, New York.

Sowa, J. F. 2000. *Knowledge Representation: Logical, Philosophical, and Computational Foundations*. Brooks Cole Publishing Co.

Stewart, K. G. 2009. Non-jointness and scope economies in the multiproduct symmetric generalized mcfadden cost function. *Journal of Productivity Analysis* **to appear**.

Williams, H. P. 1996. Duality in mathematics and linear and integer programming. *Journal of Optimization Theory and Applications* **90** 257–278.