UNIVERSITY OF Southampton

University of Southampton Research Repository
ePrints Soton

UNIVERSITY OF SOUTHAMPTON

Distributed Mechanisms for Multi-Agent Systems: Analysis and Design

by

Rajdeep K. Dash

A thesis submitted for the
degree of Doctor of Philosophy

Faculty of Engineering, Science and Mathematics
School of Electronics and Computer Science
Intelligence, Agents, Multimedia Group

June 2006

ABSTRACT

There is an increasing need for multi-agent systems to operate under decentralised control regimes that support openness (individual components can enter and leave at will) and enable components representing distinct stakeholders with different aims and objectives to interact effectively. To this end, this thesis explores issues associated with using techniques from Game Theory and Mechanism Design to organise and analyse such systems. In particular, emphasis is given to distributed mechanisms in which there is distributed allocation (no single centre determines the allocation of the resources or the tasks) and distributed information (agents require information privately known by other agents in order to determine their own valuation or cost). Such mechanisms are important because, in comparison to their centralised counterparts, they are robust to a single-point failure, the computational burden can be potentially shared amongst many agents, and there is a reduction in bottlenecks since not all communication need pass through a single point. As a result, distributed mechanisms are better suited to many types of multi-agent application.

To provide a grounding for the mechanisms we develop, the thesis contains a running example of a multi-sensor network scenario. In these systems, distributed allocation mechanisms are desirable since they are robust and reduce bottlenecks in the communication system. Furthermore, we show that distributed information naturally arises by deriving an information-theoretic valuation function. This scenario also gives rise to two additional requirements that are addressed within this thesis: (i) constrained capacity, whereby suppliers can only provide a limited amount of goods or services at any given time and (ii) uncertainty in task completion, whereby sensors potentially fail after they have been assigned tasks.

Specifically, we focus on the Vickrey-Clarke-Groves (VCG) mechanisms and investigate ways of extending it so as to address the requirements that arise within distributed setting in general and sensor networks. In particular, we choose the VCG as our point of departure since it is a mechanism that is efficient, individually rational and incentive compatible. Unfortunately, it is brittle in the sense that it does not conserve these desirable properties when considering the requirements that we outlined above. Therefore, we develop novel mechanisms that do.

In more detail, the first part of this thesis considers two distributed allocation mechanisms — a simultaneous auction environment and Continuous Double Auction (CDA). In the former, bidders place sealed bids in a number of selling auctions which are concurrently offering items. This results in a distributed allocation whereby the winner at each auction is determined by the seller conducting it. For this case, we derive the optimal strategy of the bidders using a game-theoretic approach. In the CDA, buyers and sellers, respectively, submit bids and asks continuously and the market clears when a bid is higher than an ask; meaning that the allocation is again determined in a distributed way. Furthermore, CDAs are known to yield close to efficient allocations, under certain conditions, even when utilising very simple strategies. However, in

our case, we need to modify their format in order to deal with the requirement of constrained capacity. In both of these mechanisms, we study the system's loss in efficiency that ensues from distributing the allocation and find that it is $\frac{1}{e}$ in the simultaneous auction case and upto 35% in the continuous double auction case.

The second part of this thesis is concerned with designing mechanisms when agents have distributed information within the system. Such settings are more general than those more traditionally studied in that they encompass the fact that agents can potentially change their valuation or cost upon knowing a signal about the system (which they have not observed) that was hitherto unknown to them. Specifically, we first show that interdependent valuations arise naturally within a sensor network when we develop an information-theoretic valuation function. To account for this, we significantly extend the VCG mechanism in order to deal with these interdependent valuations. We then go on to develop a mechanism that can deal with uncertainty in task allocation. In both of these cases, our mechanisms are shown to be efficient, individually rational and incentive compatible. Moreover, their computational properties are studied and efficient algorithms are designed (based on linear and dynamic programming) in order to speed up the computation of the allocation problem which is generally $\mathcal{NP}$-hard.

Contents

List of Figures

List of Tables

Nomenclature

Calligraphic Symbols

$\mathcal{K}$ Set of possible allocations.

$\mathcal{I}$ Set of (buyer) agents.

$\mathcal{J}$ Set of (seller) agents.

$\mathcal{M}$ Mechanism specifying how an allocation should be determined.

Greek Symbols

$\widehat{\theta}_i$ Agent i's reported type.

Γ Game Form.

Θ_i Set of types for agent i.

θ_i Type of (or independent signal observed by) agent i. $\theta_i \in \Theta_i$.

Roman Symbols

$c_i(.)$ Cost of agent i of performing tasks specified in (.) .

$E_{\theta_{-i}}[.]$ Expectation of $[.]$ given the types of all other agents.

$f(.)$ Probability density function of (.).

$F(.)$ Cumulative density function of (.).

$g(.)$ Outcome function.

i Index of one agent in the set of agents $\mathcal{I}$

$-i$ The set of agents $\mathcal{I} \setminus i$.

j Index of one agent in the set of agents $\mathcal{J}$

$-j$ The set of agents $\mathcal{J} \setminus j$.

$\widehat{K}^*$ Implemented efficient allocation given reported types $\widehat{\theta}_i$. May not correspond to efficient allocation if agents lie.

K	A particular allocation.
K^*	Efficient allocation, i.e. the one which maximises the sum of utility of all agents.
k_i	Allocation to agent i
M	Number of sellers in an auction.
N	Number of buyers in an auction.
r_i	Overall transfer to agent i
$scf(.)$	Social choice function describing a particular desiderata.
s_i	Strategy i has selected to use.
S_i	Strategy space of agent i.
u_i	Utility of agent i.
$v_i(.)$	Value that agent i holds for a good/allocation.

Acronyms

CDA Continuous Double Auction

CMD Computational Mechanism Design

DMD Distributed Mechanism Design

MD Mechanism Design

MSN Multi-Sensor Network

MAS Multi-Agent System

VCG Vickrey-Clarke-Groves

ZI zero-intelligence

POS Probability of Success

LP Linear Programming

DP Dynamic Programming

Acknowledgements

I would first like to express my heartfelt gratitude to my supervisor Professor Nick Jennings. His help, insight, encouragement and patience have been greatly helped to steer me into the right direction. The discussions we had were always helpful and taught me the essential skills of research. His rapid and efficient turnover of papers has always amazed me. The dreaded “why?”s that peppered initial versions of a paper resulted in more grounded research and have sharpened my inquisitiveness. Work iteratively became stronger, better explained and more coherent at each review process.

I would also like to acknowledge the people without whose help and advice this thesis would not have been possible. Jusso Valïmaki and Robin Mason at University of Southampton introduced me to microeconomics and game theory. Their lectures were so clear that even coming from a background of electronic engineering, I soon managed to get a grasp of these theories. David Parkes at Harvard University provided me with a lot of insight during my visit into the world of computational mechanism design. I also had interesting discussions on sensor networks with Andy Wright at BAE systems. These led to much of the motivation behind the research carried out within this thesis.

During the course of this thesis, I have had the pleasure of collaborating with number of outstanding individuals who each contributed in making this research possible. Enrico Gerding and David Yuen worked with me on research in the simultaneous auction environment. Krishnen Vytelingum, Minghua He and Adam Sykulski collaborated with me on a number of papers linked with the continuous double auction protocol, as well as on strategy design. Viet Dung Dang, Alex Rogers, Steve Reece, Iead Rezek and Paritosh Padhy worked on sensor-related papers with me. Alex Rogers also collaborated on research in interdependent mechanism design, as well as in market mechanisms for constrained capacity suppliers. Esther David contributed to the latter research as well. Gopal Ramchurn, Juan Antonio Rodriguez, Andrea Giovannucci and Bastian Blankenburg worked with me on trust-based scenarios.

I also gratefully acknowledge the financial support of the Overseas Research Studentship and the BAE studentship. This meant that I was not hampered by financial constraints and instead could dedicate my time to research. Also, the AAMAS scholarships (2004,2005 and 2006) and EUMAS scholarship (2004) enabled me to attend these very relevant conferences.

During my time as a PhD student, I have been able to interact within a number of social circles within which I have met many friends. These friends have ensured that my time within Southampton has been a very enjoyable experience. I would like especially mention the IAM group, the LCU and the SCR as the three most important circles.

Last, but not least, I would like to thank my family who have provided me with unstinting support during the course of my thesis.

To the Guiding Hand . . .

Chapter 1

Introduction

There is an increasing need for computer systems that operate a decentralised control regime, that are open (individual components can enter and leave at will) and that contain a number of components representing distinct stakeholders with different aims and objectives. Relevant examples include grid computing [Foster and Kesselman, 1999], the semantic web [Berners-Lee, 1999], pervasive computing [Huang et al., 1999], e-commerce [Wellman, 2004], mobile computing [Radrinath et al., 1993] and peer-to-peer systems [Shneidman and Parkes, 2003]. For these complex systems, it has been argued that agent-based approaches, with their emphasis on autonomous actions and flexible interactions, are a natural computational model [Jennings, 2001]. In such Multi-Agent Systems (MASs), there are two fundamental design issues that need to be addressed. First, there is a need to specify the *protocols* that govern the interactions. These cover issues such as how the actions of the agents translate into an outcome, what range of actions are available to the participants, and whether the interactions occur over a series of steps or are one-shot. Second, given the prevailing protocol, there is a need to define the *strategy* (mapping from state history to action) for each agent.

Now, in some cases, a designer may be able to impose both the protocol and the strategy of each agent. In such settings, the agents can cooperate to find a good system-wide solution [Padhy et al., 2006; Jennings and Bussmann, 2003; Pynadath and Tambe, 2003; Shoham and Tennenholtz, 1992]. This cooperation amongst agents can be structured using a variety of planning, distributed constraint optimisation, coalition formation and scheduling algorithms that have been proposed [Lesser and Corkill, 1981; Durfee and Lesser, 1989; Boutilier, 1999; Ramamritham et al., 1989; Yokoo et al., 1998; Dang et al., 2006]. However, such methods fail in systems where the agents represent distinct stakeholders whose aim is to maximise their own profit in the system (e.g. in Grid computing where the agents represent different end users and in e-commerce scenarios where the agents represent the buyers and sellers). They fail because in such cases they present the opportunity for the agents to gain an advantage by misreporting their position (either their needs or their resources). For example, an agent might over-report its need for memory capacity on a computational grid so that when the distributed constraint optimisation process is carried out, it gets allocated more memory than its share in an efficient allocation.

Another example is in peer-to-peer systems where the case of free-riding (i.e. where agents under-state their available resource so as not to be asked to contribute to the system) has been well documented [Adar and Huberman, 2000]. In both of these cases and many others besides, the safest assumption to make is that if agents can act so as to get more benefit, then they will do so. Thus, by default, agents should be assumed to be self-interested, rational problem solvers.

Stated in this way, it is obvious that microeconomics [MasColell et al., 1995] — the study of the decision-making behaviour of self-interested agents as they interact with their environment — should be able to provide useful insights into the design process for systems that operate a decentralised control regime. Specifically, a clear parallel emerges between the self-interested agents that are trying to find their best strategy in large, open, distributed computer systems and the economic model of rational beings trying to maximise their gain from a market. In particular, there are two points of focus from which a designer needs to carry out a non-cooperative strategic analysis. In the first one, the designer of a system can only impose the protocol (and has no control over what strategies the agents adopt) and designs it so as to ensure that certain properties are guaranteed within the desired protocol. In the second one, the designer of a participating agent is faced with a particular system having a pre-specified protocol and designs the strategy of an agent such that it maximises its utility (or profit) in the system.

Given this insight, this thesis focuses on applying the theories developed in microeconomics to the analysis and design of distributed protocols for MASs, that is, protocols in which the allocation of resources and the gathering of information are carried out by multiple agents (cf. the gathering of information into a single agent (the centre) which then determines the allocation in centralised protocols). In fact, these market-based techniques are already starting to be applied in domains such as grid computing [Wolski et al., 2001], peer-to-peer systems [Shneidman and Parkes, 2003], multirobot coordination [Gerkey and Mataric, 2002b] and mobile computing [Bredin et al.]. In this vein, in this thesis, we choose the particular application scenario of Multi-Sensor Networks (MSNs) where each sensor node is represented as an agent (the justification for this choice is given in section 1.2). Thus, we will take into consideration the particular constraints that these MSNs impose on the design process. Specifically, in a MSN, a distributed control scheme is preferred since a trusted centre that decides on the outcome may not always be present or desirable (since it is then a critical single point of failure). Furthermore, as a result of physical and temporal constraints, a single sensor may not be able to be tasked to do all the readings required within the system (e.g. the maximum number of readings a sensor can make may be limited by its battery power or the maximum swivel speed of its sensor head). Moreover, in MSNs, the distributed information gathered is typically fused together which means that the value of an observation is contingent on signals that are observed by other agents. Finally, sensors might fail in undertaking tasks that have been assigned to them. These failures may occur due to uncontrolled reasons (e.g. a sudden battery failure will stop a sensor from making a reading of the environment) or due to a conscious decision (e.g. the sensor diverts resources to another more rewarding task).

In dealing with these constraints, the first part of this thesis analyses and designs markets in which the allocation is not determined centrally by a controlling agent. In the second part of the thesis, mechanisms which deal with distributed information are designed. In both parts, we are also concerned with developing protocols that are resilient to the deficiencies of the individual agents (such as their limited capacity to perform tasks and provide resources as well as their propensity to fail). Specifically, this thesis addresses the following requirements:

Requirement 1. *Distributed Allocation*: The allocation of tasks and resources within the system should be carried out without the use of a central controller.

Requirement 2. *Limited Capacity*: The protocol should be able to deal with the situation whereby individual agents being limited in the number of tasks they can carry out.

Requirement 3. *Distributed (Interdependent) Information*: The protocol should incorporate the fact that agents may form their preferences over the allocations based on private signals observed by other agents.

Requirement 4. *Uncertainty in Task Completion:* The protocol should be robust to the fact that certain agents may fail to successfully carry out the tasks assigned to them.

It should be noted that whilst these requirements are inspired from a MSN scenario, the analyses and solutions we present are broadly applicable to open MASs in general. To this end, we now provide a background in order to position the challenges that this thesis has addressed in the broad set of challenges that MAS designers encounter when designing and analysing distributed mechanisms.

1.1 Background

In this section, we provide a more detailed description of the problems that this thesis seeks to address. Specifically, we will provide an overview of the economic foundations of this endeavour and then detail the challenges that become more imperative in a MAS (as opposed to traditional economic settings). We then give a broad overview of the MSN scenario which is employed as a running example throughout this thesis. In so doing, we provide the background for our work by positioning it on the canvas of challenges that need to be addressed when designing MASs and, more specifically, MSNs with selfish agents.

1.1.1 Economic Foundations

In micro-economics, there are two strands dealing with the result of aggregating the decision-making of individual agents:

1. Theory of Competitive Equilibrium [MasColell et al., 1995]. This studies the *equilibrium* conditions (conditions characterising the stable state) that arise when a *large* number of agents compete with each other in a given environment. Here, each agent is assumed to be rational in that it tries to maximise its *utility* (a measure of the "goodness" the agent derives from the outcome), based on its information about the environment.

2. Game Theory [Osborne and Rubinstein, 1994]. This studies the behaviour of agents in interactive decision problems where the actions of one agent affect both the selected actions of another and the resulting equilibrium.

The main difference between these two theories is in how the agent models its environment before making a decision. The former assumes that each agent is interested only in some environmental parameters (such as prices and availability of resources), whereas in the latter each agent additionally takes into account the behaviour of other agents and how they may influence these parameters. Thus in game theory, the behaviour of other agents is incorporated into an agent's decision making process[1]. Though we investigate both these approaches in this thesis, the work described herein focuses mainly on the latter. This is because it is a more principled way of achieving desirable properties in a MAS (in that it relies on mathematical models to prove certain properties, rather than experimental evaluation). Furthermore, as we discuss in chapter 2, designing systems using game-theoretic approaches gives us the necessary conditions for designing systems where there might not be a central controller. In particular, we use game theory to design multi-agent systems that address the last three requirements listed above (limited capacity, interdependent information and uncertainty in task completion). However, one important shortcoming of designing systems using game theory is that the resultant system is almost invariably centralised[2]. On the other hand, approaches adopting the competitive equilibrium tend to ascribe very little power to a centre or have no centre at all. Thus, these approaches can shed light on how to design systems using game theoretic techniques, but which have distributed allocation mechanisms.

To date, both of these approaches have been investigated in MASs. Wellman's seminal work on Market Oriented Programming [Wellman, 1993] was based on the competitive equilibrium approach and has subsequently been extended to numerous applications [Clearwater, 1996; Kraus, 2001]. In this work, the main point of focus has been the design of agent strategies for relatively complex market institutions in which the agents are assumed to be selfish but not strictly rational (as defined in Chapter 2)[3]. In particular, strategies have been developed using various above mentioned heuristics for these specific settings [He et al., 2006; Vytelingum et al., 2004;

[1]In very large systems the two theories yield similar models and answers since a single agent then has little effect on the whole environment (especially if the environment is nearing equilibrium).

[2]One could argue that this shortcoming is a result of researchers using the revelation principle (discussed in chapter 2) too literally, rather than of game theory in itself. However, the fact is that most systems designed using game theory involve a centre.

[3]This occurs since the agents make decisions without considering the full impact of their actions on other agents Friedman and Rust [1992]. This may be due to their lack of knowledge of the other agent's action set, the payoff matrix, or the fact that they believe that their actions will not have an impact in large market.

Airiau and Sen, 2003; Byde et al., 2000; Yarom et al., 2004]. In this context, one of the simplest and most commonly studied institutions is the CDA in which traders submit offers to buy (bid) and offers to sell (ask) at any time during the trading period and in which the market clears[4] continuously [Friedman and Rust, 1992]. As a result, certain of the global properties of such institutions, such as speed of convergence of the market towards the equilibrium and the proximity of experimental and theoretical equilibria[5], rely on the particular strategies that the agents adopt. In fact, in [Vytelingum et al., 2004] we design a strategy which performs better than current strategies in both a local (profit-maximising) and global context. However, in this thesis, since we concentrate on the baseline performance of protocols, we employ the methodology advocated by [Gode and Sunder, 1993] and apply it to the CDA protocol we design in chapter 4 (thereby simultaneously addressing requirements 1 and 2).

Game theory has also been heavily used to analyse and design strategies in various markets [Engelbrecht-Wiggans and Weber, 1979; Szentes and Rosenthal, 2003; Fatima et al., 2005]. Such game-theoretic approaches to the design of strategies (which we use in chapter 3 for a distributed marketplace) differ from the heuristic approaches in that they yield predictable equilibrium strategies under the assumption that the agents are rational. As a result, the design of protocols for predictable systems (i.e. systems in which certain global properties can be guaranteed) within MASs has been mainly based on Game Theory [Parkes, 2001; Zlotkin and Rosenschein, 1996; Sarne and Kraus, 2005; Sandholm, 2003; Wurman et al., 2001]. Furthermore, this approach models the interactions between agents mathematically, resulting in a more principled way of building protocols whereby the properties of the protocol can be proven or disproven theoretically rather than empirically. In particular, the techniques used are drawn from Mechanism Design (MD) which is the area of micro-economics concerned with how to design systems, using tools developed by game theory analysis (e.g. Nash Equilibrium, Dominant Strategy, Bayesian-Nash equilibrium), such that certain system-wide properties (e.g. efficiency, stability, fairness) emerge from the interaction of the constituent components. Here the *mechanism* is viewed as the whole system; consisting of the set of agents with their utility functions, their action sets and the protocol. In contrast to Market Oriented Programming, agents in this case will always adopt *one* strategy since they are incentivised to do so as a result of the design of the protocol[6]. In the MAS context, MD has been mostly used for the design of auction protocols for the allocation of resources and tasks and in this thesis we study how to design mechanisms under the constraints imposed by MASs (chapters 4, 5 and 6). However, we will first discuss the broad range of challenges that arise in designing MASs using MD (sections 1.1.2 and 1.1.3), before focusing on the specific challenges we address in this thesis.

[4]The market clears as soon as a bid exceeds an ask.

[5]The theoretical equilibrium is the one achieved as the number of agents in the market tends to infinity.

[6]The adoption of one strategy is the objective of the design of the mechanism. In certain protocols, agents may face multiple equilibria, in which case the system designer can introduce a correlating device so as to favor the adoption of one of the equilibria [Fudenberg and Levine, 1993].

1.1.2 Computational Challenges

As a result of the assumptions made in traditional MD, its application in MASs is not straightforward. In traditional MD, for example, agents are assumed to be rational and no consideration is given to how computationally hard it is to select the appropriate strategy, the centre is assumed to be able to compute the outcome of the protocol once the agents have transmitted their strategies to it, the agents are assumed to undertake and successfully complete tasks assigned to them, communication between the agents and the system is generally assumed to be free and faultless, and the system is assumed to know the number of agents that are present [Dash et al., 2003; Parkes, 2001; Rubinstein, 2002; Conitzer and Sandholm, 2002; Dash et al., 2004; Glazer and Rubinstein, 1998]. However these assumptions are problematic in computational settings because in addition to the issues that we highlighted in the previous section, the following challenges are also present:

1. The mechanism will not be able to compute the outcome if this is an intractable problem (e.g. computing the allocation in certain types of sealed bid auctions is what prevented their application in FCC spectrum auctions [Klemperer, 2002]).
2. The agents themselves do not have the unbounded computational power required for calculating their preferences for all possible outcomes as is required, in general, to produce an optimal strategy.
3. Communication is not necessarily cost-free and may also be prone to errors.
4. The set of agents may vary with time due to the open nature of the system.
5. The presence of money, a common denominator by which every good can be valued, is an important component in traditional MD. However, in many MASs, such a common numéraire does not exist naturally and in many cases has to found or constructed.

The field that seeks to address some of these limitations and, thereby, apply MD techniques to computational problems is called Computational Mechanism Design (CMD). It could be argued that a new field is not required since we can decompose the problem of using MD in a MAS into its economic part (MD) and its computational part (MAS) and then attack the problem in a modular fashion. However this approach fails to recognise that at each stage of the design process both economic and computational principles need to be addressed. In fact, in many cases, principles from one of the areas can help to solve a problem in the other. For example, one could make finding an undesirable equilibrium strategy (economic problem) so intractable (computational solution) that no agent would be able or wish to do so. Similarly, one could make optimal strategies tractable (computational problem) by designing mechanisms that have a simple, truth-revealing equilibrium (economic solution).

However, despite the range of challenges that are present in CMD, so far research has mainly concentrated on specifying centralised protocols that operate under the constraints imposed by limited computational resources [Parkes, 2001; Sandholm, 2003; Nisan and Ronen, 1999][7]. As a result, there is an inherent assumption in most existing work in this area that the agents have a direct line of communication with this centre and can play their strategies or reveal their types simultaneously. Furthermore, it has been assumed that each agent forms its valuation based solely on information that is privately observed by it. However these assumptions do not always hold in distributed open systems and hence there is a strong need to move from CMD into the realms of Distributed Mechanism Design (DMD). To this end, the next section details the additional challenges involved in moving from CMD to DMD.

1.1.3 Distribution Challenges

DMD is concerned with the design of large-scale distributed systems consisting of multiple autonomous, selfish and rational agents in which there is no centre imposing an outcome and in which both the information or communication protocol are distributed over the agents. Thus, DMD is still concerned with the computational problems outlined in the previous section, but it differs in that there is no centre that decides on the outcome and the information an agent uses to make its choices is distributed. Such distributed mechanisms have a number of *advantages* over their centralised counterparts including:

Tractability. A distributed mechanism allows the burden of computation to be transferred from a central node in the mechanism to the numerous constituent agents that go to make up the system. This is akin to transforming the problem into a distributed optimisation task that exploits the computational resources of many agents.

Robustness. In a centralised mechanism, the communication channels linking the centre are critical for the system and failure may incapacitate the operation of the entire system. However in a decentralised setting, failure of these channels will not incapacitate the mechanism, though it may lead to a sub-optimal solution.

Trustworthiness. The issue of trustworthiness in the centre is an ever-present problem in a centralised mechanism. This is, we believe, a factor that has limited the use of Internet auctions since the agents have to trust that the auctioneer will not manipulate the mechanism for its own profit. In a distributed mechanism, since there is no central agent computing the outcome, there can be a higher degree of trust in the mechanism once the incentive issues to do with agents are addressed. However, the problem of trust between agents assumes greater importance in this context than in the centralised mechanism since now each agent depends on each other to carry out the mechanism.

Reduction of Bottlenecks. In distributed mechanisms there is no longer a single point through which all communication has to pass. Thus, there is no longer such an obvious and large bottleneck.

[7]The work is also referred to as Algorithmic Mechanism Design by [Nisan and Ronen, 1999].

However, these advantages come at a price. At one extreme, it may be possible to dispense with MD altogether and simply ask about the "price of anarchy" [Papadimitriou, 2001; Roughgarden, 2002] or the economic cost of just implementing distributed solutions with no carefully designed mechanism. In many cases, we believe that this cost will be too high and the challenge for the community remains in designing distributed mechanisms that retain the normative goals of MD. To this end, we explain the three core challenges in DMD [Dash et al., 2003]:

1. **Distribution of allocation**: In traditional MD, there is a centre that computes and enforces the outcomes. However, in DMD, we aim to study mechanisms in which such a centre does not exist. Such a constraint may arise naturally due to the computational structure of a system (as discussed above) or may be imposed by the system designer seeking the advantages of a distributed mechanism. For example, in current P2P networks, such as Gnutella and KaZaa, *free-riding* is a well-documented problem. In response, a number of studies have considered the economics of tit-for-tat where agents can only receive resources to the degree that they contribute them [Lai et al., 2003]. However, such a tit-for-tat approach is blind to the heterogeneity of local agents, that will likely differ in their computational resources and data content and quality. To this end, a classical approach in which the allocation is not computed by a central agent has been to implement markets akin to the CDA. However, such mechanisms rely on multiple trades before a stable allocation is determined. Furthermore, the assumption of direct lines of communication to a central information repository is still present (i.e. agents know which current items are available and at what price they are trading). As a result, there is a heavy communication load in such mechanisms. Another approach in this context is to implement a mechanism using distributed algorithms, whilst addressing the additional incentive considerations that occur when the same agents (with the set of agents being greater than two) that are implementing the mechanism are also strategic and self-interested [Shneidman and Parkes, 2003; Parkes and Shneidman, 2004]. However, whilst the computational burden has been removed from the centre in this approach, the centre is still required in order to enforce the mechanism. Finally, another line of research has investigated the use of distributed auctions held simultaneously by a number of sellers in order to allocate goods [Engelbrecht-Wiggans and Weber, 1979; Krishna and Rosenthal, 1996; Gerding et al., 2006a,b; Greenwald et al., 2001; Airiau and Sen, 2003; Preist et al., 2001]. In this case, whilst there is no centre either computing or enforcing the mechanism, the strategic decisions made by the agents need to be studied and the equilibrium is not guaranteed to be efficient.

2. **Distribution of information**: In traditional MD, agents are assumed to privately observe an idiosyncratic signal (such as their tastes and preferences) and then formulate their valuation as a function of this signal. This signal is commonly referred to as the type of the agent and the resulting valuation is known as its *private* valuation. However, in distributed settings agents often form their valuations of the items based on the information observed

by other agents, resulting in *interdependent* valuations[8]. For example, in grid computing, an agent estimating the cost of performing a particular task would base its estimate on which resources it will require in order to complete the task. These resources will typically be distributed over the grid with a number of agents having access to them. As a result, it will have to base its cost estimate on the signals observed by other agents, thereby resulting in interdependent valuations. Thus an agent might require x seconds on a microprocessor (to which it has direct access and can observe) and y bytes of memory (which another agent monitors) to complete a task. Another example is in P2P systems, where very often the value of a particular download to an agent might depend on how much the other agents value that download. This is especially true for downloads of systems that require a network to work, such as online games and chat engines where there has to be a sufficient number of users for the download to be worthwhile or when downloading files that appeal to a large base of users such as popular songs and movie clips.

3. **Distribution of communication system**: In DMD we can no longer assume direct lines of communication and hence need to rely on the agents themselves routing information for other agents. For example, in an inter-domain routing problem, each of the nodes routing traffic can be considered as an agent in a MAS. Hence, since these agents can lie about the cost incurred in passing messages in their routes, a DMD approach is required to provide agents with incentives to reveal truthful information and support the selection of the shortest path for the routing of messages. However this efficient outcome should be computed without overburdening the network with messages just to find it! Research in this area has developed mechanisms that have been tailored to specific topologies [Feigenbaum et al., 2002, 2001]. This, in turn, may point to adopting a design methodology similar to that in traditional MD, where solutions and mechanisms are developed for restricted topologies (as opposed to specific utility functions and trading environments in MD).

In summary, in this section, we have provided a list of the key challenges that exist when designing MASs using MD. However, as stated earlier, we shall concentrate in this thesis on four specific requirements that need to be addressed. Requirements 2 and 4 concentrate on challenges that fall under CMD, while requirements 1 and 3 address the first two challenges of DMD. We note here that we will not be addressing the challenges of distributed communication since we feel that it has been already been researched quite extensively [Shneidman and Parkes, 2003; Feigenbaum et al., 2002, 2001; Monderer and Tennenholtz, 1999; Bachrach and Rosenschein, 2005; Anderegg and Eidenbenz, 2003]. We next provide an overview of the MSN scenario, from which these requirements have been inspired.

[8]This problem is documented in purely economic settings as well (such as wildcatters bidding for a strip of land with potential resources under it [Krishna, 2002; Dasgupta and Maskin, 2000; Jehiel and Moldovanu, 2001]). However, it becomes more endemic in computational settings since as a result of the network of agents, information gathered by one agent more often than not affects that of the others.

1.2 Running Scenario

Multi-Sensor Networks (MSNs) are networks of small sensor nodes where each node typically consists of a micro-controller, a radio front-end, a power supply and one or more sensors for sensing the physical environment [Akyildiz et al., 2002; Padhy et al., 2006; Deshpande et al., 2004; Rao et al., 1991; Culler et al., 2004]. As such, they require a decentralised control regime (pertaining to both the way that the sensors perform their tasks and where information is distributed amongst the sensors) and are potentially open systems with distinct stakeholders. Hence, they provide a compelling area for the application of MAS since they are open, dynamic systems in which there are numerous points at which decisions and actions have to be carried out. Specifically, each agent (residing in the micro-controller) takes decisions on the following aspects (as shown in figure 1.1):

1. Task Scheduling: The agent decides the timing and nature of the sensing tasks that the sensor node should carry out.
2. Resource Allocation: The agent decides on the apportionment of the limited resource (e.g. power, bandwidth and/or computational resource) between the different tasks it may be required to carry out.
3. Communication Protocol: The agent decides the sources from which to receive data, the data it will transmit and the sinks to which to transmit data.

Thus, the network of sensors, in which each sensor is autonomously deciding on its actions and resource usage, can naturally be represented as a MAS. Now, in cases where all the sensor nodes are owned by a single stakeholder, this is best modelled using a cooperative MAS approach in which the agents are designed so as to work in tandem towards the system goal [Padhy et al., 2006; Deshpande et al., 2004; Akyildiz et al., 2002]. However, there are increasingly applications where each sensor (or group of sensors) may be individually-owned by different stakeholders. Such scenarios occur in applications like traffic control where each sensor is owned by a particular vehicle [Wu et al., 2005], in pico-satellite projects where multiple companies own very small satellites monitoring a certain area [Heidt et al., 2000], and in disaster relief examples where different governmental and non-governmental organisations share information gathered by their sensors to coordinate efforts in a natural disaster [Jennings et al., 2006]. In such applications, the sensors are operating in competitive rather than cooperative environments, and, as such, will attempt to optimise their own gain from the network, at a cost to the overall performance of the entire network. This selfish sensor perspective can still be applied when a group of sensors are owned by different stakeholders. In this case, there can be a broker sensor that offers/seeks services on the behalf of the network. These services will be comprised of collections of the atomic services provided by the individual sensors and thus the task allocation problem would occur both at the level of the market and the broker level. We do not investigate this perspective any further within this thesis since we believe that the groundwork for individual selfish sensors should first be laid before investigating such hierarchical systems.

In more detail, figure 1.1 shows a typical sensor network with the physical and agent-based representation of each sensor node. The sensors within this MSN have three possible capabilities, namely to sense the temperature and pressure of the environment, as well as visually track targets. Each sensor has a schedule of sensing tasks during which they gather data from the environment. They then send and receive data from each other using the bandwidth constrained communication links. In this thesis, we study how these types of MSNs can be managed using the economic mechanisms we develop. In particular, the requirements that have been outlined above are all reflected within the MSN depicted in figure 1.1 in the following way:

1. Within a MSN, sensors can be tasked by other sensors to sense various signals from the environment against a payment. Now, a trusted centre may not exist in such scenarios leading to sensors holding simultaneous auctions for their services. This scenario, from which requirement 1 is inspired, in studied in greater detail in chapter 3.

2. Sensors are typically constrained by the amount of power and/or bandwidth which is available to them. This leads to the sensor being able to carry out only a few of the total tasks that are demanded. As a result, we incorporate such sensors with limited capacity (requirement 2) into the design of the mechanism in chapter 4.

3. The value for the particular data gathered by a sensor can depend on the data which has been gathered by other sensors. This is especially true when sensors fuse information and results in the interdependent valuation model (requirement 3) considered in chapter 5.

4. A sensor may not always report a true value due to a variety of reasons including faults, maliciousness or noisy communication channels. Thus requirement 4 is inspired from such a failure-prone environment and is dealt with in chapter 6.

Having thus provided the scenario for this thesis, we now briefly cover some of the related research in the area of market-based task allocation within sensor networks. The specifics of the related work on each of the above issues are dealt with in the corresponding chapter.

1.2.1 Market-Based Task Allocation in Sensor Networks

Task allocation within sensor networks has traditionally been analysed under the assumption that the sensors will work towards the global objective of the MSN. This has been due to the fact that most of these MSNs are owned by a single stakeholder and has thus resulted in the adoption of cooperative approaches [Padhy et al., 2006; Lesser et al., 2003; Clearwater, 1996].

However, market-based techniques are increasingly being adopted in order to control sensor networks. Within this space, a number of approaches consider imbuing the sensors with selfishness

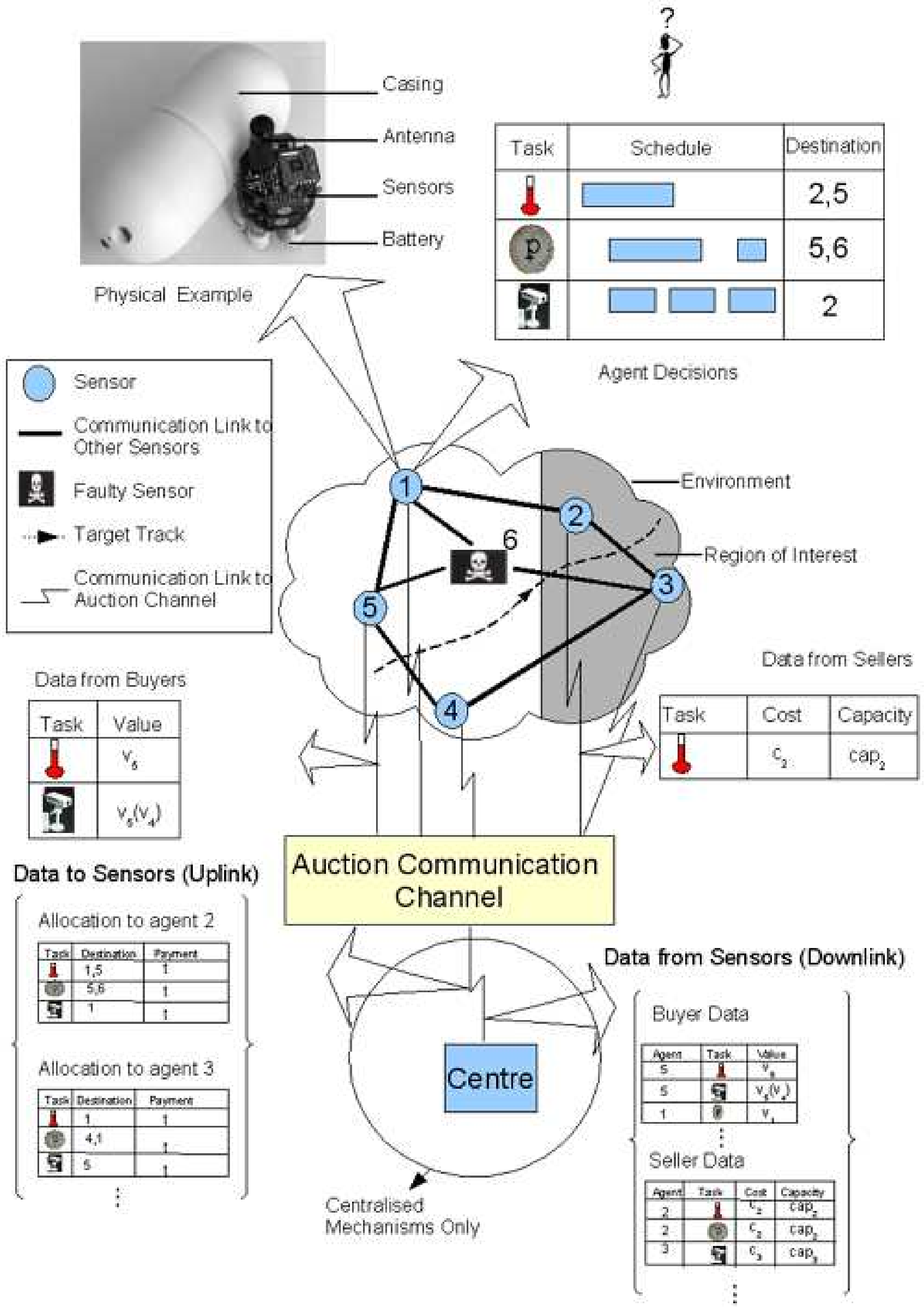


FIGURE 1.1: An overview of a MSN showing the physical components of a sensor node, the decisions faced by an agents controlling a node, and common problems encountered within the network.

in order to achieve a distributed control regime (e.g. [Gerkey and Mataric, 2002a; Sadagopan and Krishnamachari, 2004; Rogers et al., 2005]), whilst [Dash et al., 2005] consider sensors that are selfish due to their distinct ownerships. We consider each of these main items of related work in turn.

In more detail, Gerkey and Mataric [2002a] develop the MURDOCH protocol, which is loosely based on a multiple auction model, for task allocation amongst a system of robots (which is a collection of sensors and actuators). More specifically, in one particular instance of this protocol, the following correspondence with auctions would hold: the item being auctioned would be a task, the auctioneer would be the robot requesting the task, and the bid is a fitness measure which other idle robots provide. However, since they are operating within a cooperative environment no payments are made in the system. Rather, selfishness and the auction protocol are used as a means of carrying out a distributed allocation mechanism. The concept of selfishness is further investigated in [Sadagopan and Krishnamachari, 2004] where they study how the routing of data from sensors to a certain destination in the MSN. They construct a game whereby despite the selfish actions of the agents, an optimal load-balanced data gathering tree results in the network. However, the concept of selfishness is selectively applied since the agents higher up in the data gathering tree have to commit to providing bandwidth for transmission of data until they are saturated. These agents are thus providing a service for no apparent gain. Rogers et al. [2005] correct this by designing a protocol in which parent nodes in the networks are incentivised to forward data by the payments provided by the child nodes. These payments are conditioned on the power that a sensor expends when forwarding data and the resulting protocol has a close to optimal performance. However, the designed payment protocol is based on the assumption of an inverse square power law governing the power expenditure (and hence is not generalisable to cases where this law is not obeyed) and is not robust to selfishness since it does not contain a way of guaranteeing that the parent node will conform to the protocol once it has been paid (such as in [Blankenburg et al., 2005]). The selfishness-related drawbacks of the systems discussed here are not major if the whole system is owned by a single stakeholder who can program each sensor to behave as it wishes. This is because the sensors will then conform to the designed protocol and selfishness is used as a means to achieve a decentralised allocation scheme. However, it also implies that these systems would fail if they were adopted in a context where the sensors are owned by different stakeholders and thus are selfish by nature rather than by design. This is because such sensors will be designed by each individual stakeholder that can take advantage of these drawbacks for their selfish gain, which will, in turn lead to a degradation of system performance.

In contrast, in [Dash et al., 2005] a centralised auction is designed for the allocation of data between self-interested sensors. In this protocol, the value the sensors place on data gathered from other sensors is dependent on their own private information, as well as that of the other sensors. A trusted centre computes the allocation and then provides the payments once the allocation is carried out. This protocol is discussed in greater depth in chapter 5.

Having thus explained the context of the research conducted within this thesis, we now detail the specific research contributions.

1.3 Research Contributions

The research reported in this thesis stems from our analysis and design of distributed allocation and distributed information mechanisms. This research thus provides the following insights into these two crucial aspects of distributed mechanisms:

- **Distributed Allocation:** Distributed allocation mechanisms require no centre to compute and enforce the overall allocation. In this context, we study two such mechanisms: simultaneous auctions (in Chapter 3) and the CDA (in Chapter 4). In both cases, we find that the efficiency of the distributed mechanism implemented is less than the full efficiency that can be achieved with centralised mechanism. Thus, we can infer that there is a cost of distributing a mechanism in that we can no longer achieve full efficiency. Nevertheless, we find that this cost is not overly prohibitive and in certain scenarios may be justified in order to gain the advantages of distributed mechanisms. Furthermore, the implementation of distributed allocation mechanisms for scenarios involving multiple goods and complex utility functions is not straightforward and thereby requires a significant design effort.
- **Distributed Information:** Distributed information mechanisms pertain to those situations where the agents require distributed information in order to formulate their preferences over outcomes. In such cases, we find that traditional mechanisms cannot incentivise the agents to choose strategies that lead to desirable outcomes. Therefore, we design novel efficient mechanisms (in Chapters 5 and 6) to deal with such distributed information.

Furthermore, in studying the computational properties of the centralised mechanisms we design in Chapters 4, 5 and 6, we find that often the computation of the solution is hard. This results from the fact that such mechanisms require the exact computation of solutions in order to guarantee their economic properties. Nevertheless, we reduce the computational load on the centre in each of these mechanisms by designing appropriate algorithms that exploit the problem structure.

Finally, we study the application scenario of a MSN composed of individually-owned sensors. This provides a canonical problem in which the specific requirements we outlined earlier are exhibited. We find that before designing mechanisms for these problems, it is important to properly formulate the specific goals that each individual sensor is trying to achieve. Moreover, in addressing the specific requirements we advance the state of the art in the following areas:

1. **Distributed Allocation.** In Chapter 3, we develop for the first time the optimal strategy of a buyer wishing to acquire a single unit in a simultaneous auction market consisting of a number of sellers each auctioning off a single identical item. This allows us to analyse the equilibrium behaviour when the buyers in the market are of three types: (i) global (such buyers can bid in all auctions), (ii) dynamic local (such buyers can only bid at one auction, but can randomly choose which auction to participate in) and (iii) static local (such buyers can only bid in one predetermined auction). We also prove that the lower bound on the efficiency of such markets is $1-\frac{1}{e}$. Furthermore, in Chapter 4, we develop a novel clearing scheme (employing a distributed allocation mechanism based on the CDA) so as to deal with constrained capacity suppliers. We empirically show that the efficiency of such a mechanism is fairly high (around 83%), even when employing very basic bidding and selling strategies [9].

2. **Limited Capacity.** We develop two new mechanisms for the case of multiple suppliers with limited capacities competing to satisfy a demand. The first mechanism is centralised and ensures the desirable properties of incentive compatibility, efficiency, individual rationality and robustness via the introduction of a novel penalty scheme. We provide an algorithm that computes the allocation in pseudo-polynomial time. The second mechanism is the distributed mechanism mentioned above and is based on the CDA protocol. This mechanism requires the design of a novel clearing scheme in order to address the issue of constrained capacities. Furthermore, this mechanism is fair in that it allows an approximately equal sharing of profits between buyers and sellers.

3. **Distributed (Interdependent) Information.** We show for the first time how to derive utility functions for a MSN scenario from information theory using a distributed information filter (which is a distributed way of measuring the information gain that a measurement provides). Furthermore, we develop a novel mechanism for the allocation of multiple goods (tasks) in the case when agents form their valuations from observations made by other agents (i.e. interdependent valuations). This mechanism is proven to be incentive compatible, efficient and individual rational.

4. **Uncertainty in Task Completion.** We develop a novel mechanism that accounts for the case in which an assigned allocation may not always be completed to the pre-specified level. Furthermore, different agents have different views about whether an allocation has been completed successfully. Thus, each agent has a measure of how well the other agents are likely to perform a particular task (which we term as *trust*). Hence, for the first time, we incorporate trust within the design of a mechanism. We then go on to study the economic properties of the mechanism and evaluate its performance against other closely-related mechanisms. We finally implement our mechanism using both linear and dynamic programming techniques that reduces the complexity of computing the optimal allocations and payments.

[9]The use of more sophisticated strategies generally leads to an increase in efficiency [Vytelingum et al., 2004]

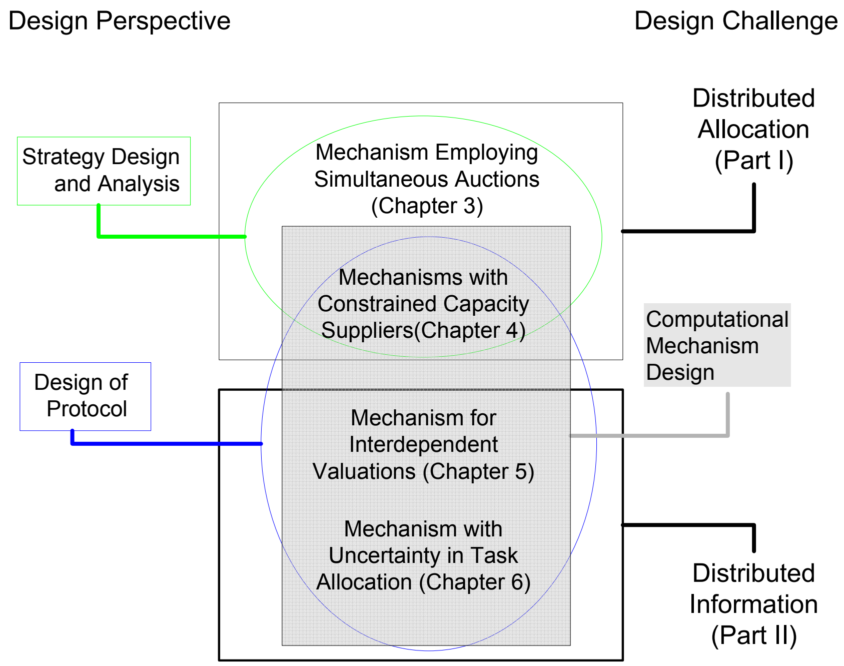


FIGURE 1.2: Positioning of work done in this thesis in relation to challenges involved in CMD and DMD.

To summarise, the work described in this thesis addresses a number of issues that arise when using MD for the design of MASs. In effect, our aim is to apply the theoretical work developed herein to MSN scenarios and simultaneously design distributed systems that address the challenges that have been outlined in the previous sections. This is illustrated diagrammatically by figure 1.2 which shows how the various strands of our work (which are detailed in the different chapters) are positioned according to whether they deal with distributed allocation or distributed information.

The work carried out in relation to this thesis has resulted in the publication of the following papers which are reported within this thesis:

- R. K. Dash, D. C. Parkes and N. R. Jennings (2003) "*Computational Mechanism Design: A Call to Arms*" IEEE Intelligent Systems 18 (6) 40-47. (Chapters 1 and 2)
- E. H. Gerding, R. K. Dash, D. C. K. Yuen and N. R. Jennings (2006). *Optimal Bidding Strategies for Simultaneous Vickrey Auctions with Perfect Substitutes*, Proc. of the 8th Int. Workshop on Game Theoretic and Decision Theoretic Agents, (AAMAS 06), 10-17. (Chapter 3)
- R. K. Dash, P. Vytelingum, A. Rogers, E. David and N. R. Jennings (2006) *Market-based task allocation mechanisms for limited capacity suppliers* IEEE Trans on SMC (Part A). (Chapter 4)

- R. K. Dash, A. Rogers, S. Reece, S. Roberts, and N. R. Jennings (2005) *Constrained bandwidth allocation in multi-sensor information fusion: a mechanism design approach* Proc. 8th Int. Conf. on Information Fusion, Philadelphia, USA. (Chapter 5)

- R. K. Dash, A. Rogers and N. R. Jennings (2004) *"A mechanism for multiple goods and interdependent valuations"* Proc. 6th Int. Workshop on Agent-Mediated E-Commerce, New York, USA, 197-210. (Chapter 5)

- A. Rogers, R. K. Dash, S. Reece, S. Roberts, and N. R. Jennings (2006) *Computational Mechanism Design for MultiSensor Information Fusion* Proc. 5th Int. Conf. on Autonomous Agents and Multi-Agent Systems (AAMAS 06), Hakodate, Japan. (Demo Paper)(Chapter 5)

- A. Rogers, R. K. Dash, N. R. Jennings, S. Reece and S. Roberts (2006) *Computational mechanism design for information fusion within sensor networks* Proc. 9th Int. Conf. on Information Fusion (Fusion 06), Florence, Italy. (Chapter 5)

- R. K. Dash, S. D. Ramchurn, and N. R. Jennings (2004) "Trust-based mechanism design" Proc. 3rd Int. Conf. on Autonomous Agents and Multi-Agent Systems (AAMAS04), New York, USA 748-755. (Chapter 6)

Furthermore, this work has also spawned a number of publications that are not been reported here (since they do not fit perfectly into the context of this thesis). Nevertheless, these papers relate to this thesis in the following ways:

- V. D. Dang, R. K. Dash, A. Rogers and N. R. Jennings (2006) Overlapping coalition formation for task distribution in multi-sensor networks Proc. 21st National Conference on AI (AAAI), Boston, USA.

  This paper investigates the use of cooperative coalition formation with sensor networks when sensors can belong to more than one coalition. It is related to the MSN scenario considered in this thesis and builds upon the use of an information theoretic base for utility functions (as in Chapter 5). However, in this paper, the sensors are cooperative and the focus is on devising algorithms that allow such a computationally complex task to be achieved.

- E. Gerding, A. Rogers, R. K. Dash and N. R. Jennings (2006) *Competing Sellers in Online Markets: Reserve Prices, Shill Bidding, and Auction Fees* Proc. 5th Int. Conf. on Autonomous Agents and Multi-Agent Systems (AAMAS 06), Hakodate, Japan, 1208-1210.

  This paper investigates how sellers can improve their revenue in a simultaneous auction environment using two common devices, namely reserve pricing (where they set a publicly known minimum price for sale) and shilling (where they anonymously collude with a bidder to set a minimum sale price). As such, this paper is related to the simultaneous auction environment studied within Chapter 3, though it concentrates on the seller side of the auction.

- P. Padhy, R. K. Dash, K. Martinez and N. R. Jennings (2006) *A utility-based sensing and communication model for a glacial sensor network* Proc. 5th Int. Conf. on Autonomous Agents and Multi-Agent Systems (AAMAS 06), Hakodate, Japan, 1353-1360.

  This paper investigates the use of utility functions within a MSN in order to maximise the data gathered by the sensors, whilst minimising the power consumed. It is related to the MSN scenario considered in this thesis though differs crucially in that it operates within a cooperative rather than a selfish context.

- P. Vytelingum, R. K. Dash, M. He, and N. R. Jennings (2005) A framework for designing strategies for trading agents Proc. Int Workshop on Trading Agent Design and Analysis, IJCAI 05, Edinburgh, Scotland, 7-13.

- P. Vytelingum, R. K. Dash, M. He, A. Sykulski and N. R. Jennings (2006) Trading strategies for markets: A design framework and its application Lecture Notes in Artificial Intelligence

  These papers propose a framework for designing strategies for trading agents that takes into consideration the various signals that these agents receive whilst trading. It has then been applied in order to design a strategy for the the trading agent competition [Wellman et al., 2004]. This research is related to Chapters 3 and 4, in that it concentrates on the design of strategies, though the technique employed has a heuristic base rather than the game-theoretic one used in Chapter 3.

- P. Vytelingum, R.K. Dash, E. David and N. R. Jennings (2004) ”A risk-based bidding strategy for continuous double auctions” Proc. 16th European Conference on Artificial Intelligence (ECAI 04), Valencia, Spain, 79-83.

  This paper investigates the use of risk with a bidding strategy for the CDA by changing the degree of aggressiveness of the strategy in relation to its value/cost and its prediction of the equilibrium price. The resultant strategy outperforms current strategies that have been proposed for the CDA and improves market efficiency. It is thus related to the research carried out in Chapter 4 on the CDA mechanism.

- B. Blankenburg, R. K. Dash, S. D. Ramchurn, M. Klusch, and N. R. Jennings (2005) Trusted kernel-based coalition formation Proc. 4th Int Joint Conf on Autonomous Agents and Multi-Agent Systems (AAMAS 05), Utrecht, Netherlands.

  This paper considers task allocation with uncertainty in task completion when using a cooperative game theoretic approach. As such, it is related to Chapter 6, though the solution concepts employed are based on coalitional games rather than MD.

- I. Rezek, S. J. Roberts, A. Rogers, R. K. Dash and N. R. Jennings (2005) Unifying learning in games and graphical models Proc. 8th Int. Conf. on Information Fusion, Philadelphia, USA.

  This paper looks at integrating fictitious play (which is a model of learning in games) with probabilistic graphical models. As such it is related to the underlying technique used within this thesis, namely game theory and views the field from a probabilistic point of view.

1.4 Thesis Structure

This section outlines the structure of the thesis giving a summary of the work presented in each chapter.

Chapter 2 discusses the theories relevant to our work, by reviewing the economic principles behind mechanism design. We discuss the possibility and impossibility results that relate to the different game theoretic equilibria, thereby outlining the implementable social choice functions (i.e functions that specify what the desirable system-wide properties are).

Chapters 3 and 4 consider the case of distributed allocation in which there is no trusted centre that collects all the bids and performs the required calculation for an allocation. However, they differ in the distributed protocol which they implement. Chapter 3 considers the case where there is no coordination mechanism available to the buyers and sellers. In contrast, Chapter 4 compares the CDA protocol (in which there is an indirect coordination mechanism in the form of the billboard posting current bids and asks) to a canonical centralised protocol (the Vickrey-Clarke-Groves Mechanism).

In Chapter 3, we first discuss a scenario in which buyer-sensors have to bid for tasks to be performed by seller-sensors. The scenario leads to a model whereby multiple sellers are selling identical items simultaneously, whereas buyers have to choose the bids they place at each of the sellers. We study the optimal strategies that the global bidders should employ when faced with different combinations of local and global bidders. In so doing, we prove that the global bidder should always place non-zero bids in all available auctions, irrespective of a local bidder's valuation distribution. We then study the computational problem of finding the optimal strategy and prove that, for non-decreasing valuation distributions, the problem of finding the optimal bids reduces to two dimensions. These results hold both in the case where the number of local bidders is known (i.e. static local bidders) and when this number is determined by a Poisson distribution (i.e. dynamic local bidders). In addition, by combining analytical and simulation results, we demonstrate that similar results hold in the case of several global bidders, provided that the market consists of both global and local bidders. Finally, we address the efficiency of the overall market, and show that information about the number of local bidders is an important determinant for the way in which a global bidder affects efficiency.

In Chapter 4 we discuss a scenario which concerns the supply of tasks by sellers in a market where the total demand exceeds the maximum that any of the individual sellers can supply. The sellers have a particular cost structure consisting of a fixed cost and a unit cost of production. We develop a modified centralised protocol in which we allow the sellers to communicate these defining characteristics of their cost function along with their capacity. We also show that the application of a penalty scheme is sufficient to ensure desirable economic properties of the mechanism. We also study the computational complexity of finding the best allocation using this protocol. We then analyse the CDA where both sellers and buyers participate in a market and thus the items that needs to be allocated are distributed over all agents in the systems, but every agent knows which items are being allocated and the status of the market. We study the economic properties of our modified CDA protocol by using very simple strategies and show that the performance of this protocol is satisfactory when compared to the centralised protocol.

In Chapters 5 and 6, we study another form of distribution, namely the distribution of information (which can be characterised as the interdependence of valuations). Chapter 5 develops a general mechanism for the case when there are multiple items in the market and the valuations of a buyer depend on its own observation, as well as signals observed by other buyers in the system. To this end, in Chapter 6, we look at a particular form of interdependent signals, namely trust, and develop a mechanism that incentivises the agents to report truthfully about their observed trust measure.

More specifically, in Chapter 5, we argue that interdependent valuations are common in MASs and then go on to develop a general mechanism that has desirable economic properties. We study its computational properties and show that the mechanism adds a computational load only on the centre (as compared to classic mechanisms). We also investigate an application for this mechanism which concerns a multi-sensor target detection scenario in which multiple individually-owned sensors are monitoring a particular area with each sensor having a particular accuracy of measurement. We model this as a MAS and propose a valuation function based around Information Theory that calculates the value each sensor has for information gained by other sensors.

In Chapter 6 we design a mechanism in which the uncertainty in the completion of a task is taken into account. We first investigate the case when each agent can report on its own uncertainty. We then analyse the more general case where each agent can report on other agents' uncertainties. Thus, we cannot hope to achieve a strong equilibrium (like in Chapter 5) and instead opt for a weaker equilibrium condition (ex-ante Nash equilibrium). We analyse the properties of this mechanism and benchmark it against other comparable mechanisms. We study the computational properties of our mechanism and implement it using both linear and dynamic programming techniques that reduce the amount of computation required for finding the optimal allocations and payments by reducing the size of the search space and reusing past solutions.

Finally in Chapter 7, we summarize the main achievements of this thesis and how well they satisfy the requirements discussed in this chapter. We also discuss the broad future research directions that have been identified for the fields of DMD and its application within MSNs.

Chapter 2

Mechanism Design

Mechanism Design (MD) is the area of micro-economics concerned with how to design systems, using tools developed by game theory analysis, such that certain system-wide properties emerge from the interaction of the constituent components. As such it provides the basis on which a large part of this thesis rests. We therefore provide in this section a brief mathematical outline of the rich and important body of research to which MD has given rise (see [Jackson, 2000; Osborne and Rubinstein, 1994; MasColell et al., 1995; Krishna, 2002; Klemperer, 2002] for more comprehensive reviews.). This chapter thus presents traditional MD, which has concerned itself with how to satisfy certain economic criteria (such as efficiently allocating resources, maximising revenues or having a fair system) given the setting of selfish agents in interactive decision making. The newer challenges within mechanism design, namely computational and distribution challenges have been reviewed in Chapter 1 and the related work specific to each chapter is discussed more extensively in each of the chapters.

In more detail, we will first present a basic model of a mechanism in Section 2.1 and explain how different solution concepts may arise in mechanisms in Section 2.2. We then present some of the social choice functions that traditional MD has concerned itself with in Section 2.3. We explain what can and what cannot be achieved in Section 2.4. Within this section, we present an important mechanism, namely the Vickrey-Clarke-Groves (VCG) mechanism, which we refer to extensively within this thesis.

2.1 Mechanisms

A mechanism (or game form) $\Gamma = (\mathcal{I}, \Theta, S, g(.))$ consists of a set of agents $\mathcal{I} = \{1 \ldots, I\}$ that each have a strategy set S_i. Each agent chooses its strategy $s_i \in S_i$ from its particular strategy set given the private information contained in its types $\theta_i \in \Theta_i$ and an outcome function, $g : S_1 \times \cdots \times S_I \to \mathcal{O}$, which sets the outcome. The way that each agent chooses its strategy depends on how we model that agent. For example, it is commonly assumed that agents choose their strategy so as to achieve their best outcome. The "goodness" of the outcome $o \in \mathcal{O}$ is measured by a utility function [Varian, 1999] that gives a numerical value to each outcome (higher being better) with $u_i : \mathcal{O} \times \Theta_i \rightarrow \Re$. Thus the assumption of each agent looking for its best outcome, also known as the rationality assumption [Rubinstein, 2002], can be stated in terms of the utility function as:

Assumption 2.1. **Rationality Assumption.** An agent is termed *rational* if it chooses its best strategy, s_i^*

$$s_i^* = \arg \max_{s_i \in \Sigma_i} E[u_i(g(s_1, \ldots, s_I))] \tag{2.1}$$

where $E[.]$ is the expectation operator.

Given this setting, the mechanism is then designed so as to satisfy certain criteria which are encompassed in the social choice function (SCF) $scf : \Theta_1 \times \cdots \times \Theta_I \rightarrow \mathcal{O}$. We hence have $scf(\theta)$ being the outcome that satisfies the particular criteria set by the designer. Since the aim of the mechanism is to achieve $scf(\theta)$, we can restate the objective of the designer as being:

$$g(s_1, \ldots, s_I) = scf(\theta) \tag{2.2}$$

We say that a mechanism Γ *implements* $scf(\theta)$ whenever equation 2.2 is satisfied. The central question in MD asks which set of desiderata (or which $scf(\theta)$) can and cannot be achieved under a certain solution concept (which is a state that can be predicted to occur given a certain mechanism). This question is partially answered in terms of results, termed *possibility* and *impossibility* theories, which are discussed in greater detail in section 2.4 (after a discussion on the solution concepts and desirable desiderata set). One obvious way of trying to achieve $scf(\theta)$ is to ask the agents to report their types truthfully such that $s_i = \theta_i$ and then set $g(.) = scf(.)$. Such mechanisms, in which $S_i = \Theta_i$, are called *direct mechanisms* (a.k.a *direct revelation* mechanisms):

Definition 2.1. Direct Mechanism. A direct mechanism is one in which the strategy space, S_i, available to each agent is reporting its type Θ_i.

However, a straightforward implementation of a direct mechanism does not guarantee that the agents would communicate the true values, θ. In order for this to happen, we need to build into the mechanism the incentives for the agents to reveal their types truthfully. If any agent i finds that $u_i(g(\hat{\theta}_i, \theta_{-i})) \geq u_i(g(\theta_i, \theta_{-i}))$ where the reported type $\hat{\theta}_i \neq \theta_i$, then it has an incentive to report $\hat{\theta}_i$. Hence, we require $u_i(g(\hat{\theta}_i, \theta_{-i})) < u_i(g(\theta_i, \theta_{-i}))$ for all $i \in \mathcal{I}$ to ensure that each agent reports its true type. Mechanisms in which this occurs are called *incentive compatible* mechanisms:

Definition 2.2. Incentive Compatible Mechanism. In an incentive compatible mechanism, each agent $i \in \mathcal{I}$ has an incentive to tell the truth about its type; that is, $u_i(g(\theta_i, \theta_{-i})) > u_i(g(\hat{\theta}_i, \theta_{-i})), \ \forall \ \hat{\theta}_i \in \Theta_i, \ \ \hat{\theta}_i \neq \theta_i$.

Thus we can see that we can implement the desired $scf(.)$ by selecting a direct mechanism in which $g(.) = scf(.)$ and setting the outcome function such that the mechanism is incentive compatible. This insight is commonly termed the *revelation principle*:

Theorem 2.3. ***Revelation Principle.*** *The revelation principle states that if a mechanism* $\Gamma = (\mathcal{I}, \Theta, S, g(.))$ *implements* $scf(.)$*, then there exists an incentive-compatible direct revelation (ICDR) mechanism,* $\Gamma = (\mathcal{I}, \Theta, scf(.))$ *that implements* $scf(.)$.

The revelation principle is a powerful tool for analysis. It enables attention to be restricted to the class of ICDR mechanisms in the derivation of mechanisms that are possible/impossible to implement. These mechanisms are easier to analyse since we only need to consider the case where agents have the restricted strategy space of just revealing their types. However, it should be borne in mind that the revelation principle does not imply that we only ever need to consider ICDR mechanisms. This is because these mechanisms may not have the desired computational properties. In short, the revelation principle is of prime importance within MD because of two main considerations:

1. *Theoretical.* It concentrates attention on incentive-compatible direct mechanisms for the development of impossibility and possibility results.

2. *Practical.* A designer can characterise the $scf(.)$ that needs to be satisfied in ICDR mechanisms. Then, this can be used to provide a *normative* guide for the outcome and payments that must be computed in a realised implementation, that need not itself be an ICDR mechanism. Hence, the revelation principle provides a necessary (but not sufficient) condition for the existence of distributed mechanisms and hence provides a good basis for DMD.

It is important to emphasise what the revelation principle does *not* provide. First, to reiterate the last point on practicality, it provides normative goals for mechanism design, but does not imply that the only mechanisms that are interesting in practice are direct-revelation mechanisms. As we discussed in chapter 1, there are a whole host of reasons why centralised direct-revelation mechanisms may be problematic from a computational perspective. Second, the normative goals are only relevant when agents are actually able to play the equilibrium strategies assumed in mechanism design. This assumption may itself not be reasonable with computationally-bounded agents.

To illustrate the points made in this section, we consider the example of the single-item English auction (in which agents can bid at the current ask price or leave the auction and the price increases by some minimal bid increment $\epsilon > 0$ until only one agent is left). Using the revelation principle, it is known that this non-ICDR mechanism[1] can be transformed into an equivalent ICDR mechanism, in this case a *Vickrey* auction (a sealed-bid action where the winner is the

[1]This is non-ICDR since agents only reveal their private value incrementally, thereby giving only the information that their value is higher (or not) than the current bid.

highest bidder but who pays the second highest bid) [Krishna, 2002]. In more detail, the English auction ensures that the item is allocated to the agent having the highest valuation of the item (an economic desideratum termed *allocative efficiency*). The Vickrey auction also has the same property, but it is an ICDR mechanism since the agents reveal their types (in this case the valuation of the items) and it is incentive-compatible (the agents can be proven to have highest utility when revealing their types truthfully). Notice, however, that the auctions are conducted in a different manner; the most salient difference being that the English auction is iterative in that bidders incrementally increase their bids, while in the Vickrey auction there is a single submission of bids. This demonstrates how a direct mechanism can have an indirect counterpart that satisfies the same theoretical goals, but which has very different practical implementations.

Having discussed the ICDR mechanisms, we now need to study the solution concepts. The role of these concepts is to indicate which particular action or strategy a rational agent would employ under the mechanism we are designing. This, in turn, allows an analysis and prediction of the mechanisms that is being designed.

2.2 Solution Concepts

Mechanism design requires a solution concept to predict the strategies the agents will select in various circumstances. Knowing these strategies will, in turn, ensure that the properties of a particular mechanism can be predicted. Ostensibly, a mechanism may implement $scf(\theta)$ under a wide variety of solution concepts, of which we only provide a few of the most important ones here (see [MasColell et al., 1995] for a more in-depth study). As stated in chapter 1, it is up to the designer to select the appropriate solution concept which is achieved by setting S and $g(s)$. These design parameters, along with the design environment, will lead to different kinds of solutions arising. For analysis, we can partition games into cooperative and non-cooperative games.

In this thesis we focus on competitive game theory purely because it has been the more researched field in terms of mechanism design and is more applicable to the situations which we wish to study (see Chapter 1). We present the three most important solution concepts in competitive game theory below (see [Osborne and Rubinstein, 1994] for a fuller treatment). Each of the solution concepts presented require stronger assumptions about agents and are, therefore, a weaker implementation concept (i.e. the confidence with which the equilibrium can be predicted is weaker or the environment in which the implementation is carried out is more restrictive). Nevertheless, all these solution concepts are based around the notion of a *best-response* strategy, which is the best strategy to play given the (expected) actions of other agents. These solution concepts relate to *strategic games* (a.k.a normal form games).

Definition 2.4. Strategic Game. A *strategic game* is one where each agent $i \in \mathcal{I}$ chooses its strategy $s_i \in S_i$ based on its preferences or type $\theta_i \in \Theta_i$ which then leads to an outcome $o \in \mathcal{O}$

determined by the outcome function $g(.)$. Thus a strategic game is completely defined by the tuple $\Gamma = (\mathcal{I}, \Theta, S, g(.))$

Thus a strategic game is a one-shot game. The agents choose their actions and the outcome function determines the outcome. To this end, within stategic games we define the following three equilibrium solutions:

Definition 2.5. Dominant Strategy. In a *dominant strategy* equilibrium each agent has a best-response strategy no matter what strategy is selected by the other agents. Formally, we have:

$$s_i^*(\theta_i) = \arg\max_{s_i} u_i(\theta_i, g(s_i(\theta_i), s_{-i}(\theta_{-i}))), \quad \forall s_{-i}, \forall \theta_{-i} \tag{2.3}$$

for all $\theta_i \in \Theta_i$.

Definition 2.6. *ex post* Nash. Each agent's strategy is a best-response to the strategy of other agents, no matter what their types, as long as they also play an equilibrium strategy:

$$s_i^*(\theta_i) = \arg\max_{s_i} u_i(\theta_i, g(s_i(\theta_i), s_{-i}^*(\theta_{-i}))), \quad \forall \theta_{-i} \tag{2.4}$$

for all $\theta_i \in \Theta_i$, where $s_{-i}^*(\theta_{-i})$ denotes the strategies selected by other agents.

Definition 2.7. Bayesian-Nash. Each agent selects a best-response strategy to maximise its expected utility given its beliefs about the distribution over types:

$$s_i^*(\theta_i) = \arg\max_{s_i} E_{\theta_{-i}}[u_i(\theta_i, g(s_i(\theta_i), s_{-i}^*(\theta_{-i})))] \tag{2.5}$$

for all $\theta_i \in \Theta_i$.

A dominant strategy equilibrium is a very robust solution concept because an agent does not need to form beliefs either about the rationality of other agents or about the distribution over the types of other agents. An example of a dominant strategy implementation is the Vickrey auction. In this auction, the best strategy for an agent is to bid truthfully. This is irrespective of what the other agents bid.

An *ex post* Nash equilibrium requires common knowledge about the rationality of agents, but does not require any knowledge about type distributions. In this sense, *ex post* Nash has a no-regret property such that an agent does not want to deviate from its strategy even once it knows the strategies and types of other agents. As a simple example, a straightforward bidding strategy in which an agent bids in each round of an ascending-price Vickrey auction [2] to maximise its utility is an *ex post* Nash equilibrium [Gul and Stacchetti, 2000; Parkes, 2001].[3]

[2]This is a modified Vickrey auction where now the auction is conducted over multiple rounds. At each round, the results of the previous round are known and the auction ends when there is no change in results over two rounds.

[3]This is not a dominant strategy in this relaxed auction because another agent might condition a "crazy strategy" such as "I will bid to $1 million" on the price hitting a particular target value. In this case an agent that would otherwise win should submit a jump bid past this target value.

The weakest solution concept adopted in mechanism design is the *Bayesian-Nash* equilibrium (BNE). In a BNE an agent must both hold beliefs about the rationality of other agents, and also correct beliefs about the distribution on types of other agents. The first-price sealed bid auction is a classic example with a simple Bayesian-Nash equilibrium. For example, given a symmetric distribution of agent types with values that are identically and independently distributed $v_i \sim U(0, 1)$ the symmetric BNE is for agents to play $s_i^*(v_i) = (|\mathcal{I}| - 1)v_i/|\mathcal{I}|$.

Given these solution concepts, we now focus on what properties we want to emerge from the mechanism when a solution has been reached.

2.3 Implementation of Social Choice Functions

Social choice functions are functions that are used to describe the outcomes in a game. As designers we seek to implement SCFs with desirable properties, and in the strongest-possible equilibrium solution concept (because this then guarantees that the properties will be achieved). However, we will see that there are often properties that cannot be implemented in any mechanism, even in a Bayesian-Nash equilibrium.

Typical desiderata in SCFs include [MasColell et al., 1995]:

Pareto optimality: SCF $scf(\cdot)$ chooses an outcome o^* such that there is no other outcome $o' \in \mathcal{O}$ in which one agent is better off without one of the others being worse-off. (i.e. at o^*, if $u_i(\theta_i, o') > u_i(\theta_i, o^*)$ for some i then $u_j(\theta_j, o') < u_j(\theta_j, o^*)$ for some $j \neq i$.)

Maximise social welfare: SCF $scf(\cdot)$ chooses an outcome o^* to maximise the total utility of agents, i.e. $scf(\theta) = \max_{o \in \mathcal{O}}[\sum_{i \in N} u_i(\theta_i, o)]$. Social-welfare maximising outcomes are always Pareto optimal, and are sometimes called the *efficient* outcome.

Maximise utility of centre: Maximise the expected utility to the auctioneer across all possible mechanisms. Here, we consider outcomes that decompose into an allocation and agent payments, and select a SCF $scf(\cdot)$ that maximises the expected value of the centre for any goods not sold and the expected payment received by the centre.

Another constraint that is often placed on SCFs that involve payments is that of *budget-balance*, such that the total payment made by agents should exactly equal zero (so that money is neither injected into a system nor removed from it). This property is especially important in systems that must be *self-sustaining* and require no external benefactor to input money into the system, or a central authority that collects payments.

Finally, a designer of an open system should provide an incentive for agents to join the system. Such *individually-rational* mechanisms ensure the agent perceives a greater interest in joining the mechanism rather than remaining outside it.

We now have the ingredients to construct mechanisms that satisfy particular desiderata. However, as we shall see in the next section, there are some fundamental constraints on the mechanisms we can build to satisfy a certain set of desiderata.

2.4 Important Impossibility and Possibility Theorems

In this section, we briefly cover some of the key impossibility results (results that prove the impossibility of implementing certain SCFs under certain solution concepts) and possibility results (which define mechanisms in which the environment and solution concept are chosen such that certain SCFs are always satisfied). We have already expressed an avowed desire to implement mechanisms in dominant strategies in section 2.2 due to the strength of the solution concept. However, an important negative result, the Gibbard-Satterthwaite impossibility theorem [Gibbard, 1973], states that it might be impossible to do so. Before stating the theorem, we shall first define a *dictatorial* SCF:

Definition 2.8. Dictatorial. A SCF $scf(.)$ is dictatorial if $\exists i: \quad \forall\theta = \{\theta_1, \ldots, \theta_I\} \in \Theta$, $scf(\theta) \in \{o \in \mathcal{O} : u_i(o, \theta_i) \geq u_i(o', \theta_i) \;\; \forall o' \in \mathcal{O}\}$

Thus, in a dictatorial SCF, the outcome is *always* the chosen outcome of only one agent. The Gibbard-Satterthwaite theorem is then:

Theorem 2.9. ***Gibbard-Satterthwaite Impossibility Theorem.*** *In a setting consisting of*

- *agents with general utility functions (i.e they are derived from complete, transitive and strict preferences),*
- *a finite set of outcomes $\mathcal{O}$, with more than 3 possible outcomes (i.e. $|\mathcal{O}| \geq 3$)*

any SCF which is incentive-compatible in dominant strategies is dictatorial.

The above theorem is quite negative, because coupled with the revelation principle, it implies that we cannot implement *any* mechanism based on dominant strategy given the (quite general) conditions in these settings. As a result, even pareto-optimality, one of the most basic desideratum, cannot be satisfied. One way to circumvent this impossibility result is to restrict the utility functions of the agents and the environment in which they are operating. The restriction most commonly applied is a simple *exchange* environment (one in which goods are not produced but only exchanged) in which the agents are assumed to have a *quasi-linear* utility function. In order to define a quasi-linear utility function, we first decompose the outcome into two parts. Let $o = (k, t)$ denote the outcome with $k \in \mathcal{K}$ defining the allocation in the space of possible allocations $\mathcal{K}$ and let $t = (t_1, \ldots, t_N)$ be the transfer of money among agents.

Definition 2.10. Quasi-linear Utility Function. A quasi-linear utility function is one which can be expressed as:

$$u_i(k, t_i, \theta_i) = v_i(k, \theta_i) + t_i \tag{2.6}$$

where $v_i(k, \theta_i)$ is the value of allocation k to agent i given its type θ_i.

Thus an agent with a quasi-linear utility function does not differentiate between two outcomes, one in which there is an allocation k with no transfers of money and another one in which there is no allocation and it is being paid its value of the allocation, $v_i(k, \theta_i)$.

We will now present two mechanisms – the Vickrey-Clarke-Groves (VCG) and d'Aspremont-Gerard-Varet (dAGVA) in the next section that achieve different sets of the SCF. The mechanisms are similar in that both achieve incentive-compatibility and efficiency. However, whilst the VCG mechanism is individually-rational but not budget-balanced, the dAGVA mechanism is budget-balanced but not individually-rational.

2.4.1 Direct Mechanisms

With the restriction of quasi-linearity, we then have a family of direct mechanisms, termed Vickrey-Clarke-Groves mechanisms, that implement an efficient and individually-rational SCF where truth-telling is a dominant strategy [Groves, 1973; Vickrey, 1961; Clarke, 1971; Mas-Colell et al., 1995]. These mechanisms form the basis of much of the work presented in this thesis.

The VCG mechanism has an outcome function specified by an allocation rule $\mathcal{M}$ and a payment function $\mathbf{r}$[4]. A typical forward VCG auction proceeds as follows[5] :

1. The auctioneer posts the set of items M it wishes to sell.
2. Each agent i then reports its valuation function $v_i(K, \theta_i)$ for every possible allocation $K \in \mathcal{K}$. $\mathcal{K}$ is the set of all possible sets of the items in M.
3. Each agent i also reports its type $\widehat{\theta}_i$.
4. The centre then solves the following equation to find the efficient allocation:

$$\widehat{K}^* = \arg\max_{K \in \mathcal{K}} \sum_{i \in \mathcal{I}} v_i(K, \widehat{\theta}_i) \tag{2.7}$$

[4] Though a complete mechanism is defined by the tuple $< \mathcal{I}, \Theta, S, g(.) >$, we will in the case of auctions refer to the outcome function $g(.)$ (which is defined by the tuple $< \mathcal{M}, \mathbf{r} >$) as the mechanism. This is because auctions are direct mechanisms whereby the strategy space S is the same as the type space Θ.

[5] A forward auction is one in which the auctioneer sells items and receives bids for them, whereas a reverse auction is one in which the auctioneer is buying items and receives asks for them.

5. It also computes each transfer r_i from each agent as:

$$r_i = \left[\max_{K \in \mathcal{K}} \sum_{j \in -i} v_j(K, \widehat{\theta}_j) \right] - \left[\sum_{j \in -i} v_j(\widehat{K}^*, \widehat{\theta}_j) \right] \text{ where } -i \equiv \mathcal{I} \setminus i \tag{2.8}$$

The VCG mechanism is *strategyproof* i.e. it is incentive-compatible under dominant strategies [MasColell et al., 1995]. It achieves its strategyproofness via its payment scheme which aligns the utility of the agent with the agent's marginal contribution to the mechanism. In fact, in a VCG mechanism, it can be observed that if an allocation $\widehat{K}^*$ is implemented, then the agent derives a utility of:

$$\begin{aligned} u_i(K, \theta_i) &= v_i(\widehat{K}^*, \theta_i) - r_i(\widehat{K}^*, \widehat{\theta}_i) \\ &= v_i(\widehat{K}^*, \theta_i) + \left[\sum_{j \in -i} v_j(\widehat{K}^*, \widehat{\theta}_j) \right] - \left[\max_{K \in \mathcal{K}} \sum_{j \in -i} v_j(K, \widehat{\theta}_j) \right] \end{aligned} \tag{2.9}$$

Thus, agent i can only manipulate (and try to maximise) the first two terms of equation 2.9. We also note that this is the same maximisation that is precisely done by the centre in equation 2.7. Thus, no matter what the other agents report, agent i can do no better than report its valuation truthfully, thereby leading to the mechanism being strategyproof. Furthermore, since the marginal contribution of an agent can only ever be non-negative, it can also be deduced that the VCG is also individually-rational. Finally, given that all agents report truthfully, then the mechanism implements the efficient allocation K^* which satisfies:

$$K^* = \arg\max_{K \in \mathcal{K}} \sum_{i \in \mathcal{I}} v_i(K, \theta_i) \tag{2.10}$$

However the VCG mechanism is not budget balanced. It often runs at a budget deficit although in an auction setting, the mechanism will run at a surplus to the auctioneer. Budget balance is an important criteria, for example, in the generalised setting of exchanges with multiple buyers and sellers and a mechanism serving as an intermediary. Even within these settings, the VCG mechanism is not budget-balanced and will run at a deficit [Parkes et al., 2001]. In fact, the Hurwicz impossibility theorem [Hurwicz] tells us it is futile to search for an incentive-compatible mechanism implementing efficient, budget balanced SCF in dominant strategies:

Theorem 2.11. ***Hurwicz Impossibility Theorem.*** *There does not exist any incentive-compatible mechanism that implements a SCF that is efficient and budget-balanced in dominant strategy equilibrium, even with quasi-linear preferences.*

There are then two ways around this problem. We can clear exchanges sub-optimally to explicitly sacrifice some efficiency in return for budget-balance [McAfee, 1992; Parkes et al., 2001].

Alternatively, in order to be able to achieve budget-balance and efficiency, we can use a weaker implementation concept (namely Bayesian-Nash equilibrium). Under this solution concept, it is then possible to use the d'Aspremont-Gerard-Varet-Arrow (dAGVA) mechanism [d'Aspremont and Gerard-Varet, 1979; Arrow] so as to achieve both budget balance, efficiency and incentive compatibility. Though we do not use the dAGVA in this thesis, we intend to use it in future work due to the fact that it is also budget-balanced. We thus present it here for completeness. We also believe that in certain systems, this requirement may be quite important and thus justify the use of mechanisms derived from the dAGVA despite the weaker solution concept.

The dAGVA mechanism, also known as an "expected form" Groves mechanism, achieves individual rationality, efficiency and budget-balance under Bayesian-Nash equilibrium. It consists of an allocation rule which is the same as in the VCG mechanism but differs crucially in its payment scheme (though the structure is quite similar). The "expected form" arises in this case because the centre forms an expectation over the types of the agents in $-i$ when calculating the impact of agent i. In more detail, the dAGVA mechanism proceeds as follows:

1. The auctioneer posts the set of items M it wishes to sell.
2. Each agent i then reports its valuation function $v_i(K, \theta_i)$ for every possible allocation $K \in \mathcal{K}$
3. Each agent i also reports its type $\widehat{\theta}_i$
4. The centre then solves the following equation to find the efficient allocation:
$$\widehat{K}^* = \arg\max_{K \in \mathcal{K}} \sum_{i \in \mathcal{I}} v_i(K, \widehat{\theta}_i) \tag{2.11}$$
5. It also computes each transfer r_i from each agent as:
$$r_i = x_i(\widehat{\theta}_{-i}) - E_{\theta_{-i}} \left[\max_{K \in \mathcal{K}} \sum_{j \in -i} v_j(K(\widehat{\theta}_i, \theta_{-i}), \theta_j) \right] \tag{2.12}$$

The dAGVA mechanism thus preserves incentive-compatibility since in this case the *expected* utility that an agent derives is then:

$$\begin{aligned} u_i(K, \widehat{\theta}_i) &= E_{\theta_{-i}} \left[\max_{K \in \mathcal{K}} v_i(K(\widehat{\theta}_i, \theta_{-i}), \theta_i) \right] - r_i(K, \theta_i) \\ &= E_{\theta_{-i}} \left[\max_{K \in \mathcal{K}} \sum_{i \in \mathcal{I}} v_i(K(\widehat{\theta}_i, \theta_{-i}), \theta_i) \right] - x_i(\widehat{\theta}_{-i}) \end{aligned} \tag{2.13}$$

Thus, we again have that the agent can only control the first part of the transfer in equation 2.13, which it maximises when submitting $\widehat{\theta}_i = \theta_i$.

Eff	B.B	IR	Solution Concept	Possible & **Impossible**	Environment
No			Dominant	**Gibbard - Satterthwaite**	General
	No				
		No			
Yes	No	Yes	Dominant	VCG	Quasi-linear utility Exchange Env
No	No		Dominant	**Hurwicz**	Exchange Env
Yes	Yes	No	Bayesian Nash	dAGVA	Quasi-linear utility Exchange Env
No	No	No	Bayesian Nash	**Myerson Satterthwaite**	Quasi-linear utility Exchange Env

TABLE 2.1: Table showing possibility and impossibility results. The first three columns show the SCFs that can/cannot be achieved in tandem

The dAGVA mechanism can also achieve budget-balance by careful selection of the function $x_i(\widehat{\theta}_{-i})$. In effect, for budget-balance, we require that $\sum_{i \in \mathcal{I}} r_i = 0$ which implies that:

$$\sum_{i \in \mathcal{I}} \left(x_i(\widehat{\theta}_{-i}) - E_{\theta_{-i}} \left[\max_{K \in \mathcal{K}} \sum_{j \in -i} v_j(K(\widehat{\theta}_i, \theta_{-i}), \theta_j) \right] \right) = 0$$

Thus, any $x_i(\widehat{\theta}_{-i})$ satisfying the above, would lead to a budget-balanced mechanism. One possible form is that $x_i(\widehat{\theta}_{-i})$ is the average of the negative part of the transfer of all the other agents (see equation 2.12):

$$x_i(\widehat{\theta}_{-i}) = \frac{1}{|\mathcal{I}| - 1} \sum_{j \in -i} E_{\theta_{-j}} \left[\max_{K \in \mathcal{K}} \sum_{j \in -i} v_j(K(\widehat{\theta}_j, \theta_{-j}), \theta_j) \right]$$

However, achieving incentive-compatibility, budget-balance, efficiency and individual rationality in the dAGVA mechanism is impossible due to the Myerson-Satterthwaite Impossibility Theorem [Myerson and Satterthwaite, 1983].

Theorem 2.12. ***Myerson-Satterthwaite Impossibility Theorem.*** *There does not exist any mechanism that implements a SCF that is efficient, budget-balanced and individually rational in Bayesian-Nash strategy equilibrium, even with quasi-linear preferences.*

Hence, in the context of designing systems for MAS, we can either use the VCG mechanism in order to achieve efficiency, incentive-compatibility and individual rationality under dominant strategies or opt for the weaker solution concept of Bayesian-Nash equilibrium so as to achieve budget-balance as well while sacrificing individual rationality. Table 2.1 presents a summary of the results discussed in this section.

2.5 Summary

In this section we have given a brief overview of the economic principles involved in mechanism design. We provided a generic model of a mechanism and discussed some of the different solution concepts under which a mechanism may be implemented. We then studied a number of desiderata which we might wish a SCF to be endowed with while explaining which particular set can or cannot be achieved under the respective solution concepts. While discussing the theories in this chapter, we have implicitly made three assumptions (which are also common to most work in these areas):

1. There is always a trusted centre that can gather the necessary data from the agents nd can compute and enforce the outcome.
2. In a reverse auction, an agent has the capacity to fulfill the demand required by the auctioneer.
3. An agent's valuation or cost is derived from a private observation of its type only.
4. Once an agent has been allocated a task in a reverse auction, it will complete it to the predefined specifications which have been agreed with the allocator.

However, as we argued chapter 1, these assumptions do not always hold in MASs. Thus,in the next four chapters, we deal with the challenges posed by removing each assumption. Specifically, we study the case where there is no trusted centre in Chapter 3 by analysing a simultaneous auction scenario. We then remove the assumption of unconstrained capacity in Chapter 4 and design a centralised mechanism with desirable SCFs for this case. Within this chapter, we also design a distributed mechanism and compare its performance with that of the centralised one. We then address the third assumption by designing a mechanism for multiple goods and interdependent valuations in Chapter 5. Finally, we remove the last assumption in Chapter 6 by considering agents that have a certain failure rate and we go on to design a mechanism with desirable SCFs for this case.

Part I

Distributed Allocation Mechanisms

The first part of this thesis will consider issues associated with distributed allocation mechanisms. This is a core challenge within distributed mechanisms (as highlighted in red in figure I.1). Specifically, this challenge considers how to design mechanisms when there is no trusted centre who collects data from all the agents and determines the allocation of resources and payments within the system.

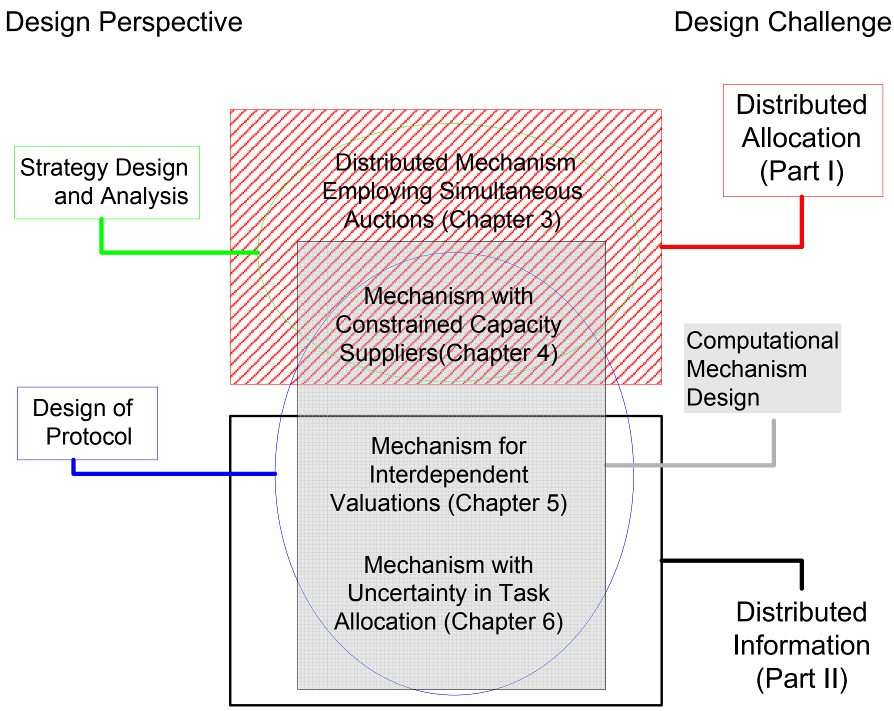


FIGURE I.1: The challenges addressed and the design perspective of part I of the thesis

Within distributed allocation mechanisms, the allocation of resources and payments must be determined via the interactions of each agent rather than at a central point. Such mechanisms are very attractive for sensor networks since they have the advantages of tractability, robustness, trustworthiness and reduction of bottlenecks (see Chapter 1 for a more detailed discussion). Now, within a cooperative setting, distributed task allocation has been extensively studied [Lesser and Corkill, 1981; Jennings and Bussmann, 2003; Pynadath and Tambe, 2003; Kraus et al., 1998]. However, the implementation of these mechanisms remain a challenge when considering selfish agents since these agents act to maximise their own utilities and therefore would not collaborate unless there is an incentive to do so. As a result, the distributed allocation mechanisms we study in this thesis all show a certain loss of efficiency when compared to their centralised counterparts.

In more detail, Chapter 3 reports on the optimal strategies that should be adopted by agents within a simultaneous auction environment. Here the distributed allocation occurs since each of the seller agents independently determine which buyer agent will be allocated their service. We then analyse another distributed mechanism based on the CDA in Chapter 4 whilst considering constrained capacity suppliers. In this case, the distributed allocation emerges out of the interactions between buyers and sellers. In order to benchmark the distributed mechanism, we design a centralised protocol for this scenario.

Chapter 3

A Mechanism Employing Simultaneous Auctions

In this chapter, we address requirement 1 of this thesis (as detailed in chapter 1), by studying a distributed allocation mechanism. We do so by analysing a market in which the goods are auctioned concurrently by a number of sellers, rather than by a single centralised auctioneer. Thus, the allocation of the goods is not computed by a centre, but rather is determined by the behaviour of the buyers in each of the parallel auctions. This therefore results in a distributed allocation mechanism whose properties we study in this chapter. Furthermore, we choose these simultaneous auctions, since they provide us with a baseline performance for distributed allocation mechanisms in which the agents are selfish. This is because neither the sellers nor the buyers can coordinate in order to set the price of an item (unlike in the CDA where this occurs indirectly via a billboard).

In order to study the distributed allocation mechanism, we first need to design and analyse the optimal strategy for a bidder (assumed to be rational) in such a market. We can then investigate an important global property of this distributed market, namely its efficiency, contingent upon this strategy. Now, the optimal strategy for a bidder is dependent on the type of competing bidders it faces and the amount of knowledge it has about the market (as we shall see later on in this chapter). Furthermore, the efficiency of the market depends on the type of bidders that participate in these markets.

The remainder of this chapter is structured as follows. Section 3.1 places this research in the global context of MASs and details the advances we make to the state of the art in this area. In Section 3.2, we describe the MSN scenario in which such distributed auctions occur. We then discuss the related work in the field of simultaneous auction in Section 3.3. In Section 3.4 we describe the bidders and the auctions in more detail. In Section 3.5 we investigate the case with a single global bidder and characterise the optimal bidding behaviour for it. Section 3.6 considers the case with multiple global bidders and in Section 3.7 we address the market efficiency and the impact of a global bidder. Section 3.8 concludes and discusses future work.

3.1 Introduction

In recent years, there has been a surge in the application of auctions, both online and within multi-agent systems [Airiau and Sen, 2003; Clearwater, 1996; Gerding et al., 2006b; Dash et al., 2005; Rosenthal and Wang, 1996; Roth and Ockenfels, 2002]. As a result, there are an increasing number of auctions offering very similar or even identical goods and services. In eBay alone, for example, there are often hundreds or sometimes even thousands of concurrent auctions running worldwide selling such substitutable items[1]. Against this background, it is important to develop bidding strategies that agents can use to operate effectively across a wide number of auctions. To this end, in this chapter we devise and analyse optimal bidding strategies for a bidder that participates in multiple simultaneous auctions for goods that are perfect substitutes.

To date, much of the existing literature on simultaneous auctions focuses either on complementarities, where the value of items together is greater than the sum of the individual items, or on heuristic strategies for simultaneous auctions (see Section 3.3 for more details). In contrast, here we consider bidding strategies analytically and for the case of perfect substitutes. In particular, our focus is on simultaneous Vickrey or second-price sealed bid auctions. We choose these because they are communication efficient (since they are direct mechanisms as defined in Chapter 2) and well known for their capacity to induce truthful bidding [Krishna, 2002], which makes them suitable for many multi-agent system settings. Within this setting, we are able to characterise, for the first time, a bidder's utility-maximising strategy for bidding in any number of such auctions and for any type of bidder valuation distribution.

In more detail, we first consider a market where a single bidder, called the *global bidder*, can bid in any number of auctions, whereas the other bidders, called the *local bidders*, are assumed to bid only in a single auction. For this case, we find the following results:

- Whereas in the case of a single second-price auction a bidder's best strategy is to bid its true value, this is generally not the case for a global bidder. As we shall show, its best strategy is in fact to bid below its true value.
- We are able to prove that, even if a global bidder requires only one item, the expected utility is maximised by participating in all the auctions that are selling the desired item.
- Finding the optimal bid for each auction can be an arduous task when considering all possible combinations. However, for most common bidder valuation distributions, we are able to significantly reduce this search space.
- Empirically, we find that a bidder's expected utility is maximised by bidding at a relatively high value in one of the auctions, and equal or lower in all other auctions.

We then go on to consider markets with more than one global bidder. Due to the complexity of the problem, we combine analytical results with a discrete simulation in order to numerically

[1]To illustrate, at the time of writing, over one thousand eBay auctions were selling the iPod mini 4GB.

derive the optimal bidding strategy. By so doing, we find that, in a market with only global bidders, the optimal strategy does not converge. In fact it fluctuates between two states. If the market consists of both local and global bidders, however, the global bidders' strategy quickly reaches a stable solution and we approximate a symmetric Nash equilibrium outcome.

Finally, we consider the issue of market efficiency when there are such simultaneous auctions. Efficiency is an important system-wide consideration within multi-agent systems since it characterises how well the allocations in the system maximise the overall utility (see section 2.3). Now, efficiency is maximised when the goods are allocated to those who value them the most. However, a certain amount of inefficiency is inherent to a distributed market where the auctions are held separately. In this chapter, we measure the inefficiency of markets with local bidders only and consider the impact of global bidders on this inefficiency. In so doing, we first prove that the efficiency of distributed markets with only local bidders has a lower bound given by $1 - 1/e$. Furthermore, we find that the presence of a global bidder has a slight, but positive, impact on the efficiency when the number of local bidders is known, but is, in general, negative when there exists uncertainty about the exact number of bidders. Therefore, information about the market plays an important role in the social welfare of the system.

In the next section, we discuss how a market consisting of multiple simultaneous auctions arises within the MSN scenario we introduced in section 1.2.

3.2 Distributed Allocation within the MSN Scenario

Within this chapter, we consider a sensor network in which a trusted centre does not exist (as shown in figure 3.2). Given this constraint, the individual sensors in the region of interest then have to sell their services independently, whilst the sensors wishing to acquire data about this region will have to choose which auction(s) to attend and participate in.

Hence, this MSN can be modelled as a distributed market in which simultaneous auctions of the same sensing service are being conducted by sensors of a particular type. The bidders in this market are the sensors wishing to acquire data that this service potentially provides. Now, each of these buying sensors will attach different levels of importance to the data due to the different reasons they may require it for (e.g. a sensor that is carrying out wide-area surveillance for military purposes will be more interested in improving its view if it detects a plane-like body than a sensor interested in habitat-monitoring). As a result, each of these buying sensors will have a certain individual value for the service which will be determined by the goal set by their owners. Furthermore, the buyers do not discriminate between the services provided by the different agents and thus the item provided by each of the selling sensors can be viewed as completely substitutable. We shall consider two types of sensors in this scenario. The first type is one that is severely constrained in the bandwidth available to it and therefore decides to bid at only one auction (since it cannot commit bandwidth to receiving more than one service). In contrast, the second type has sufficient bandwidth to place bids at all the available auctions.

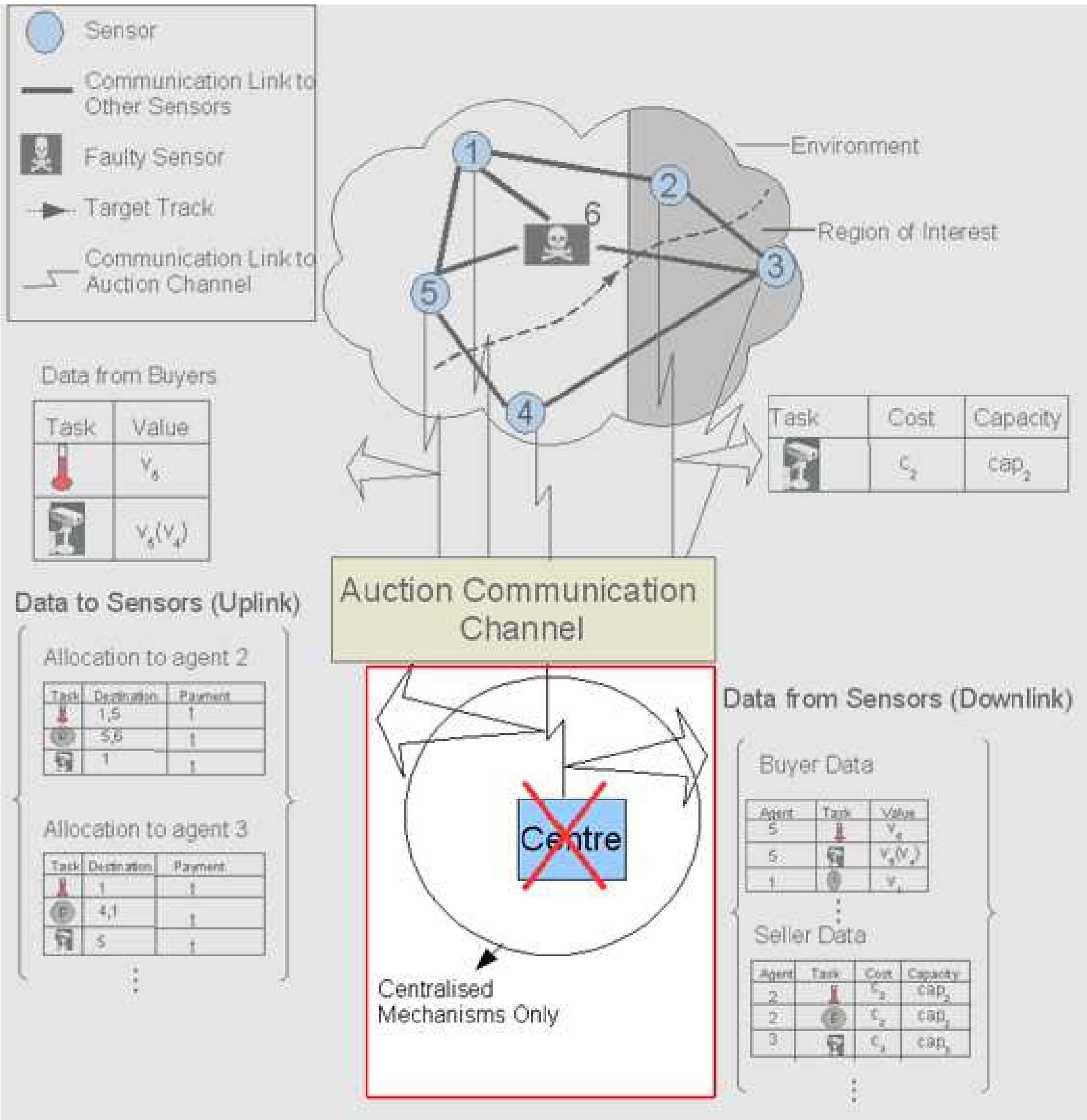


FIGURE 3.1: Multisensor scenario showing highlighting the distributed mechanism requirement addressed within this chapter.

Having thus described our scenario, we now provide an overview of the research that has been carried out within simultaneous auctions.

3.3 Related Work

Research in the area of simultaneous auctions can be segmented along two broad lines. On the one hand, there is the game-theoretic analysis of simultaneous auctions which concentrates on studying the equilibrium strategy of rational agents [Engelbrecht-Wiggans and Weber, 1979; Krishna and Rosenthal, 1996; Lang and Rosenthal, 1991; Rosenthal and Wang, 1996; Szentes and Rosenthal, 2003]. Such analyses are typically used when the auction format employed in the simultaneous auctions is the same (e.g. there are N second-price auctions or N first-price auctions). On the other hand, heuristic strategies have been developed for more complex settings when the sellers offer different types of auctions or the buyers need to buy bundles of goods over distributed auctions [Airiau and Sen, 2003; Byde et al., 2000; Greenwald et al., 2001; Anthony and Jennings, 2003]. This chapter adopts the former approach in studying a market of M second-price simultaneous auctions since this approach yields provably optimal bidding strategies. Furthermore, it allows us to predict equilibrium strategies and thus the steady state in the markets. This then allows us to place worst-case guarantees on such distributed allocation mechanisms.

In this case, the seminal paper by Engelbrecht-Wiggans and Weber [1979] provides one of the starting points for the game-theoretic analysis of distributed markets where buyers have substitutable goods. Their work analyses a market consisting of couples having equal valuations that want to bid for a dresser. Thus, the couple's bid space can at most contain two bids since the husband and wife can be at most at two geographically distributed auctions simultaneously. They derive a mixed Nash equilibrium (see section 2.2) for the special case where the number of buyers is large and also study the efficiency of such a market and show that for local bidders the market efficiency is $1 - 1/e$. Our analysis differs from theirs in that we study simultaneous auctions in which bidders have different valuations and the global bidder can bid in all the auctions simultaneously (which is entirely possible for the sensor scenario we consider (as discussed in section 3.2), as well as more generally in online auctions).

Following this, Krishna and Rosenthal [1996] then studied the case of simultaneous auctions with complementary goods. They analyse the case of both local and global bidders and characterise the bidding of the buyers and resultant market efficiency. The setting they provide is further extended to the case of common values by Rosenthal and Wang [1996]. However, neither of these works extend easily to the case of substitutable goods which we consider. This case is studied in [Szentes and Rosenthal, 2003], but the scenario considered is restricted to three sellers and two global bidders and with each bidder having the same value (and thereby knowing the value of other bidders). The space of symmetric mixed equilibrium strategies is derived for this special case, but again our result is more general.

3.4 Bidding in Multiple Vickrey Auctions

The model consists of M sellers, each of whom acts as an auctioneer. Each seller auctions one item; these items are complete substitutes (i.e., they are equal in terms of value and a bidder obtains no additional benefit from winning more than one item). The M auctions are executed simultaneously; that is, they end simultaneously and no information about the outcome of any of the auctions becomes available until the bids are placed[2]. We also assume that all the auctions

[2]Although this chapter focuses on sealed-bid auctions, where this is the case, the conditions are similar for last-minute bidding in iterative auctions such as eBay [Roth and Ockenfels, 2002].

are symmetric (i.e. a bidder does not prefer one auction over another). Finally, we assume free disposal (i.e. a bidder can freely dispose of items if it is allocated more than it requested) and bidders with quasi-linear utility functions (see Section 2.4).

3.4.1 The Auctions

The seller's auction is implemented as a second-price sealed bid auction, where the highest bidder wins but pays the second-highest price. This format has several advantages for an agent-based setting. Firstly, it is communication efficient. Secondly, for the single-auction case (i.e., where a bidder places a bid in at most one auction), the optimal strategy is to bid the true value and thus requires no computation (once the valuation of the item is known). This strategy is also weakly dominant[3] (see chapter 2) and is therefore independent of the other bidders' decisions. As a result, it requires no information about the preferences of other agents (such as the distribution of the valuations).

3.4.2 Global and Local Bidders

We distinguish between global bidders and local bidders. The former can bid in any number of auctions, whereas the latter only bid in a single auction. Local bidders are assumed to bid according to the weakly dominant strategy and bid their true valuation. We consider two ways of modelling local bidders: *static* and *dynamic*. In the first model, the number of local bidders is assumed to be known and equal to N_l for each auction. In the latter model, on the other hand, the average number of bidders is equal to N_l, but the exact number is unknown and may vary for each auction. This uncertainty is modelled using a Poisson distribution (more details are provided in Section 3.5.1).

As we will later show, a global bidder that bids optimally has a higher expected utility compared to a local bidder, even though the items are complete substitutes and a bidder only requires one of them. Nevertheless, we can identify a number of compelling reasons why not all bidders would choose to bid globally [4]:

- **Information.** Bidders may simply not be aware of other auctions selling the same type of item. Even if this is known, however, a bidder may not have sufficient information about the distribution of the valuations of other bidders and the number of participating bidders. Whereas this information is not required when bidding in a single auction (because of the

[3] A weakly dominant strategy differs from a dominant one in that employing a weakly dominant strategy results in the agent deriving at least as much utility as employing any other strategy (as opposed to deriving strictly more utility in the dominant strategy case). Since the difference is not consequential in the choice of the strategy, we shall henceforth refer to equilibrium strategy in a Vickrey auction or a VCG mechanism as dominant.

[4] It can be argued that the latter three reasons can be incorporated into the utility function of the agent so as to give a more grounded model. However, this is beyond the scope of this chapter, but nevertheless, explains why bidders do not bid globally.

dominance property in a second-price auction), it is important when bidding in multiple simultaneous auctions. Such information can be obtained by an expert user or be learned over time, but is often not available to a novice.

- **Bounded Rationality.** As will become clear from this chapter, an optimal strategy for a global bidder is harder to compute than a local one. A bidder will therefore only bid globally if the costs of computing the optimal strategy outweigh the benefits of the additional utility.
- **Participation Costs.** Although the bidding itself may be automated by an autonomous agent, it still takes time and/or money, such as entry fees and time to setup an account, to participate in a new auction. Occasional users may not be willing to make such an investment, and they restrict themselves to sellers or auctions with which they are familiar.
- **Risk Attitude.** Although a global bidder obtains a higher utility on average, such a bidder runs a risk of incurring a loss (i.e., a negative utility) when winning multiple auctions. A risk averse bidder may not be willing to take that chance, and so may choose to participate only in a single auction to avoid such a potential loss.
- **Budget Constraints.** Related to the previous point, a budget constrained bidder may not have sufficient funds to make a loss in case it wins more than one auction. In more detail, for a fixed budget b, the sum of bids should not exceed b, thereby limiting the number of auctions a bidder can participate in and/or lowering the actual bids that are placed in those auctions.

From the above, we believe it is reasonable to expect a combination of global and local bidders, and for only a few of them to be global bidders. In this chapter, we analyse the case of a single global bidder theoretically, and then use a computational approach to address the case with at least two such bidders.

3.5 A Single Global Bidder

In this section, we provide a theoretical analysis of the optimal bidding strategy for a global bidder, given that all other bidders are local and simply bid their true valuation. After we describe the global bidder's expected utility in Section 3.5.1, we show in Section 3.5.2 that it is always optimal for such a bidder to participate in the maximum number of auctions available. Subsequently, in Section 3.5.3 we discuss how to significantly reduce the complexity of finding the optimal bids for the multi-auction problem, and we then apply these methods to find optimal strategies for specific examples.

3.5.1 The Global Bidder's Expected Utility

We use the following notation. The number of sellers (or auctions) is $M \geq 2$ and the number of *local* bidders is $N_l \geq 1$. A bidder's valuation $v \in [0, v_{max}]$ is randomly drawn from a cumulative distribution F with probability density f, where f is continuous, strictly positive and has support $[0, v_{max}]$. A global bid $\mathcal{B}$ is a set containing a bid $b_i \in [0, v_{max}]$ for each auction $1 \leq i \leq M$ (the bids may be different for different auctions). For ease of exposition, we introduce the cumulative distribution function for the first-order statistics $G(b) = F(b)^{N_l} \in [0, 1]$, denoting the probability of winning a specific auction conditional on placing bid b in this auction, and its probability density $g(b) = dG(b)/db = N_l F(b)^{N_l - 1} f(b)$. Now, the expected utility U for a global bidder with global bid $\mathcal{B}$ and valuation v is given by:

$$U(\mathcal{B}, v) = v \left[1 - \prod_{b_i \in \mathcal{B}} (1 - G(b_i)) \right] - \sum_{b_i \in \mathcal{B}} \int_0^{b_i} y g(y) dy \tag{3.1}$$

Here, the left part of the equation is the valuation multiplied by the probability that the global bidder wins *at least* one of the M auctions and thus corresponds to the expected benefit. In more detail, note that $1 - G(b_i)$ is the probability of *not* winning auction i when bidding b_i, $\prod_{b_i \in \mathcal{B}} (1 - G(b_i))$ is the probability of not winning any auction, and thus $1 - \prod_{b_i \in \mathcal{B}} (1 - G(b_i))$ is the probability of winning at least one auction. The right part of equation 3.1 corresponds to the total expected costs or payments. To see the latter, note that the expected payment of a single second-price auction when bidding b equals $\int_0^b y g(y) dy$ (see [Krishna, 2002]) and is independent of the expected payments for other auctions.

Clearly, equation 3.1 applies to the model with *static* local bidders (i.e., where the number of bidders is known and equal for each auction (see Section 3.4.2)). However, we can use the same equation to model *dynamic* local bidders in the following way:

Lemma 1. By replacing the first-order statistic $G(y)$ with:

$$\hat{G}(y) = e^{N_l (F(y) - 1)} \tag{3.2}$$

and the corresponding density function $g(y)$ with $\hat{g}(y) = N_l \, f(y) e^{N_l (F(y) - 1)}$, equation 3.1 becomes the expected utility where the number of local bidders in each auction is described by a Poisson distribution with average N_l (i.e. where the probability that n local bidders participate is given by $P(n) = N_l^n e^{-N_l} / n!$).

Proof. To prove this, we first show that $G(\cdot)$ and $F(\cdot)$ can be modified such that the number of bidders per auction is given by a *binomial* distribution (where a bidder's decision to participate is given by a Bernoulli trial [5]) as follows:

$$G'(y) = F'(y)^{\mathcal{N}\updownarrow} = (1 - p + p\,F(y))^N, \tag{3.3}$$

where p is the probability that a bidder participates in the auction, and N is the total number of bidders. To see this, note that not participating is equivalent to bidding zero. As a result, $F'(0) = 1 - p$ since there is a $1 - p$ probability that a bidder bids zero at a specific auction, and $F'(y) = F'(0) + p\,F(y)$ since there is a probability p that a bidder bids according to the original distribution $F(y)$. Now, the average number of participating bidders is given by $N_l = p\,N$. By replacing p with N_l/N, equation 3.3 becomes $G'(y) = (1 - N_l/N + (N_l/N)F(y))^N$. Note that a Poisson distribution is given by the limit of a binomial distribution. By keeping N_l constant and taking the limit $N \to \infty$, we then obtain $G'(y) = e^{N_l(F(y)-1)} = \hat{G}(y)$. $\square$

The results that follow apply to both the static and dynamic model unless stated otherwise.

3.5.2 Participation in Multiple Auctions

We now show that, for any valuation $0 < v < v_{max}$, a utility-maximising global bidder should always place non-zero bids in all available auctions. To prove this, we show that the expected utility increases when placing an arbitrarily small bid compared to not participating in an auction. More formally:

Theorem 3.1. *Consider a global bidder with valuation $0 < v < v_{max}$ and global bid $\mathcal{B}$, where $b_i \leq v$ for all $b_i \in \mathcal{B}$. Suppose $b_j \notin \mathcal{B}$ for $j \in \{1, 2, \ldots, M\}$, then there exists a $b_j > 0$ such that $U(\mathcal{B} \cup \{b_j\}, v) > U(\mathcal{B}, v)$.*

Proof. Using equation 3.1, the marginal expected utility for participating in an additional auction can be written as:

$$U(\mathcal{B} \cup \{b_j\}, b) - U(\mathcal{B}, v) = vG(b_j) \prod_{b_i \in \mathcal{B}} (1 - G(b_i)) - \int_0^{b_j} yg(y)dy \tag{3.4}$$

Now, using integration by parts, we have $\int_0^{b_j} yg(y) = b_jG(b_j) - \int_0^{b_j} G(y)dy$ and the above equation can be rewritten as:

$$U(\mathcal{B} \cup \{b_j\}, b) - U(\mathcal{B}, v) = G(b_j) \left[v \prod_{b_i \in \mathcal{B}} (1 - G(b_i)) - b_j \right] + \int_0^{b_j} G(y)dy \tag{3.5}$$

Let $b_j = \epsilon$, where ϵ is an arbitrarily small strictly positive value. Clearly, $G(b_j)$ and $\int_0^{b_j} G(y)dy$ are then both strictly positive (since $f(y) > 0$). Moreover, given that $b_i \leq v < v_{max}$ for $b_i \in \mathcal{B}$

[5] This is a commonly assumed distribution that governs the participation of bidders within auctions [David et al., 2005]

and that $v > 0$, it follows that $v \prod_{b_i \in \mathcal{B}}(1 - G(b_i)) > 0$. Now, suppose $b_j = \frac{1}{2} v \prod_{b_i \in \mathcal{B}}(1 - G(b_i))$, then $U(\mathcal{B} \cup \{b_j\}, b) - U(\mathcal{B}, v) = G(b_j) \left[\frac{1}{2} v \prod_{b_i \in \mathcal{B}}(1 - G(b_i))\right] + \int_0^{b_j} G(y)dy > 0$ and thus $U(\mathcal{B} \cup \{b_j\}, b) > U(\mathcal{B}, v)$. □

3.5.3 The Optimal Global Bid

A general solution to the optimal global bid requires the maximisation of equation 3.1 in M dimensions, an arduous task, even when applying numerical methods. In this section, however, we show how to reduce the entire bid space to two dimensions in most cases (one continuous, and one discrete), thereby significantly simplifying the problem at hand. First, however, in order to find the optimal solutions to equation 3.1, we set the partial derivatives to zero:

$$\frac{\partial U}{\partial b_i} = g(b_i) \left[v \prod_{b_j \in \mathcal{B} \setminus \{b_i\}} (1 - G(b_j)) \; - b_i \right] = 0 \tag{3.6}$$

Now, equality 3.6 holds when either $g(b_i) = 0$ or $\prod_{b_j \in \mathcal{B} \setminus \{b_i\}}(1 - G(b_j))v - b_i = 0$. In the dynamic model, $g(b_i)$ is always greater than zero, and can therefore be ignored (since $g(0) = Nf(0)e^{-N_l}$ and we assume $f(y) > 0$). In the static model, $g(b_i) = 0$ only when $b_i = 0$. However, theorem 3.1 shows that the optimal bid is non-zero for $0 < v < v_{max}$. Therefore, we can ignore the first part, and the second part yields:

$$b_i = v \prod_{b_j \in \mathcal{B} \setminus \{b_i\}} (1 - G(b_j)) \tag{3.7}$$

In other words, the optimal bid in auction i is equal to the bidder's valuation multiplied by the *probability of not winning any of the other auctions*. It is straightforward to show that the second partial derivative is negative, confirming that the solution is indeed a maximum when keeping all other bids constant. Thus, equation 3.7 provides a means to derive the optimal bid for auction i, given the bids in all other auctions.

3.5.3.1 Reducing the Search Space

In what follows, we show that, for non-decreasing probability density functions, such as the uniform and logarithmic distribution, the optimal global bid consists of at most two different values for any $M \geq 2$. That is, the search space for finding the optimal bid can then be reduced to two continuous values. Let these values be b_{high} and b_{low}, where $b_{high} \geq b_{low}$. More formally:

Theorem 3.2. *Suppose the probability density function f is non-decreasing within the range $[0, v_{max}]$, then the following proposition holds: given $v > 0$, for any $b_i \in \mathcal{B}$, either $b_i = b_{high}$, $b_i = b_{low}$, or $b_i = b_{high} = b_{low}$.*

Proof. Using equation 3.7, we can produce M equations, one for each auction, with M unknowns. Now, by combining these equations, we obtain the following relationship: $b_1(1 - G(b_1)) = b_2(1 - G(b_2)) = \ldots = b_m(1 - G(b_m))$. By defining $H(b) = b(1 - G(b))$ we can rewrite the equation to:

$$H(b_1) = H(b_2) = \ldots = H(b_m) = v \prod_{b_j \in \mathcal{B}} (1 - G(b_j)) \tag{3.8}$$

In order to prove that there exist at most two different bids, it is sufficient to show that $b = H^{-1}(y)$ has at most two solutions that satisfy $0 \leq b \leq v_{max}$ for any y. To see this, suppose there exists a third solution $b_j \neq b_{low} \neq b_{high}$. From equation 3.8 it then follows that there exists a y such that $H(b_j) = H(b_{low}) = H(b_{high}) = y$. Therefore, $H^{-1}(y)$ must have at least three solutions, which is a contradiction.

Note that a sufficient condition for the above to hold is for $H(b)$ to be strictly concave[6] for $0 \leq b \leq v_{max}$. Now, the function H is strictly concave if and only if the following holds:

$$\frac{d^2H}{db^2} = \frac{d}{db}\left(1 - b \cdot g(b) - G(b)\right) = -\left(b\frac{dg}{db} + 2g(b)\right) < 0$$

By performing standard calculations, we obtain the following condition for the static model:

$$b\left((N_l - 1)\frac{f(b)^{N_l}}{F(b)} + N_l\frac{f'(b)}{f(b)}\right) > -2 \; for \; 0 \leq b \leq v_{max}, \tag{3.9}$$

and similarly for the dynamic model we have:

$$b\left(N_l\, f(b) + \frac{f'(b)}{f(b)}\right) > -2 \; for \; 0 \leq b \leq v_{max}, \tag{3.10}$$

where $f'(b) = df/db$. Since both f and F are positive, conditions 3.9 and 3.10 clearly hold for $f'(b) \geq 0$. In other words, conditions 3.9 and 3.10 show that $H(b)$ is strictly concave when the probability density function is non-decreasing for $0 \leq b \leq v_{max}$. □

Note from conditions 3.9 and 3.10 that the requirement of non-decreasing density functions is sufficient, but far from necessary. Although we are as yet not able to make a more precise formal characterisation, in practice even most density functions with decreasing parts satisfy these conditions. Moreover, the requirement for $H(b)$ to be strictly concave is also stronger than necessary in order to guarantee only two solutions. As a result, for practical purposes, we expect the reduction of the search space to apply in most cases.

Given there are at most 2 possible bids, b_{low} and b_{high}, we can further reduce the search space by expressing one bid in terms of the other. Suppose the buyer places a bid of b_{low} in M_{low}

[6] More precisely, $H(b)$ can be either strictly convex or strictly concave. However, it is easy to see that H is not convex since $H(0) = H(v_{max}) = 0$, and $H(b) \geq 0$ for $0 < b < v_{max}$.

auctions and b_{high} for the remaining $M_{high} = M - M_{low}$ auctions, equation 3.7 then becomes:

$$b_{low} = v(1 - G(b_{low}))^{M_{low}-1}(1 - G(b_{high}))^{M_{high}},$$

and can be rearranged to give:

$$b_{high} = G^{-1}\left(1 - \left[\frac{b_{low}}{v(1 - G(b_{low}))^{M_{low}-1}}\right]^{\frac{1}{M_{high}}}\right) \tag{3.11}$$

Here, the inverse function $G^{-1}(\cdot)$ can usually be obtained quite easily. Furthermore, note that, if $M_{low} = 1$ or $M_{high} = 1$, equation 3.7 can be used directly to find the desired value.

Using the above, we are able to reduce the bid search space to a single continuous dimension, given M_{low} or M_{high}. However, we do not know the number of auctions in which to bid b_{low} and b_{high}, and thus we need to search M different combinations to find the optimal global bid. Moreover, for each combination, the optimal b_{low} and b_{high} can vary. Therefore, in order to find the optimal bid for a bidder with valuation v, it is sufficient to search along one continuous variable $b_{low} \in [0, v]$, and a discrete variable $M_{low} = M - M_{high} \in \{1, 2, \ldots, M\}$.

3.5.3.2 Empirical Evaluation

In this section, we present results from an empirical study and characterise the optimal global bid for specific cases. Furthermore, we measure the actual utility improvement that can be obtained when using the global strategy. The results presented here are based on a uniform distribution of the valuations with $v_{max} = 1$, and the static local bidder model, but they generalise to the dynamic model and other distributions. Figure 3.2 illustrates the optimal global bids and the corresponding expected utility for various M and $N_l = 5$, but again the bid curves for different values of M and N_l follow a very similar pattern. Here, the bid is normalised by the valuation v to give the bid fraction $x = b/v$. Note that, when $x = 1$, a bidder bids its true value.

As shown in Figure 3.2, for bidders with a relatively low valuation, the optimal strategy is to submit M equal bids at, or very close to, the true value. The optimal bid fraction then gradually decreases for higher valuations. Interestingly, in most cases, placing equal bids is no longer the optimal strategy after the valuation reaches a certain point. At this point, a so-called pitchfork bifurcation is observed and the optimal bids split into two values: a single high bid and $M - 1$ low bids. This transition is smooth for $M = 2$, but exhibits an abrupt jump for $M \geq 3$. In all experiments, however, we consistently observe that the optimal strategy is always to place a high bid in one auction, and an equal or lower bid in all other auctions. In case of a bifurcation and when the valuation approaches v_{max}, the optimal high bid becomes very close to the true value and the low bids go to almost zero[7].

[7]Note in Figure 3.2 that the low bids are significantly higher than zero at this point. This is because as v approaches v_{max}, the low bids have very little impact on the utility and finding the optimum numerically at this point requires an extremely high precision.

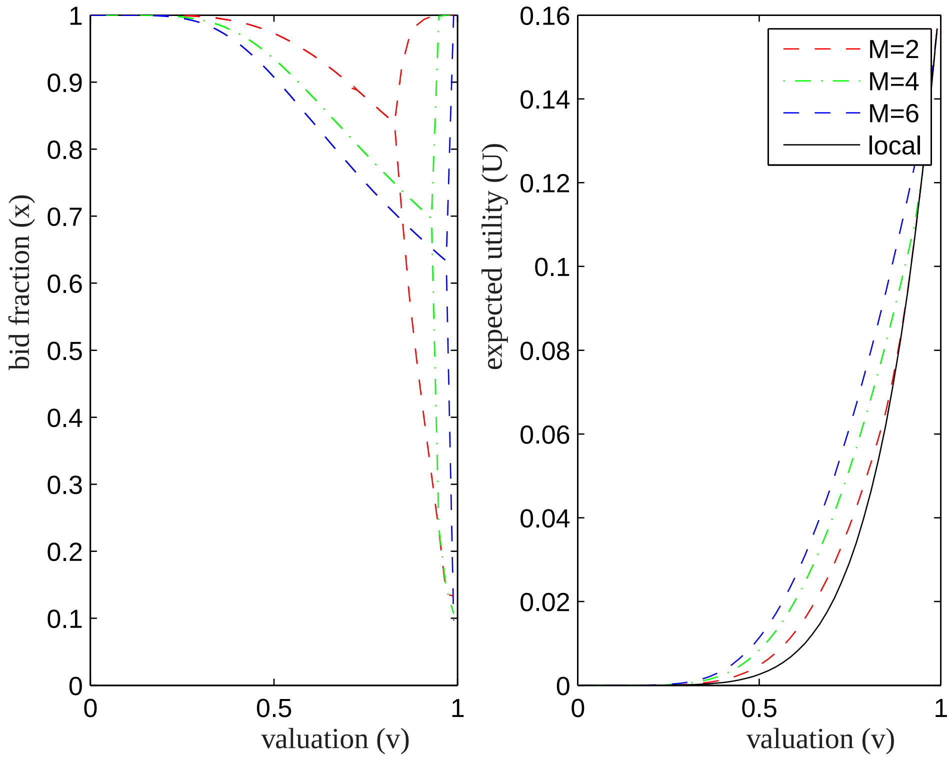


FIGURE 3.2: The optimal bid fractions $x = b/v$ and corresponding expected utility for $N_l = 5$, static local bidders, and varying M.

As illustrated in Figure 3.2, the utility of a global bidder becomes progressively higher with more auctions. In absolute terms, the improvement is especially high for bidders that have an above average valuation, but not too close to v_{max}. The bidders in this range thus benefit most from bidding globally. This is because bidders with very low valuations have a very small chance of winning any auction, whereas bidders with a very high valuation have a high probability of winning a single auction and benefit less from participating in more auctions. In contrast, if we consider the utility *relative* to bidding in a single auction, this is much higher for bidders with relatively low valuations . In particular, we notice that a global bidder with a low valuation can improve its utility by up to M times the expected utility of bidding locally. Intuitively, this is because the chance of winning one of the auctions increases by up to a factor M, whereas the increase in the expected cost is negligible. For high valuation buyers, however, the benefit is not that obvious because the chances of winning are relatively high even in case of a single auction.

3.6 Multiple Global Bidders

As argued in section 3.4.2, we expect a real-world market to exhibit a mix of global and local bidders. Whereas so far we assumed a single global bidder, in this section we consider a setting where multiple global bidders and local bidders interact. The analysis of this problem is complex, however, as the optimal bidding strategy of a global bidder depends on the strategy of other global bidders. A typical analytical approach is to find the symmetric Nash equilibrium solution [Engelbrecht-Wiggans and Weber, 1979; Gerding et al., 2006b; Rosenthal and Wang, 1996; Szentes and Rosenthal, 2003], which occurs when all global bidders use the same strategy to produce their bids, and no (global) bidder has any incentive to unilaterally deviate from the chosen strategy. Due to the complexity of the problem, however, here we combine a computational simulation approach with the analytical results from section 3.5. The simulation works by iteratively finding the best response to the optimal bidding strategies in the previous iteration. If this results in a stable outcome (i.e., when the current and previous optimal bidding strategies are the same), the solution is by definition a (symmetric) Nash equilibrium.

In more detail, the simulation is based on the observation that the valuation distribution F of the local bidders corresponds to the distribution of bids (since local bidders bid their true valuation). Therefore, by maximising equation 3.1 we find the best response given the current distribution of bids. Now, we first discretize the space of possible valuations and bids. Then, by performing this maximisation for each bidder *type*, where a bidder type is defined by its (discrete) valuation v, we find a new distribution of bids. Note that this distribution can include bids from any number of both global and local bidders, where the latter simply bid their true valuation. This distribution of bids can then be used to find a new best response, resulting in a new distribution of bids, and so on, for a fixed number of iterations or until a stable solution has been found.

In what follows, we first describe the simulation settings, and then apply the simulation to settings with global bidders only, followed by settings with both global and local bidders.

3.6.1 The Setting

The simulation is based on discrete valuations and bids. The valuations are natural numbers ranging from 1 to $v_{max} \in \mathbb{N}$, where v_{max} is set to 1000. Each valuation $v \in \{1, 2, \ldots, v_{max}\}$ occurs with equal probability, equivalent to a uniform valuation distribution in the continuous case. Note, however, that even though the bidder valuations are distributed uniformly, the resulting distribution of bids is typically not uniform (since global bidders typically bid below their valuation). The number of different bid levels that a bidder is allowed is set to $\mathcal{L} \in \mathbb{N}$. Thus, a bidder with valuation v can place the bids $b \in \{v/\mathcal{L}, 2v/\mathcal{L}, \ldots, v\}$. For the results reported here, we use $\mathcal{L} = 300$. The initial state can play an important role in the experiments. Therefore, to ensure our results are robust, experiments are repeated with different random initial bid distributions. In the following, we assume the number of local bidders to be static and use N_g and N_l to denote the number of global and local bidders respectively.

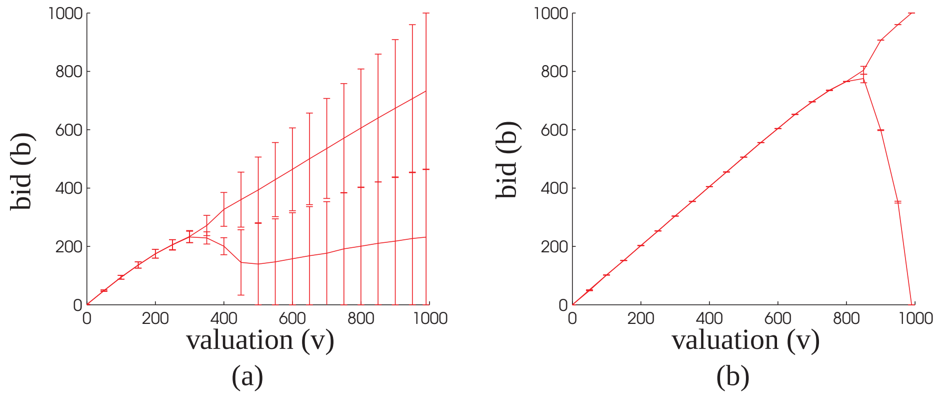


FIGURE 3.3: Best response strategy for 2 auctions and 3 global bidders without local bidders (a), and with 10 local bidders (b), averaged over 10 iterations and 20 runs with different initial conditions. The measurements are taken after an initialisation period of 10 iterations. The error-bars indicate the standard deviation.

3.6.2 The Results

First, we describe the results with no local bidders (i.e., $N_l = 0$). For this case, we find that the simulation does not converge to a stable state. That is, when the number of (global) bidders is at least 2, the best response strategy keeps fluctuating, irrespective of the number of iterations, and of the initial state. The fluctuations, however, show a distinct pattern and more or less alternate between two states. Specifically, figure 3.3a depicts the average best response strategy for $N_g = 3$ and $M = 2$. Here, the standard deviation is a gauge for the amount of fluctuation and thus the instability of the strategy. In general, we find that the best response for low valuations remain stable, whereas the strategy for bidders with high valuations fluctuates heavily, as is shown in Figure 3.3a. These results are robust for different initial conditions and simulation parameters.

If we include local bidders, on the other hand, we observe that the strategies stabilise. In particular, Figure 3.3b shows the simulation results for the same settings as before except with both local and global bidders. As can be seen from this figure, the variation is very slight and only around the bifurcation point. We note that these outcomes are obtained after only a few iterations of the simulation. The results show that the principal conclusions in case of a single global bidder carry over to the case of multiple global bidders. That is, the optimal strategy is to participate in all auctions and to bid high in one auction, and equal or lower in other auctions. A similar bifurcation point is also observed. These results are also obtained for other values of M, N_l, and N_g. Moreover, the results are very robust to changes to the parameters of the simulation.

To conclude, even though a theoretical analysis proves difficult in case of several global bidders, we can approximate a (symmetric) Nash equilibrium for specific settings using a discrete simulation in case the system consists of both local and global bidders. Our experiments show that, even in the case of multiple global bidders, the best strategy is to bid in multiple auctions.

Thus, our simulation can be used as a tool to predict the market equilibrium and to find the optimal bidding strategy for practical settings where we expect a combination of local and global bidders.

3.7 Market Efficiency

Efficiency is an important system-wide property since it characterises to what extent the market maximises social welfare (i.e. the sum of utilities of all agents in the market). To this end, in this section we study the efficiency of markets with either static or dynamic local bidders, and the impact that a global bidder has on the efficiency in these markets. Specifically, efficiency in this context is maximised when the bidders with the M highest valuations in the entire market obtain a single item each. More formally, we define the efficiency of an allocation as:

Definition 3.3. Efficiency of Allocation. The efficiency η_K of an allocation K is the obtained social welfare proportional to the maximum social welfare that can be achieved in the market and is given by:

$$\eta_K = \frac{\sum_{i=1}^{N} v_i(K)}{\sum_{i=1}^{N} v_i(K^*)}, \tag{3.12}$$

where $K^* = \arg\max_{K \in \mathcal{K}} \sum_{i=1}^{N} v_i(K)$ is an efficient allocation, $\mathcal{K}$ is the set of all possible allocations, and $v_i(K)$ is bidder i's utility for the allocation $K \in \mathcal{K}$.

Now, as argued in the section 3.1, a market consisting of a number of distributed sellers is likely to be inefficient [8]. Within this market, the loss of efficiency arises from the fact that when the buyers select an auction they are not aware of the other buyers who are going to be in that auction. As a result, despite the individual second-price auctions being efficient (see chapter 2), the overall market might not be so. In order to visualise the effect of the buyers' decision, consider a market of 3 buyers, $\{B_1, B_2, B_3\}$ (ordered such that B_1 has the highest valuation and B_3 has the lowest) and 2 sellers, $\{S_1, S_2\}$. Then there is a possibility that B_1 and B_2 compete against one another in one auction thereby allowing B_3 to win the other auction. This results in an overall inefficient allocation since B_1 and B_3 win a good each. Figure 3.4 shows four possible outputs which could result in this auction, with the first one being the inefficient output discussed. This loss could certainly be mitigated if the buyers were free to move across auctions and were aware of the participants in each auction. This is however a case that we do not consider here since we are interested in knowing how much efficiency is lost in a distributed market.

[8] An exception is when $N_l = 1$ and bidders are static, since the market is then completely efficient without a global bidder. However, since this is a very special case and does not apply to other settings, we do not discuss it further here.

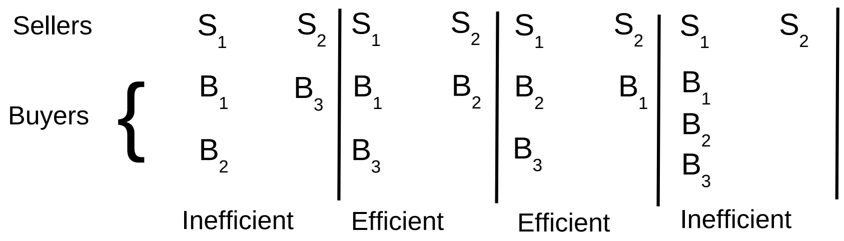


FIGURE 3.4: Four states for a market consisting of two distributed sellers each selling a single homogeneous good and three buyers each interested in a single good. The other four states are mirror images.

3.7.1 Local Bidders Only

The expected efficiency of a market consisting of local bidders only would depend on the probability that each of the possible allocations occur. Now, there are two statistical variables which will affect the calculation of the expected efficiency: (i) the values drawn by the bidders and (ii) the auctioneer at which they elect to bid. The former depends on $f(v)$ whereas the latter is dependent on the strategy of the bidders (which in the case of uninformed buyers is to pick an auctioneer at random). We first derive the efficiency of a market when the values of the buyers are known.

Theorem 3.4. *A market consisting of several distributed auctions conducted by M sellers with N uninformed buyers (whose values for the object are ordered as $\{v_1, \ldots, v_N\}$) has an expected efficiency of:*

$$E(\eta, v_1, \ldots, v_N, M) = \frac{\sum_{i=1}^{N} v_i (\frac{M-1}{M})^{i-1}}{\sum_{i=1}^{\min(M,N)} v_i} \tag{3.13}$$

Proof. The expected efficiency of the market is dependent upon the expected overall value that the market derives. This, in turn, is dependent on two factors; namely the value of a particular allocation and the probability that that allocation occurs. The probability that buyer i wins a particular auction conducted by seller j is given by:

$$\begin{aligned} P(i \text{ wins auction } j) &= \prod_{k=1}^{i-1} P(k \text{ does not turn up at auction} j) \\ &\quad \times P(i \text{ turns up at an auction} j) \\ &= \left(\frac{M-1}{M}\right)^{i-1} \times \frac{1}{M} \end{aligned}$$

Since there are M independent auctions being held, then the probability that i wins is given by $\left(\frac{M-1}{M}\right)^{i-1}$. Thus, the expected utility from this market is given by:

$$E(\eta, v_1, \ldots, v_N, M) = \frac{\sum_{i=1}^{N} v_i (\frac{M-1}{M})^{i-1}}{\sum_{i=1}^{\min(M,N)} v_i}$$

□

Equation 3.13 provides us with an expected efficiency measure given the number of buyers, N, the number of sellers, M, and the buyer's valuations $\{v_{B_1}, \ldots, v_{B_N}\}$. Now, if the valuations are not known, we can derive the expected efficiency from the probability density function of the valuations as:

$$\begin{aligned} E(\eta, f(v), M, N) &= E\left[\frac{\sum_{i=1}^{N} v_i (\frac{M-1}{M})^{i-1}}{\sum_{i=1}^{\min(M,N)} v_i}\right] \\ &= \sum_{i=1}^{N} E\left[\frac{v_i (\frac{M-1}{M})^{i-1}}{\sum_{i=1}^{\min(M,N)} v_i}\right] \end{aligned}$$

Figure 3.5 shows the expected efficiency that occurs when the number of buyers is varied from 1 to 25 and the number of sellers is varied from 3 to 15 and the buyers' valuations are drawn from the uniform distribution $\mathcal{U}[0, 1]$. As can be observed, there is first an initial linear decrease in efficiency and then there is a logarithmic increase in efficiency as the number of buyers increase. This is because there are two effects which are being seen in the graphs. As the number of buyers increase, there are more possible states where inefficiency can occur (i.e. when high-value buyers end up within the same auction.). However, simultaneously, the buyers' valuations are less likely to be very different such that the resultant loss of efficiency is small. At the beginning, when the number of buyers is roughly less than or equal to the number of sellers, the first effect dominates thereby decreasing the efficiency. Then the second effect starts being the dominating factor as the number of buyers increase. Another interesting point to note from the above graph is that the minimum expected efficiency of this protocol decreases at a decreasing rate as the number of sellers increase. This suggests that the minimum expected efficiency is probably bounded. In fact, the expected efficiency is bounded at $1 - 1/e$ as we now prove.

Theorem 3.5. *The minimum expected efficiency of a distributed market consisting of uninformed buyers is bounded at* $1 - 1/e$*.*

Proof. Let x be such that:

$$x = \frac{\sum_{i=1}^{\min(M,N)} v_i}{\min(M, N)}$$

Then, since $\frac{M-1}{M} < 1$, it follows that:

$$\sum_{i=1}^{N} v_i (\frac{M-1}{M})^{i-1} \geq \sum_{i=1}^{N} x (\frac{M-1}{M})^{i-1}$$

which implies that:

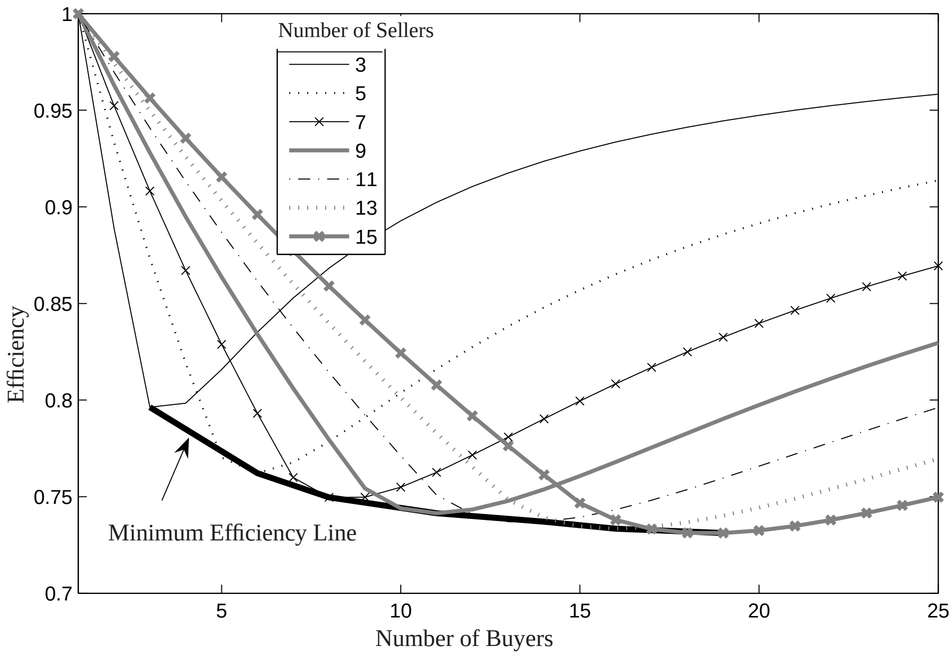


FIGURE 3.5: Expected efficiency of distributed market (singly-endowed sellers and single-object buyers) of homogeneous goods with upto 15 sellers and 25 buyers and buyers' valuations drawn from a uniform distribution.

$$\frac{\sum_{i=1}^{N} v_i (\frac{M-1}{M})^{i-1}}{\sum_{i=1}^{\min(M,N)} v_i} \geq \frac{\sum_{i=1}^{N} x (\frac{M-1}{M})^{i-1}}{x \min(M,N)}$$

and hence, from equation 3.13:

$$E(\eta, v_1, \ldots, v_N, M) \geq \frac{\sum_{i=1}^{N} x (\frac{M-1}{M})^{i-1}}{x \min(M,N)} \tag{3.14}$$

Now, consider the following two cases within this market, namely $M < N$ and $N \leq M$. In the former case, the inequality given by 3.14 becomes:

$$\begin{aligned} E(\eta, v_1, \ldots, v_N, M) &\geq \frac{\sum_{i=1}^{N} x (\frac{M-1}{M})^{i-1}}{xM} \\ &\geq \frac{\sum_{i=1}^{M} x (\frac{M-1}{M})^{i-1}}{xM} \\ &\geq \frac{\sum_{i=1}^{M} (\frac{M-1}{M})^{i-1}}{M} \end{aligned}$$

and in the latter case, the inequality is:

$$
\begin{aligned}
E(\eta, v_1, \ldots, v_N, M) &\geq \frac{\sum_{i=1}^{N} x(\frac{M-1}{M})^{i-1}}{xN} \\
&\geq \frac{\sum_{i=1}^{N}(\frac{M-1}{M})^{i-1}}{N} \\
&\geq \frac{\sum_{i=1}^{M}(\frac{M-1}{M})^{i-1}}{N} - \frac{\sum_{i=N+1}^{M}(\frac{M-1}{M})^{i-1}}{N} \\
&\geq \frac{\sum_{i=1}^{M}(\frac{M-1}{M})^{i-1}}{M} + \frac{M-N}{N}\frac{\sum_{i=1}^{M}(\frac{M-1}{M})^{i-1}}{M} \\
&\quad - \frac{\sum_{i=N+1}^{M}(\frac{M-1}{M})^{i-1}}{M} - \frac{M-N}{N}\frac{\sum_{i=N+1}^{M}(\frac{M-1}{M})^{i-1}}{M} \\
&\geq \frac{\sum_{i=1}^{M}(\frac{M-1}{M})^{i-1}}{M} - \frac{\sum_{i=1}^{M}(\frac{M-1}{M})^{i-1}}{M} \\
&\quad + \frac{M}{N}\frac{\sum_{i=1}^{M}(\frac{M-1}{M})^{i-1} - \sum_{i=N+1}^{M}\frac{M-1}{M})^{i-1}}{M} \\
&\geq \frac{\sum_{i=1}^{M}(\frac{M-1}{M})^{i-1}}{M} - \frac{\sum_{i=1}^{M}(\frac{M-1}{M})^{i-1}}{M} \\
&\quad + \frac{\sum_{i=1}^{N}(\frac{M-1}{M})^{i-1}}{N} \\
&> \frac{\sum_{i=1}^{M}(\frac{M-1}{M})^{i-1}}{M}
\end{aligned}
$$

Hence, we can deduce from both scenarios that:

$$
E(\eta, v_1, \ldots, v_N, M) \geq \frac{\sum_{i=1}^{M}(\frac{M-1}{M})^{i-1}}{M}
$$

Now,

$$
\begin{aligned}
\frac{\sum_{i=1}^{M}(\frac{M-1}{M})^{i-1}}{M} &= \frac{1}{M}\frac{1-\left(\frac{M-1}{M}\right)^{M}}{1-\frac{M-1}{M}} \\
&= 1 - \left(\frac{M-1}{M}\right)^{M}
\end{aligned}
$$

Hence, the limit of minimum efficiency as M tends to infinity is:

$$
\begin{aligned}
\lim_{M\to\infty} E(\eta, v_1, \ldots, v_N, M) &= \lim_{M\to\infty}\left[1 - \left(\frac{M-1}{M}\right)^{M}\right] \\
&= 1 - \lim_{M\to\infty}\left(\frac{M-1}{M}\right)^{M} \\
&= 1 - \frac{1}{e}
\end{aligned}
$$

□

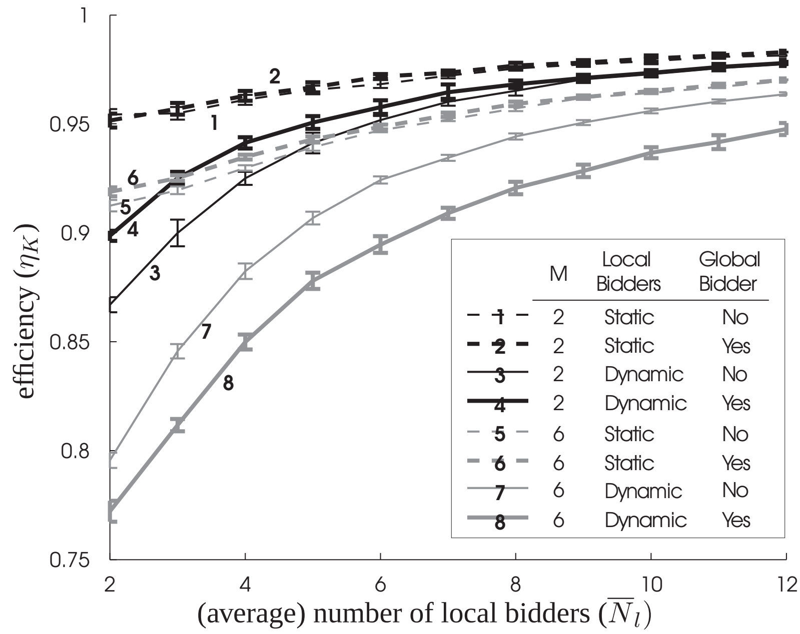


FIGURE 3.6: Average efficiency for different market settings as shown in the legend. The error-bars indicate the standard deviation over the 10 runs.

This lower bound is significant in two ways. From a theoretical perspective, it shows that distributed markets have an inherent efficiency, despite the fact that there is no centre coordinating the allocation within the system. From a practical perspective, this provides a benchmark for the design of a distributed system involving coordination (e.g. buyers/sellers forming coalitions to buy/sell goods, the overall actions of buyers and sellers being transparent via a communicating device (such as in the CDA)). The designer will have to compare the cost of implementing the communication protocol developed for coordination with the gain in efficiency of the system over $1 - 1/e$. Hence, in terms of requirement 1 (outlined in Chapter 1), this implies that distributed mechanisms can be implemented without too high a cost in terms of efficiency.

3.7.2 With Global Bidders

Now, in order to measure the efficiency of the market and the impact of a global bidder, we run simulations for the markets with the different types of local bidders. The experiments are carried out as follows. Each bidder's valuation is drawn from a uniform distribution with support $[0, 1]$. The local bidders bid their true valuations, whereas the global bidder bids optimally in each auction as described in Section 3.5.3. The experiments are repeated 5000 times for each run to obtain an accurate mean value, and the final average results and standard deviations are taken over 10 runs in order to get statistically significant results.

The results of these experiments are shown in Figure 3.6. As can be observed, the efficiency increases when N becomes larger. This is because the differences between the bidders with the highest valuations become smaller, thereby decreasing the loss of efficiency.

Furthermore, Figure 3.6 shows that the presence of a global bidder has a slightly positive effect on the efficiency in case the local bidders are static. In the case of dynamic bidders, however, the effect of a global bidder depends on the number of sellers. Thus, if M is low (i.e., for $M = 2$), a global bidder significantly increases the efficiency, especially for low values of N_l. For $M = 6$, on the other hand, the presence of a global bidder has a negative effect on the efficiency (this effect becomes even more pronounced for higher values of M). This result is explained as follows. The introduction of a global bidder potentially leads to a decrease of efficiency since this bidder can unwittingly win more than one item. However, as the number of local bidders increase, this is less likely to happen. Rather, since the global bidder increases the number of bidders, its presence makes an overall positive (albeit small) contribution in case of static bidders. In a market with dynamic bidders, however, the market efficiency depends on two other factors. On the one hand, the efficiency increases since items no longer remain unsold (this situation can occur in the dynamic model when no bidder turns up at an auction). On the other hand, as a result of the uncertainty concerning the actual number of bidders, a global bidder is more likely to win multiple items (we confirmed this analytically). As M increases, the first effect becomes negligible, whereas the second one becomes more prominent, reducing the efficiency on average.

To conclude, the impact of a global bidder on the efficiency clearly depends on the information that is available. In case of static local bidders, the number of bidders is known and the global bidder can bid more accurately. In case of uncertainty, however, the global bidder is more likely to win than one item, decreasing the overall efficiency.

3.8 Summary

In this chapter, we derive utility-maximising strategies for bidding in multiple, simultaneous second-price auctions. We first analyse the case where a single global bidder bids in all auctions, whereas all other bidders are local and bid in a single auction. For this setting, we find the counter-intuitive result that it is optimal to place non-zero bids in all auctions that sell the desired item, even when a bidder only requires a single item and derives no additional benefit from having more. Thus, a potential buyer can considerably benefit by participating in multiple auctions and employing an optimal bidding strategy. For most common valuation distributions, we show analytically that the problem of finding optimal bids reduces to two dimensions. This considerably simplifies the original optimisation problem and can thus be used in practice to compute the optimal bids for any number of auctions.

Furthermore, we investigate a setting with multiple global bidders by combining analytical solutions with a simulation approach. We find that a global bidder's strategy does not stabilise when only global bidders are present in the market, but only converges when there are local bidders as well. We argue, however, that real-world markets are likely to contain both local and global bidders. The converged results are then very similar to the setting with a single global bidder, and we find that a bidder benefits by bidding optimally in multiple auctions. For the more complex setting with multiple global bidders, the simulation can thus be used to find these bids for specific cases.

Finally, we compare the efficiency of a market with multiple simultaneous auctions with and without a global bidder. We show that, if the bidder can accurately predict the number of local bidders in each auction, the efficiency slightly increases. In contrast, if there is much uncertainty, the efficiency significantly decreases as the number of auctions increases due to the increased probability that a global bidder wins more than two items. These results show that the way in which the efficiency and thus social welfare is affected by a global bidder depends on the information available to a global bidder.

In sum, this chapter has studied a basic distributed allocation mechanism and in doing so has addressed requirement 1 outlined in Chapter 1. Furthermore, whilst we focused on deriving the strategies of the bidders within this distributed mechanism, we will change the focus to the design of protocols in the following chapters. Hence, this chapter provides us with a baseline efficiency of a distributed allocation mechanisms since the simultaneous auctions environment studied has not been engineered to achieve efficiency. Finally, in the context of MSNs, this chapter has shown that whilst a decentralised control regime can be achieved, it potentially comes at the cost of the efficiency of the whole system (i.e. the services provided by the selling sensors will not always end up with those sensors valuing it the most).

Chapter 4

Mechanisms with Constrained Capacity Suppliers

In the previous chapter, the focus was on the design of strategies for agents within a predefined decentralised protocol (namely the simultaneous auctions environment). In contrast, the focus of this chapter is on the design of the task allocation protocols. Such protocols define how tasks are allocated within a system of self-interested agents [Rosenschein and Zlotkin, 1994; Zlotkin and Rosenschein, 1996; Sandholm, 2003; Sarne and Kraus, 2005]. Specifically, we will continue studying distributed allocation within this context by designing and analysing a CDA mechanism. In so doing, we address both requirements 1 (distributed allocation) and 2 (constrained capacity) outlined in chapter 1 by achieving the following:

- Designing and analysing a centralised reverse auction mechanism that effectively deals with cases where sellers have a *constrained capacity*.
- Designing a decentralised CDA mechanism and analysing its performance using very simple strategies when the sellers have the same cost structure and capacity constraints as in the centralised mechanism.

We require both a centralised and decentralised version of the mechanism since the former guarantees efficiency within the system and provides us with a benchmark against which the performance of the latter can be measured.

The remainder of the chapter is organized as follows: Section 4.1 explains the general setting of task allocation in selfish MASs to which the research carried out in this chapter can be applied. In Section 4.2, we present the MSN context in which such a research problem arises and in Section 4.3 we detail the relevant related work. Section 4.4 describes the task allocation problem in more detail. Section 4.5 then develops our centralised auction mechanism, which is based around the VCG mechanism (see section 2.4), for the cost structure and limited capacity constraint of our domain and proves the economic and computational properties of our mechanism. In section 4.6, the decentralised CDA mechanism is then developed and analysed. We summarise the main contributions of this chapter in Section 4.7.

4.1 Introduction

In Chapter 3 we concentrated on studying the strategies that a buyer should adopt in a distributed market based on simultaneous auctions. We assumed that the sellers have a publicly known single unit capacity and bear no cost in providing their service. In this chapter, we remove the assumption that sellers have publicly known and equal capacities. Instead, we consider the case where they have finite production capacities which are privately known to them. We then consider the design of market mechanisms for the provision of services with these constrained capacity sellers. Furthermore, we deal with the case in which the *cost structure* of the sellers consists of a fixed overhead cost and a constant marginal cost. We believe that these traits (constrained capacity, fixed overhead cost and constant marginal cost) are typical of many real world applications such as electricity markets, job-shop scheduling and grid computing applications. For example, a power plant will typically have a fixed startup cost and a constant marginal cost of running the plant upto its maximum capacity [Hobbs et al., 2000]. The classic job shop scheduling problem consists of running periods composed of an initial machine set-up time (overhead cost) plus a cost per unit time (the marginal cost) and a finite capacity which these machines can run upto [Chen et al., 1998]. Finally, agents providing computational resources on the grid incur an overhead cost (computational cost of setting up the agent managing the resource on the machine) and marginal costs as they accept tasks upto the limit that their machines can support [Wolski et al., 2001].

In general, there are two broad classes of market mechanisms that can be considered when dealing with such task allocation problems. The first class, the reverse auction, involves a centralised mechanism in which sellers report their values to a centre (that has already aggregated the demand from the buyers) which then decides on the optimal allocation and the payments. The most popular such mechanism is the VCG. Its popularity arises from two attractive economic properties: it is allocatively efficient and it is individually rational (as defined in chapter 2). Unfortunately, in our case, the finite capacities of the sellers and the particular cost structure of our problem mean that the VCG no longer preserves these desirable economic properties. Thus, we need to extend the VCG mechanism in order to restore them. Such modification is important because we wish to guarantee that we find the cheapest providers and we want to ensure that participants willingly join the system. Here, we achieve these dual objectives by allowing agents to report on the triples (fixed cost, unit cost, and capacity) that characterise their types and via the use of a novel penalty scheme (detailed in section 4.5). We prove that the ensuing mechanism is strategyproof and robust to sellers being uncertain about their production capacity[1]. Furthermore, we show that the mechanism is computationally tractable since the optimal allocation can be computed in pseudo-polynomial time via the use of a dynamic programming solution.

However, a potential drawback of our modified VCG mechanism (indeed of all the mechanisms in this class) is that it is inherently centralised (as we discussed in chapter 1). That is, the task allocation is computed by a single entity, the auctioneer, who does so by collecting all the private information about the costs and capacities from the various agents. Now, in some cases, this is not a problem and the optimality of the mechanism is the over-riding concern. However, in other cases, issues such as robustness to a single point of failure and scaleability are more important and this gives rise to the desire for decentralised mechanisms (see chapter 1). Thus to cope with this situation, we also consider the decentralised CDA [Smith, 1962; Friedman and Rust, 1992]. In this protocol, buyers and sellers continuously submit bids (an offer to buy at price p_b) and asks (an offer to sell at price p_a) respectively (which are listed on a billboard) and the market clears (i.e. a transaction occurs) whenever the bid of a buyer matches the ask of a seller (i.e. when $p_b \geq p_a$). Such an auction is decentralised in that the allocation of the tasks is not computed by any single agent, but rather emerges out of the interactions of the agents in the protocol[2]. Nevertheless, despite this decentralisation, CDAs still produce solutions that are very close to the optimal, even when the participants adopt very simple strategies[3].

However, most work on CDAs assumes a cost structure that consists of a fixed marginal cost for each unit supplied and no start-up cost. This choice of cost structure is quite natural in macro-economic models and it results both in an equilibrium market price (a unique price at which buyers and sellers agree to trade) for the commodity and in efficient allocations [MasColell et al., 1995]. Unfortunately, the particular cost structure of our domain implies that no such equilibrium exists. This is due to the fact that the average unit cost of producing lower quantities is greater than that when producing larger quantities as a result of the start-up cost (this is akin to models where there are economies of scale in which the start-up cost is shared over a greater product run [MasColell et al., 1995]). The presence of a capacity constraint further complicates matters since, in general, a single seller will not be able to fully satisfy the total demand. Furthermore, since we are developing a protocol for task allocation, we consider buyers with inelastic demand (i.e. buyers do not vary their demand according to price) which, in turn, means that the CDA is focused on finding the cheapest set of seller(s) given an exact demand

[1] In certain scenarios, sellers may be uncertain about their capacity and would only have a best estimate of that capacity (e.g. in power generation scenarios a wind farm's capacity will depend on the strength of the wind and in a job-shop scheduling context the capacity of a machine might degrade stochastically over time).

[2] Even the seemingly centralised billboard in the CDA can be implemented using a broadcast communication protocol that mimics the typical "shouts" in the original trading pit [Friedman and Rust, 1992].

[3] In this context, a strategy is simply a method of generating a bid or an ask given the observed current market conditions (see Chapter 2). In CDAs, it has been shown that a strategy that randomly generates bids/asks between a set lower and upper bound can be extremely efficient (both for the individual participant and in terms of the effectiveness of the overall market). Such strategies are known as zero-intelligence (ZI) strategies [Gode and Sunder, 1993].

from the buyers[4]. Given these points, we need to modify the standard CDA mechanism by designing suitable clearing rules and constraining the type of offers allowed in the market in order to deal with the aforementioned issues. We then assess the allocative efficiency of our market mechanism using the same methodology as was employed by Gode and Sunder in their seminal study of the standard CDA mechanism[5] [Gode and Sunder, 1993]. This assessment shows that the allocative efficiency of our CDA protocol is fairly high (with an average value of 83% in the scenario we consider) and that our ZI2 agents are always profitable (this condition is broadly equivalent to the individual rationality condition of the centralised mechanism).

These two mechanisms have been developed because they represent complementary task allocation mechanisms for the same domain (i.e. where the sellers have finite production capacity and the cost structure we outline). Both mechanisms address requirement 2 in that they both deal with constrained capacity suppliers. However, while the extended VCG mechanism *guarantees* that the cheapest set of seller(s) is always found, it is centralised. In contrast, the mechanism derived from the CDA is decentralised (thereby addressing requirement 1), but it does not guarantee to find the cheapest set of sellers. Thus, in some cases, the centralised mechanism is more appropriate because efficiency cannot be compromised (e.g. when the costs involved are high or the set of agents participating in the market is low, thereby abating the disadvantages of centralisation). However, when decentralisation is a more desirable aspect (such as in cases where there are large numbers of agents or when robustness to failure is important), the CDA-based solution is more appropriate. Furthermore, our experimental results quantify the loss in efficiency that occurs when the decentralised system is implemented instead of its centralised counterpart (an average of 17% in the case we study). It is important to note that under both mechanisms, the sellers, though competitive, are profitable and they are hence always incentivised to participate in our systems.

In the next section, we detail the particular part of the scenario we introduced in Section 1.2 which gives rise to the challenges addressed in this chapter.

4.2 Constrained Capacity Suppliers within the MSN Scenario

The scenario in this chapter demonstrates how the cost structure (consisting of a fixed overhead cost and a constant marginal cost) and finite production capacities arise in the MSN scenario (as highlighted in figure 4.1). To this end, consider the sensors in the region of interest of the environment that can be tasked to gather data by another sensor lying outside of the region. Since the sensor requesting the task will need sufficient data in order to study a trend, we will let

[4]Inelastic demand also ensures a fair comparison with the centralised case. This is because allowing for elastic demand will result in an allocation which satisfies a demand defined by the demand and supply curves, rather than a prior demand that has been made by the buyers (which would occur with inelastic demand). It also allows us to characterise the cost of decentralising the market-based mechanism in terms of its efficiency loss.

[5]While their study employed ZI agents that operate purely on price, in our case, the sellers have to provide both a price and quantity vector. Thus we modify the ZI strategy to a ZI2 strategy that applies the same basic idea to both price and quantity.

the smallest unit of the tasks be discrete intervals of time over which the task is carried out (e.g. temperature data over 3 hours, visual data over an interval of 5 minutes). Now, when a sensor performs a sensing task, the following costs will arise:

- **Start-up cost**: This is the cost (which is measured in terms of energy loss) of powering up the sensor.
- **Marginal cost**: This additional cost per time interval is borne by the sensor as it spends energy sensing the environment.

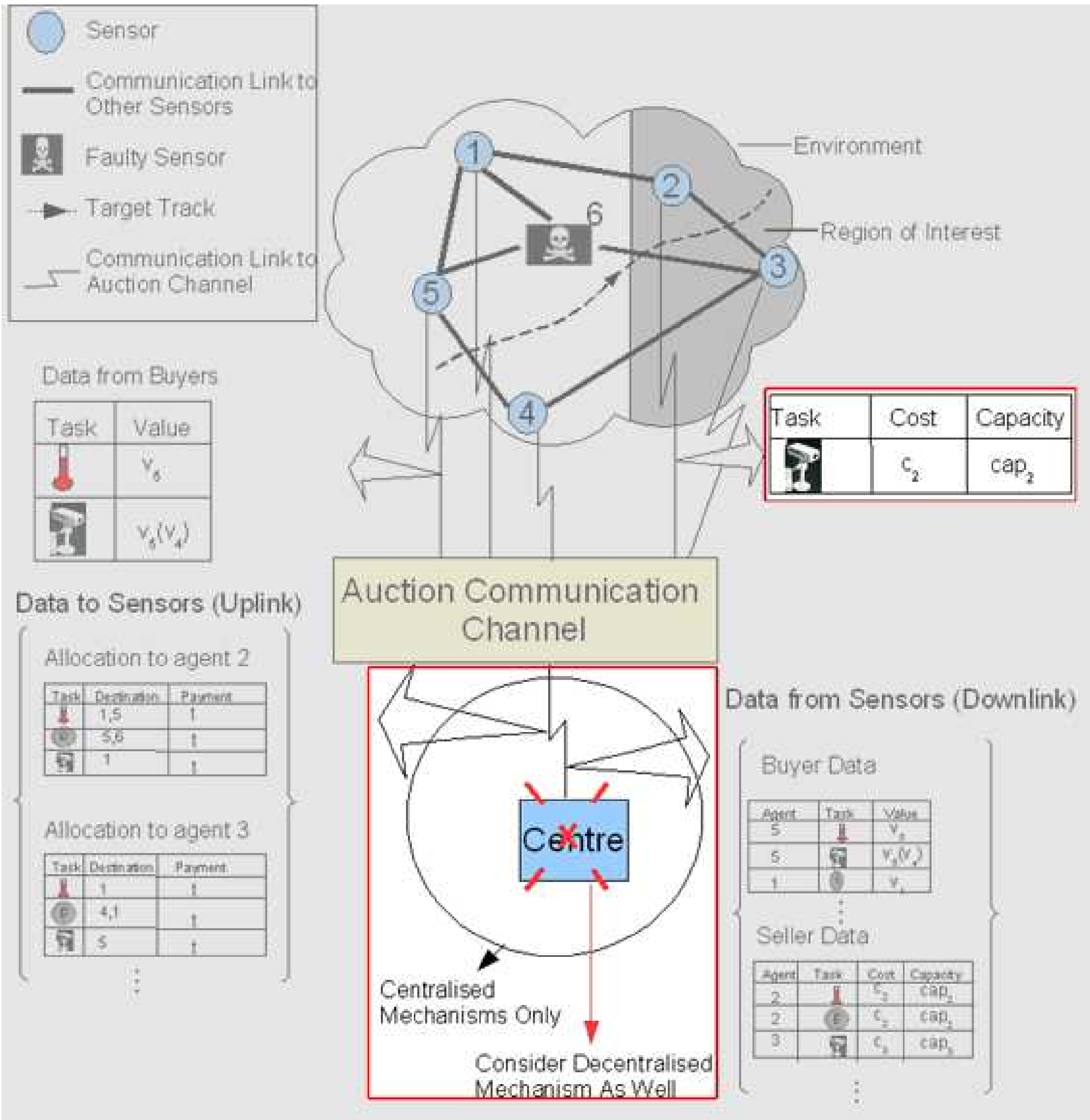


FIGURE 4.1: Figure of the MSN scenario highlighting the constrained capacity of suppliers and the centralised and decentralised mechanism considered within this chapter.

Furthermore, as a result of the limited bandwidth and power available to the sensor, it can only do a certain maximum number of tasks at any one time, thereby resulting in a capacity constraint. This leads us to the particular cost structure studied within this chapter.

Moreover, in this chapter, we again study a distributed mechanism. As we have argued in section 3.2, a distributed mechanism may sometimes be required in the MSN scenario especially if a trusted centre does not exist (as highlighted in figure 4.1). However, as opposed to chapter 3, the auction bandwidth in the scenario we consider here is not severely constrained. This is justifiable in cases where the data transmitted during an auction (bids and prices) requires much less bandwidth than the sensed data (such as visual or audio data). We therefore analyse the CDA where the allocation is calculated in a distributed fashion as the buyers and sellers continuously submit bids and asks.

Having thus described our scenario, we now provide an overview of related work on the two market mechanisms considered in this chapter, with particular focus on the cost structure we consider here.

4.3 Related Work

The VCG mechanism and its various extensions have been used in a variety of computer systems for task allocation situations. The two broad issues that have been investigated are the economic and computational properties of these mechanisms under various scenarios (see chapters 1 and 2 for an in-depth discussion). Most solutions in this area consider standard demand functions (not our cost structure) in order to derive approximate solutions to the problem or to find instances where these can be solved exactly in polynomial time [Rothkopf et al., 1998; deVries and Vohra, 2003; Fujishima et al., 1999].

However, recently, there has been increasing interest on the economic and computational properties of mechanisms using non-standard cost functions. In particular, a decreasing marginal cost structure has been considered in [Kothari et al., 2003] and a polynomially solvable, approximately strategyproof and approximately efficient (i.e. solutions which are within a bound of the optimal) auction mechanism has been devised. In addition, more general piece-wise linear continuous curves have been considered in [Eso et al., 2001], but the incentives for truthful bidding were not taken into account. Furthermore, Sandholm [2002b] and Giovannucci et al. [2004] have investigated more realistic cost curves (such as those related to volume-quantity discounts) whereas in multi-attribute bidding [Bichler and Kalagnanam, 2005] has been considered. However, none of these approaches would work for the cost structure of our domain since they do not consider both the economic and computational properties of problems with overhead cost, constant marginal cost and limited capacity simultaneously. Furthermore, unlike our work, they do not derive an efficient, strategyproof and individually-rational solution or compare it with a decentralised auction. Also, they do not consider the problem of suppliers not fulfilling their commitment. This latter problem is studied in [Dash et al., 2004] (Chapter 6) and [Porter et al., 2002]. However, the mechanism in [Porter et al., 2002] considers success and failure as a binary variable and thus does not try to incentivise agents to produce upto their maximum if ever they cannot fulfil their commitment. In [Dash et al., 2004], both the producers and consumers report over the success of a transaction and thus their mechanism is more appropriate in an iterated marketplace where the consumers can form an opinion about the success rate of each producer. As a result, in their case, the consumers bear the risk of correctly evaluating the success rate of a producer, unlike in our mechanism where it is upto the producers to correctly estimate their capacities.

The double auction class of market mechanism consists fundamentally of two categories: the clearing-house and the CDA. The former involves all bids and asks being submitted to an auctioneer and the market being cleared periodically by that auctioneer (who calculates the allocation) [Friedman and Rust, 1992]. In contrast, the latter clears continuously, with the competition in the market deciding the allocation rather than an auctioneer[Friedman and Rust, 1992]. In this context, one particularly relevant application of the double auction is by [Nicolaisen et al., 2001] in a wholesale electricity market. Specifically, they use a clearing-house double auction with discriminatory pricing. Now, while they do not look at the complexity involved with a cost structure, they do describe a market mechanism for resource allocation. In particular, the agents populating their markets adopt a sophisticated bidding behaviour (a modified Roth-Erev Reinforcement Learning algorithm [Roth and Erev, 1995]), and they evaluate the efficiency of their mechanism using such strategies. Other relevant works on the double auction include that by [McCabe et al., 1992] on the design of a clearing-house, and [Xia et al., 2004] on solving combinatorial double auction mechanisms. However, none of these mechanisms are decentralised since they involve an auctioneer that computes the allocation and prices.

Speaking more generally, most research on the CDA has been on the structure and behaviour of the mechanism. Indeed, the initial stimulation for this work comes from the field of experimental economics where experiments with human volunteers showed that small groups of traders could quickly find the equilibrium price in simulated single commodity markets [Smith, 1962; Gode and Sunder, 1993]. In line with this seminal work, many researchers then extended these simple trading strategies to generate sophisticated software agents that are capable of observing the trading behaviour of other agents in order to learn the market equilibrium price of a commodity, and thus trade more efficiently [Tesauro and Bredin, 2002; Gjerstad and Dickhaut, 1998; He et al., 2003; Vytelingum et al., 2004]. However, in all of this work, the existence of the market equilibrium at which both buyers and sellers seek to trade is a consequence of the assumption of a cost structure with an increasing marginal cost and no startup cost. Unfortunately, the cost structure of our domain destroys this market equilibrium and thus the close to optimal efficiency usually obtained by CDAs cannot be guaranteed. Specifically, this is because the different startup costs and the inelastic demand mean that a single price on which buyers and sellers agree to trade cannot be reached. To remedy this, we develop a variant of the CDA that is still reasonably efficient, but that can deal with the specific cost structure and capacity constraint in our domain.

4.4 The Allocation Problem

We now discuss in more detail the problem structure that we consider in the remainder of this chapter. The system that we wish to control consists of a set $\mathcal{J} = \{1, \ldots, n\}$ of sellers of a resource and a number of consumers with total demand D. Each seller, $j \in \mathcal{J}$, is characterised by a maximum capacity that it can provide, cap_j, and a cost function, c_j. The cost function is defined as a combination of a fixed price, fp_j, payable for any amount of production and a separate per unit price, up_j:

$$c_j = \begin{cases} 0 & \text{if } k_j = 0, \\ fp_j + k_j up_j & \text{if } 0 < k_j \leq cap_j \end{cases} \tag{4.1}$$

where k_j is the quantity of production allocated to seller j. Thus, an allocation vector $K \in \mathcal{K}$ is one in which each agent j is asked to supply a quantity k_j. We assume that both the demand and the details of the cost function are private information of the producers (also referred to as suppliers or sellers) since they represent distinct self-interested stakeholders. Given this, the overall aim of the system is to satisfy the total demand by allocating production between the different producers. Here, we assume that the resource is bought and sold in small indivisible units (as is common in most billing systems) and thus $k_j \in \mathbb{N}$.

As the designer of the whole system, we are interested in ensuring that the overall allocation, K^*, of the resource under consideration is optimum in the sense that it minimises the total cost of production. In this case, it is an optimisation problem where we minimise the sum of the individual production costs, whilst satisfying the total demand, $\sum_{j \in \mathcal{J}} k_j = D$, and the capacity constraints of each individual producer:

$$K^* = \arg \min_{K \in \mathcal{K}} \sum_j (\alpha_j fp_j + up_j k_j) \tag{4.2}$$

such that $0 \leq k_j \leq cap_j$ and where:

$$\alpha_j = \begin{cases} 0 \text{ if } k_j = 0 \\ 1 \text{ otherwise.} \end{cases}$$

The problem as described here is somewhat similar to two standard problems from the literature of operational research and scheduling; specifically the knapsack problem [Martello and Toth, 1990] and the capacitated lot-size problem [Bitran and Yanasse, 1982]. Comparing this problem to the knapsack problem, we note that we can consider each supplier to be an item to be fitted into a knapsack. The size and value of each of these items is represented by the number of units of production allocated to this supplier and the cost of producing this allocation. Unlike the standard knapsack problem, where we seek to maximise the value of items without exceeding the size of the knapsack, our goal here is to exactly fill the knapsack (i.e. satisfy demand) whilst minimising the value of items placed inside (i.e. minimise the production costs). Although we can place fractional items within the knapsack, the size of these items is restricted to integer units of production and the corresponding value of the item is given by the cost structure shown in equation 4.1.

Comparing to the standard capacitated lot-size problem, which attempts to schedule the production of a single producer over a number of days to meet a specific daily demand, we are attempting to schedule production over a number of different producers to satisfy an aggregate demand. Despite this difference, both problems share a similar cost structure, most specifically the combination of a fixed and per unit cost, and most importantly, both models share the concept of producers who have a constrained production capacity. We could thus adapt algorithms developed for the capacitated lot-size problem to our problem. However, in this chapter the goal is to show that the problem can be solved in a computationally efficient manner rather than solve the problem in the most computationally efficient manner.

Now, both the knapsack and the capacitated lot-size problems have been shown to be $\mathcal{NP}$-*hard* [Florian et al., 1980; Garey and Johnson, 1979]. However both can be solved in pseudo-polynomial time using a dynamic programming approach [Garey and Johnson, 1979] and we use this fact to present a suitable implementation of this technique for our specific problem in section 4.5.3.

Given this problem description, in the following sections we describe our two task allocation mechanisms, starting with the centralised one.

4.5 The Centralised Mechanism

We build upon the standard VCG mechanism since this has a number of desirable economic properties with respect to task allocation. Specifically, it is *efficient*, incentivises the agents to reveal their costs truthfully to the auctioneer in *dominant strategy* and guarantees a non-negative utility to the participating agents (see chapter 2 for a more detailed description of the VCG mechanism).

The standard VCG mechanism for task allocation represents the producers as agents participating in a reverse auction to satisfy the demand of the auctioneer. The agents submit their type, θ_j, in sealed bids to the auctioneer. Given these bids, the auctioneer finds the efficient allocation and then calculates the payments or transfers for each agent. It is this transfer scheme that results in the agents having truthful reporting as a dominant strategy.

However, there are two key differences between our setting and that of a standard VCG mechanism. Firstly, each agent's type has three dimensions that characterises its cost function instead of the usual one. Specifically, these dimensions are the fixed price, fp_j, the unit cost, up_j, and the capacity, cap_j i.e. $\theta_j = (fp_j, up_j, cap_j)$. Secondly, the capacity of the agent does not directly affect the cost of supplying an allocated quantity of a resource, but rather puts a limit on the amount that it can supply. This differs from the standard setting where an agent's type directly affects its cost.

To deal with this, the VCG needs to be extended in two ways. The first change is to have agents report the defining characteristics of their cost functions rather than a single cost price. The second change is a penalty scheme that incentivises the agents to report truthfully on their capacities[6]. Given this, we present the mechanism as a two-part scheme which is a transfer scheme and a penalty scheme (in sections 4.5.1 and 4.5.2). This two-part mechanism is presented for explanatory purposes only since the revelation principle (see Chapter 2) tells us that we can certainly find the equivalent one-stage scheme which will incentivise truthtelling (see section 4.5.3).

4.5.1 The Transfer Scheme

The problem at hand is then to determine the optimal allocation K^* (i.e. the one that minimises the total cost of production), while satisfying the demand $\sum k_j = D$. If the agents are incentivised to report truthfully, then the auctioneer can just take their reports and solve the optimization problem introduced in section 4.4. More generally, however, if agents report $\widehat{\theta}_j = \left(\widehat{fp_j}, \widehat{up_j}, \widehat{cap_j}\right)$, the auctioneer then solves:

$$\widehat{K}^* = \arg\min_{K \in \mathcal{K}} \sum_j \left(\alpha_j \widehat{fp_j} + \widehat{up_j} x_j\right) \tag{4.3}$$

such that $0 \leq k_j \leq \widehat{cap_j}$.

Hence, comparing equations 4.2 and 4.3, in order to achieve an efficient allocation we are left with the problem of incentivising the agents to report truthfully. If we assume rational self-interested agents, then this implies that they should maximise their own utility when reporting truthfully (otherwise they will lie!). Like most work in this area, we consider the case that the agents have a quasi-linear utility function (as defined in chapter 2).

The standard VCG mechanism achieves truth-telling by aligning the goal of each agent with that of the mechanism designer via the use of the transfer part of the mechanism (see chapter 2 for more details). It imposes a transfer on the agent which is equivalent to its marginal contribution to the society. Now, applying this insight to our multi-dimensional type domain, we advocate

[6]We should note here that the second difference does not result in interdependent valuations (as discussed in Chapter 1). While the capacities of each agent do affect the allocation of other agents (the cheapest agent will determine how much the remaining agents will obtain via its capacity), it only does so in an indirect way. Therefore, we can still aim to achieve an efficient mechanism despite the multi-dimensionality of the types since we are firmly in the realm of private values [Krishna, 2002].

TABLE 4.1: A set of three producers bidding to satisfy a demand of 200 units.

Sellers

	S_1	S_2	S_3
Capacity	100	150	175
Fixed Price	100	200	120
Unit Price	1.5	1	2

the following transfer scheme in which the agents report on all three dimensions of their types (i.e. on $\theta_j = (fp_j, up_j, cap_j)$) :

$$t_j = \Big[\min_{\substack{K \in \mathcal{K} \\ k_j \leq cap_j}} \sum_{l \in \mathcal{J} \setminus j} \Big(\alpha \widehat{fp_l} + \widehat{up_l} k_l \Big) \Big] - \Big[\sum_{l \in \mathcal{J} \setminus j} \Big(\alpha^* \widehat{fp_l} + \widehat{up_l} \widehat{k_l}^* \Big) \Big]. \tag{4.4}$$

where $\widehat{k_l}^*$ is the allocation to agent l in the optimal allocation, $\widehat{K}^*$, calculated with the reports of all the agents.

In more detail, the transfer, t_j, is the payment that agent j receives from the auctioneer for providing an allocation k_j. The transfer scheme above, as in the VCG mechanism, consists of two parts. The first calculates the total cost of the optimal allocation if agent j were not included in the set of suppliers. In the second part, first the optimal allocation with agent j is found, and then the total cost of this allocation is calculated minus the cost of this allocation to agent j. Thus, the payment that j receives is its marginal contribution to reducing the total cost of the optimal allocation. It can be observed that j will always receive a non-negative payment since the addition of a seller will only decrease the cost of the optimal allocation.

We now present an example to show why this extension of the VCG mechanism is not the only change that needs to be applied so as to incentivize the agents to report truthfully. Consider a set of sellers $(1, 2, 3)$ with different types who are participating in a reverse auction to fulfill a demand of 200 units (i.e. $D = 200$). The producers' types, (i.e. $\theta_j = (cap_j, fp_j, up_j)$), are depicted in table 4.1. They report their types to the auctioneer which then calculates the transfers according to equations 4.3 and 4.4.

Let us suppose for now that the capacity cap_j of the agents are known by the auctioneer. Then, implementing our mechanism with the transfer described by equation 4.4, the auctioneer first chooses the optimal allocation. In this case, it would be S_2 producing 150 units and S_1 producing 50 units (i.e $K = \{50, 150, 0\}$) thereby giving a total cost of 525 to the system. The transfers would then be 220 to S_1, 395 to S_2 and 0 to S_3 (i.e. $\mathbf{t} = \{220, 395, 0\}$). However, given this scheme, S_3 has an incentive to lie about its capacity and give a capacity greater than 200 (i.e. $\widehat{cap_3} \geq 200$). It would then be allocated to produce the whole demand and would be paid 525 to do so. However, as its true capacity is only 175 units, demand will not be satisfied.

Thus from the above example, we can observe that an agent has an incentive to report a higher capacity than it actually has. An agent, however, has no incentive to report a lower capacity. This is because the utility derived by an agent is equal to its marginal contribution to the society. Now, if an agent reports a capacity lower than its actual one and this misreport has an effect on the optimal allocation (i.e the capacity it reports is lower than the allocation it would have got under an optimal allocation), then it increases the total cost to the society since the minimisation in equation 4.3 would have tighter constraints. This would mean that the marginal contribution of the agent to decreasing the total cost in the society is less and hence the agent would derive a lower utility. We thus only need to worry about agents reporting a higher capacity than they actually have. We therefore impose a penalty scheme that incentivises agents to report truthfully about their capacity. In a standard VCG, such a penalty scheme does not exist since it is assumed that the producers have unlimited capacity. Furthermore, a penalty scheme imposed after the agents have supplied their allocations is the only way we can incentivise agents to report truthfully about their capacity. This is because the auctioneer will only know whether an agent has overstated its capacity if ever that agent has been allocated to produce over its true capacity (but under its declared one) after the agent has supplied its allocation.

4.5.2 The Penalty Scheme

We wish to penalise agents that report a higher capacity than they have. However, we are not concerned with untruthful reporting if this does not affect the resulting efficient allocation. This is because such agents will not derive a higher utility if their untruthful reporting has not affected the efficient allocation. Thus, we will call agents whose reported capacities affect the optimal outcome *active agents.*

In order to know whether the active agents have truthfully reported their capacity, we require a post-production stage that checks how much they actually produced. We shall assume that if an agent is asked to supply a certain amount $\widehat{k_j}^*$, and actually produces only $\overline{k_j}$, ($\overline{k_j} < \widehat{k_j}^*$), then the capacity of that agent is $\overline{k_j}$. We shall see that given the penalty we design, this assumption is satisfied with rational agents. It is only in the case of malicious agents who want to increase the cost to the system with no consideration to their own utility for which the following penalty scheme would not work.

In more detail, we impose the following penalty, p_j, if the agent does not supply the amount that it was required to supply under the optimal allocation (i.e. if $\overline{k_j} < k_j^*$):

$$p_j = t_j(k_j \leq \widehat{cap_j}) - t_j(k_j \leq \overline{k_j}) + \delta \tag{4.5}$$

where $t_i(x_j \leq \widehat{cap_j})$ is the transfer in equation 4.4 computed with the constraint $k_j \leq \hat{c}_j$ and $t_j(k_j \leq \overline{k_j})$ is the one computed with the constraint $k_j \leq \overline{k_j}$.

This penalty scheme, which is a transfer of money from the agent to the auctioneer, consists of three parts. The first is the transfer that occurs with the reported capacity $\hat{c}_j$. The second part is the transfer that would have resulted if the agent had reported its capacity as the amount that it has successfully supplied. This penalty scheme thus only penalises agents in the case where their misreported capacity has changed the allocation of supply. The third part is the one that ensures that the utility an agent derives from misreporting its capacity is strictly lower than when it tells the truth (i.e it is then a strongly dominant strategy for the agent to report its truthful capacity).

It should also be noted that though this penalty scheme has been developed for the case of agents misreporting their capacity, it would also penalise agents that have not produced the specified amount due to other reasons. This penalty scheme thus puts the onus on the agents to provide an accurate report of the amount they can produce. The δ can thus be set by the mechanism designer depending on how critical it is to meet demand. The more critical the requirement, the higher δ should be set. Evidently, this sacrifices efficiency (the agents report a lower capacity than their most likely capacity) for robustness. Another attractive aspect of this penalty scheme is that if ever an agent realises after the allocation that it cannot produce the amount assigned to it, it would still produce till its limit so as to reduce the ultimate penalty.

Thus, in our example in table 4.1, if agent S_3 reported $\widehat{cap_3} = 200$, it would be penalised $525+\delta$ (from equation 4.5). As a result, the agent does not profit by lying. In the case of the two other agents, S_1 and S_2, misreporting their types, they incur a loss in utility equal to δ.

4.5.3 The Equivalent One-Stage Mechanism

We can amalgamate the two-part mechanism presented in sub-sections 4.5.1 and 4.5.2 into an equivalent one-stage mechanism:

1. First the seller agents, S_j, provide reports of their types $\widehat{\theta}_j = (\widehat{fp_j}, \widehat{up_j}, \widehat{cap_j})$ to the center.
2. The center, having gathered total demand from the buyer agents, solves equation 4.3 and assigns production to the agents according to the optimal allocation vector $\widehat{K}^*$.
3. The center then provides the overall payment r_j to the agents once they have produced their allocation:

$$\begin{aligned} r_j &= t_j - p_j \\ &= t_j(k_j \leq \overline{k_j}) - \delta\beta_j \end{aligned} \tag{4.6}$$

   where β_j is an indicator function which is equal to 1 when $\overline{k_j} < \widehat{k_j}^*$ and 0 otherwise.

4.5.4 Properties of the Mechanism

We now prove the properties of our mechanism.

Proposition 4.1. *The mechanism is strategyproof (as defined in chapter 2).*

Proof. Here, we need to prove that truthful reporting is a dominant strategy for the agents given the transfer and penalty schemes in our mechanism. We first consider the case that the agent has not over-reported its capacity. Then its strategy is to report $\widehat{\theta}$ so as to maximise its utility:

$$\begin{aligned}
\widehat{\theta}_j = (\widehat{up_j}, \widehat{fp_j}, \widehat{cap}_j) &= \arg\max_{\widehat{\theta}_j \in \Theta_j} (up_j(\widehat{\theta}_j), K) \\
&= \arg\max_{\widehat{\theta}_j \in \Theta_j} \Big[\Big(\widehat{\alpha_j}^* (\widehat{fp_j} - fp_j) + (\widehat{up_j} - up_j) \widehat{k_j}^* \Big) - \min_{\substack{K \\ k_j \leq \widehat{cap_j}}} \sum_{j \in \mathcal{J}} \Big(\alpha_j \widehat{fp_j} + \widehat{up_j} k_j \Big) \\
&\quad + \min_{\substack{K \\ k_j \leq c\hat{a}p_j}} \sum_{j \in \mathcal{J} \setminus i} \Big(\alpha_j \widehat{fp_j} + \widehat{up_j} k_j \Big) \Big] \\
&= \arg\max_{\widehat{\theta}_j \in \Theta_j} \Big[\Big(\widehat{\alpha_j}^* (\widehat{fp_j} - fp_j) + (\widehat{up_j} - up_j) \widehat{k_j}^* \Big) - \min_{\substack{K \\ k_j \leq \widehat{c}_j}} \sum_{j \in \mathcal{J}} \Big(\alpha_j \widehat{fp_j} + \widehat{u}_j k_j \Big) \Big]
\end{aligned}$$

The first part of the maximisation is the gain/loss that an agent makes by misreporting its type, whereas the second part is the effect misreporting has on the allocation and the global cost. Hence any misreport on its type is cancelled out by the effect on the global cost. The important point to note here is that the minimisation is not carried out by the agent, but by a centre that is only aware of $\widehat{\theta}_j$. Hence, in order to maximise the term in $[.]$ above, an agent should report $\widehat{\theta}_j = (fp_j, up_j, \widehat{cap_j})$ (i.e. truthtelling in (fp_j, up_j) is a weakly dominant strategy). Thus, we have proved that the mechanism is strategyproof in (fp_j, up_j). Furthermore, we know that an agent will not report a lower capacity. Now, we prove that under the penalty scheme, the agent will not report a capacity higher than its actual one. The utility of an agent i, given that it has reported a higher capacity, is the sum of its cost, transfer and penalty. We now prove that in the case of an active agent overreporting its capacity is a strongly dominated strategy. From equations 2.6 and 4.6, the utility of an agent would then be:

$$\begin{aligned}
u_j(.) &= \max_{\widehat{\theta}_j \in \Theta_j} \Big[(\widehat{\alpha_j}^* (\widehat{fp_j} - fp_j) + (\widehat{up_j} - up_j) \widehat{k_j}^*) - \delta\beta - \min_{\substack{K \\ k_j \leq \overline{k_j}}} \sum_{j \in \mathcal{J}} (\alpha_j \widehat{fp_j} + \widehat{up_j} k_j) \Big] \\
&< \max_{\widehat{\theta}_j \in \Theta_j} \Big[(\widehat{\alpha_j}^* (\widehat{fp_j} - fp_j) + (\widehat{up_j} - up_j) \widehat{k_j}^*) - \min_{\substack{K \\ k_j \leq \widehat{c}_j \\ \widehat{cap_j} = cap_j}} \sum_{j \in \mathcal{J}} (\alpha_j \widehat{fp_j} + \widehat{up_j} k_j) \Big]
\end{aligned}$$

Thus, together with the fact that an agent would not report a lower capacity (since such a report would mean that its resulting allocation is less or equal to the one when it reports truthfully), the above proves that an agent will always report its truthful capacity, cap_j. Hence we have that the agent always reports truthfully about its type, θ_j. □

Proposition 4.2. *The mechanism is efficient (as defined in chapter 2).*

This implies that the centre finds the outcome given by equation 4.2.

Proof. The above is a result of the strategyproofness of the mechanism. Since the goal of the centre is to achieve efficiency, then given truthful reports, the centre will achieve efficiency. □

Proposition 4.3. *The mechanism is individually-rational (as defined in chapter 2).*

We again assume that the utility an agent derives from not joining the mechanism is 0. Then, we need to prove that the utility an agent derives in the mechanism is always ≥ 0.

Proof. Given the strategyproofness of the mechanism, the utility of an agent is:

$$u_j(up_j, fp_j, cap_j) = -\min_{\substack{K \\ k_j \leq cap_j}} \sum_{j \in \mathcal{J}} (\alpha_j fp_j + up_j k_j) + \min_{\substack{K \\ k_j \leq cap_j}} \sum_{j \in \mathcal{J} \setminus i} (\alpha fp_j + up_j k_j)$$

The first minimisation is over a larger set than the second one. Thus:

$$\min_{\substack{K \\ k_j \leq cap_j}} \sum_{j \in \mathcal{J}} (\alpha_j fp_j + up_j k_j) \leq \min_{\substack{K \\ k_j \leq cap_j}} \sum_{j \in \mathcal{J} \setminus i} (\alpha fp_j + up_j k_j)$$

Hence, $u_j(up_j, fp_j, cap_j) \geq 0$.

□

Proposition 4.4. *The mechanism is robust to uncertainties about the capacity of agents.*

In this case, we impose less stringent information requirements on the agents when reporting their capacity. So far, we have considered the case where prior to revealing its type, an agent is aware of its capacity. However, we believe that this may not be always practical since the capacity of a supplier may depend on numerous external factors (as discussed in section 4.1). We therefore relax this requirement and consider the case where an agent is aware of only the probability distribution function (pdf) relating to its capacity. We next prove that the designer can, via the setting of δ, force the agent to either report safe values (i.e. the agent is nearly certain that it will produce at least this capacity) or more risky but potentially more profitable ones.

Proof. We start by looking at the expected utility of an agent given that the probability distribution function of its capacity, $f(cap_j)$, ranges from a lower bound $\underline{cap_j}$ to an upper bound $\overline{cap_j}$ (the associated cumulative density function is given by $F(cap_j)$) and that the agent reports a capacity of $\widehat{cap_j}$:

$$\begin{aligned} E[u_j(\widehat{cap}_j, fp_j, up_j)] =& E\Big[- \min_{\substack{K\\ k_i \le \widehat{cap}_i}} \sum_{i \in \mathcal{J}} (\alpha_i fp_i + up_i k_i) + \min_{\substack{K\\ k_i \le \hat{cap}_i}} \sum_{i \in \mathcal{J} \setminus j} (\alpha fp_i + up_i k_i) \\ & - \delta \beta_j \Big] \end{aligned}$$

$$\begin{aligned} =& - \int_{\underline{cap_j}}^{\widehat{cap_j}} \min_{\substack{K\\ k_i \le \widehat{cap}_i}} \sum_{i \in \mathcal{J}} (\alpha_i fp_i + up_i k_i)\, f(cap_j) dcap_j \\ & - \delta F(\widehat{cap_j}) + \min_{\substack{K\\ k_j \le \widehat{cap}_j}} \sum_{i \in \mathcal{J} \setminus j} (\alpha_i fp_i + up_i k_i) \end{aligned}$$

Now, let us analyse how the reports of the agents affect their utility. The safest report is the minimum report $\underline{cap_j}$. Reporting a higher capacity $\widehat{cap}_j$ would then yield a gain of:

$$\begin{aligned} \Delta E[u_j(cap_j, fp_j, up_j)] =& - \delta F(\widehat{cap_j}) + \Big[\min_{\substack{K\\ k_j \le \underline{cap_j}}} \sum_{j \in \mathcal{J}} (\alpha_j fp_j + up_j k_j) - \\ & \int_{\widehat{cap_j}}^{\overline{cap_j}} \min_{\substack{K\\ k_i \le \widehat{cap}_i}} \sum_{i \in \mathcal{J}} (\alpha_i fp_i + up_i k_i)\, f(cap_j) dcap_j \Big] \end{aligned} \tag{4.7}$$

The agents would then try to maximise the above gain given a certain δ. Thus, the setting of δ would then depend on how certain we want the agents to be about being able to satisfy their capacity. Hence, setting δ as:

$$\begin{aligned} \delta =& \Bigg[\min_{\substack{K\\ k_j \le \underline{cap_j}}} \sum_{j \in \mathcal{J}} (\alpha_j fp_j + up_j k_j) - \\ & \int_{\widehat{cap_j}}^{\overline{cap_j}} \min_{\substack{K\\ k_i \le \widehat{cap}_i}} \sum_{i \in \mathcal{J}} (\alpha_i fp_i + up_i k_i)\, f(cap_j) dcap_j \Bigg] \Big/ F(\widehat{cap_j}) \end{aligned} \tag{4.8}$$

results in no expected gain for the agent. In fact from equation 4.7, if we consider a fixed δ, then as $\widehat{c_j}$ increases, the part in [.] increases while $-\delta P(cap_j < \widehat{cap_j})$ decreases. Thus there is a $\widehat{cap_j}$ for a fixed δ that results in a maximum gain. We can therefore conclude that as δ increases, $\widehat{cap_j} \to \underline{cap_j}$ and as δ decreases, $\widehat{cap_j} \to \overline{cap_j}$. $\square$

```
Calculate initial row of matrix c
c[0,0] ← 0
for d = 1 to D do c[0,d] ← ∞
Loop through the total number of producers
for j = 1 to n do
    Loop through the total demand
    for d = 0 to D do
        c[j,d] ← c[j-1,d]
        Loop through the total capacity of producer i
        for k_j = 1 to min{d,cap_j} do
            Compare the previous result to the current
            result and select the minimum of the two
            c[j,d] ← min{c[j,d],c[j-1,d-k_j] + fp_j + k_j up_j}
Return the final result
return c[n,D]
```

FIGURE 4.2: Pseudo-code representing the dynamic programming solution to find the optimum centralised solution in pseudo-polynomial time.

The second part of the robustness is that even if the agent realises after reporting $\widehat{cap_j}$ that $cap_j < \widehat{k_j}^*$, it will still produce upto cap_j as a result of the payment and penalty scheme.

Proof. This is evident from the way the centre pays the agents. The agents get a higher utility with a higher production since the transfer depends on how much they produce, (i.e. $\overline{x}$, after the allocation) We have shown that reporting a higher capacity (upto the true capacity) is a weakly dominant strategy. Along with the penalty scheme, this can be viewed as producing as much as it can upto its optimal allocation is a weakly dominant strategy for any agent. □

Proposition 4.5. *The optimal task allocation to the agents can be computed exactly by the centre in pseudo-polynomial time.*

Proof. The centre can calculate the task allocation to the agents exactly using dynamic programming. Specifically, we wish to calculate $c[n, D]$ — the minimum total cost to satisfy a demand of D with access to n producers. This can be solved using the recursive expressions:

$$c[0,d] = \begin{cases} 0 & \text{if } d = 0 \\ \infty & \text{if } d > 0 \end{cases}$$

$$c[i,d] = \min_{k_j} \begin{cases} c[j-1,d] \\ c[j-1,d-k_j] + fp_j + k_j up_j \end{cases}$$

such that $0 < k_j \leq cap_j$. As the production allocated to each producer is in indivisible units, we can calculate $c[n, D]$ by evaluating all nD possible values. This results in an algorithm which operates in pseudo-polynomial time.

In particular, a simple algorithm for this solution is presented in figure 4.2. Here we calculate all the values of the array, $c[n, D]$, starting from the known case $c[0, 0] = 0$ and using the recursive expressions above to calculate subsequent values. A more efficient solution could perhaps be found using primal-dual algorithms [Papadimitriou and Steiglitz, 1982]. However for the size of problem tackled here, the above solution is extremely efficient. Moreover, the same approach can then be used to calculate the resulting task allocation to the agents. □

4.6 The Decentralised Mechanism

So far we have considered a centralised mechanism in order to deal with our task allocation problem. However, as discussed in section 4.1, we sometimes require a mechanism for task allocation in which there is no centre that governs the allocations. Therefore, in this section, we consider the CDA which is just such a decentralised mechanism.

Our task allocation problem involves multiple suppliers and multiple buyers, and the matching of the two is determined by the sellers and buyers that successfully transact with one another. As discussed in sections 4.1 and 4.3, the most common CDA format assumes buyers and sellers have an increasing marginal cost and no startup cost and that the offers in the trade are via price alone. However, in our case, the total production cost depends on both the startup cost and the number of units to be sold (given the marginal cost). In fact, since the startup cost is distributed over the sale quantity, the cost price is not fixed for different numbers of units sold. As a result, the supplier cannot firmly decide on an asking price (based on the production cost per unit or *cost price*) that would allow it to be profitable and to participate in the task allocation (by transacting with potential buyers). This is because the sale quantity cannot be known *a priori*. To overcome this, we assume that it is possible for the supplier to make a prediction about the amount of units it expects to sell (since exact demand can only be estimated)[7]. Now, in traditional cost settings, a supplier can start making bids for a low quantity and slowly ramp up its price so as to ensure it does not make a loss. However, in our setting, low quantities correspond to higher unit prices. Thus the supplier is faced with the problem that reducing its price may not guarantee that it transacts and in certain cases may lead to a loss (if a buyer specifies a demand such that the ask price becomes lower than the cost price). We therefore allow sellers to communicate the amount they wish to sell to the market via a multidimensional bid consisting of both quantity and price. We also specify in our clearing rules that a transaction only occurs when a buyer makes a bid for this amount.

[7] In fact, in CDA scenarios demand cannot be known even after the bids have been submitted [Cliff and Bruten, 1997]. This is why sellers try to predict the demand in order to be more profitable [He et al., 2003].

Given this background, a key objective for the decentralised mechanism is to be *individually rational* (as defined in chapter 2). In this case, this means ensuring the suppliers can be profitable in the market so that they are incentivised to enter it in the first place. Furthermore, while the mechanism has to be individually rational, our global objective is to achieve the most efficient outcome (task allocation) that we can. Now, as we discussed in section 4.4, this is equivalent to finding the allocation that minimises total cost. In a typical CDA mechanism, the optimal task allocation occurs when the total profit of all buyers and all sellers is maximised [Friedman and Rust, 1992] and this occurs when the combined cost of sellers is minimised on the sell side[8], as the sellers with the lowest cost would be successful.

However, given our additional constraints of limited capacity and a startup cost, the seller's strategic behaviour would be more complex than that of the buyer, since, as we mention before, it additionally has to strategise over the quantity it is expected to sell. In this context, we cannot achieve full efficiency because no agent has complete information about every other agent in the market (unlike in section 4.5 where the centre is aware of everyone's cost functions and capacities) and the sellers do not have increasing marginal costs which would guarantee an equilibrium price for trade [MasColell et al., 1995].

Given this, our aim is to design a protocol that achieves a level of efficiency that is reasonably close to the optimal solution given by our centralised mechanism. To do this, we now outline our protocol, and then go on to compare its performance with its centralised counterpart in terms of task allocation efficiency.

4.6.1 The Mechanism

The protocol we propose is a variant of the multi-unit CDA. Buyers and sellers can submit offers to buy and sell multiple units of the resource, respectively, and those orders are queued in an order book which is cleared continuously (with additional constraints as a result of buyers' inelastic demands). The protocol proceeds as follows:

- Buyer i submits an offer, $bid(q, p, i)$, to buy exactly q ($q \geq 1$) units of the good at the unit price p. The utility of buyer i for a quantity other than q is 0.
- Conversely, supplier S_j submits an offer, $ask(q, p, j)$, to sell a maximum of q ($q \geq 1$) units at unit price p.
- These bids and asks are queued in an *orderbook*, which is a publicly observable board listing all the bids and asks submitted to the market (see table 4.2). The bids in the order book are sorted in decreasing order of price and the asks are in increasing order (higher bids and lower asks are more likely to result in transactions).

[8] Sell side refers to the market from the sellers' perspective.

- The *clearing rule* in the market is as follows. Whenever a new bid or ask is submitted, an attempt is made at clearing the order book. The orderbook is cleared *whenever* a transaction can occur (that is, when the lowest asking price is higher than the highest bidding price *and* any bidding offer can be cleared completely and the bidding quantity for each offer is completely satisfied by the supply to be cleared). The transaction price is set at the bidding price which we experimentally find to result in the total market profits being equally divided between the sell side and the buy side[9] [Vytelingum et al., 2004].

Order Book	
Bids *(quantity, price, buyer)*	*Asks* *(quantity, price, seller)*
(30, 2.95, 2)	(60, 2.20, 3)
(40, 2.75, 5)	(25, 2.60, 1)
(30, 2.70, 1)	(40, 3.22, 2)
(24, 2.16, 3)	(100, 3.50, 5)
	(25, 3.69, 7)
	...

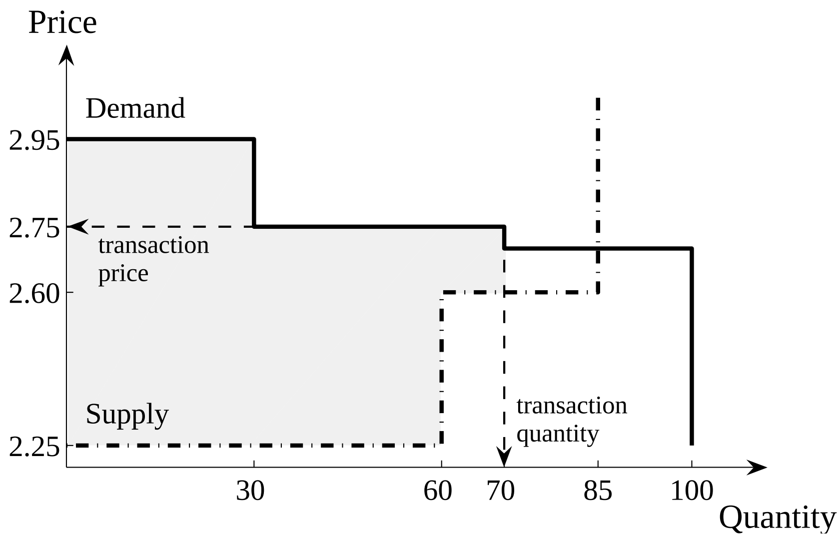
TABLE 4.2: Multi-unit CDA Order Book - before clearing

To further illustrate this process, we present a graphical representation of the clearing rule in figure 4.3. As can be seen, the offers queued in the orderbook are used to build demand and supply curves. All bids with a unit price lower than the lowest unit ask price and, similarly, all asks with a unit price higher than the highest unit bid price, cannot result in any transaction and are not represented in the figure. The transaction price and quantity are clearly shown in the figure (2.75 and 70 respectively), as the point where the demand curve crosses the supply curve under the additional constraint that bid offers are not divisible. At this transaction price, the total profit of all buyers and sellers that transact is maximised with all constraints specified by our protocols satisfied. The orderbook in table 4.2 can thus be cleared as shown in figure 4.3 resulting in the new orderbook given in table 4.3. The market clearing is then similar to solving an optimisation problem where the objective is to maximise the total profit of buyers and sellers that will transact given that cleared demand must be equal to cleared supply and no partial clearing of bid is allowed. [10]

Now in order to compare the efficiency of this protocol with that of the centralised mechanism, we assume that the buyers have high limit prices (this represents price inelasticity because buyers are willing to pay any price to acquire the goods and this is equal to an arbitrary maximum price that a bid or an ask can be submitted at). Furthermore, we adopt the approach of Gode and Sunder [1993] in employing a zero-intelligence strategy in order to find the underlying efficiency

[9]We chose this option because a mechanism where most of the profits in the market were distributed among sellers would be less appealing to buyers than one where a larger share of profits were distributed among buyers. Thus, with a similar preference among sellers (who will join a market where more profit is distributed among the sell side), a mechanism that equally distributes market profits among the buy and sell side is the rational preference for both buyers and sellers.

[10]We note that other clearing rules are also possible, for example to maximise the number of transactions or to maximise profits of the sellers only [Friedman and Rust, 1992]. However, the aim of a market mechanism is to maximise social welfare by maximising the total profit extracted in the market, and it is achieved through the simple ordering order books that publicly shows which buyers (with highest valuation of the goods) can transact with which suppliers (with the lowest ask prices).



(a)

Order Book	
Bids *(quantity, price, buyer)*	*Asks* *(quantity, price, seller)*
(30, 2.95, 2)	(60, 2.20, 3)
(40, 2.75, 5)	(25, 2.60, 1)
(30, 2.70, 1)	(40, 3.22, 2)
(24, 2.16, 3)	(100, 3.50, 5)
	(25, 3.69, 7)
	...

(b)

FIGURE 4.3: Panel (a) shows the demand and supply (curves) of the order book, with the shaded region representing allocations. Panel (b) points out the *clearable bids and asks* in the order book (shaded area in panel (a)).

Order Book	
Bids *(quantity, price, buyer)*	*Asks* *(quantity, price, seller)*
(30, 2.70, 1)	(15, 2.60, 1)
(24, 2.16, 3)	(40, 3.22, 2)
	(100, 3.50, 5)
	(25, 3.69, 7)

TABLE 4.3: Multi-unit CDA Order Book - after clearing

of our market. To this end, we next present the ZI2 that is tailored to the bidding structure of our CDA protocol, before we detail the actual evaluation.

4.6.2 The ZI2 Strategy

One of the principal concerns in developing a market mechanism is to ensure that it is efficient even when the participants adopt a simple strategic behaviour. The underlying intuition here is that by considering such behaviour, we are able to establish a lower bound on the efficiency of the mechanism and we can consider the extent to which the market mechanism itself affects the efficiency of the market. Thus, the ZI strategy is widely used for this purpose since it is not motivated by trading profit and effectively ignores the state of the market and past experience when forming a bid or an ask. It simply draws its offer price from a uniform distribution over a given range.

Since in our mechanism, the asks consist of price and quantity, we extend the ZI strategy to our *ZI2 strategy* that randomises over both price and quantity. As discussed earlier, any sophisticated strategy, on the sell side, would make some form of prediction on the number of units it is likely to sell as part of its price formation process (because information about the actual demand is not available and there is uncertainty as to whether the agent is more competitive than the other participating suppliers). Our ZI2 supplier j, instead, randomises over the expected transaction quantity to form a limit price ℓ_j which is used as in the original ZI strategy. Thus the ZI2 strategy is[11]:

For seller j,

$$\begin{aligned} \hat{q}_j &\sim \mathcal{U}(0, cap_j) \\ \ell_j &= (fp_j + \hat{q}_j up_j)/\hat{q}_j \\ p_j &\sim \mathcal{U}(\ell_j, max) \\ \textit{offer} &= ask(cap_j, p_j, j) \end{aligned} \tag{4.9}$$

For buyer i,

$$\begin{aligned} p_i &\sim \mathcal{U}(0, \ell_i) \\ \textit{offer} &= bid(q_i, p_i, i) \end{aligned} \tag{4.10}$$

Buyers are endowed with high limit prices at the beginning of the auction (because they have inelastic demand), while sellers are endowed with their cost functions and capacities (collectively referred to as the production function). Buyer i submits offers to buy the quantity q_i it requires at a unit price drawn from a uniform distribution ranging from 0 to its limit price ℓ_i (see equation 4.10). Conversely, seller j submits an ask between its limit price and max as per equation 4.9, where cap_j is its production capacity, fp_j is its startup cost, and up_j is its marginal cost.

[11] $X \sim \mathcal{U}(A, B)$ describes a discrete uniform distribution between A and B, with steps of 0.01.

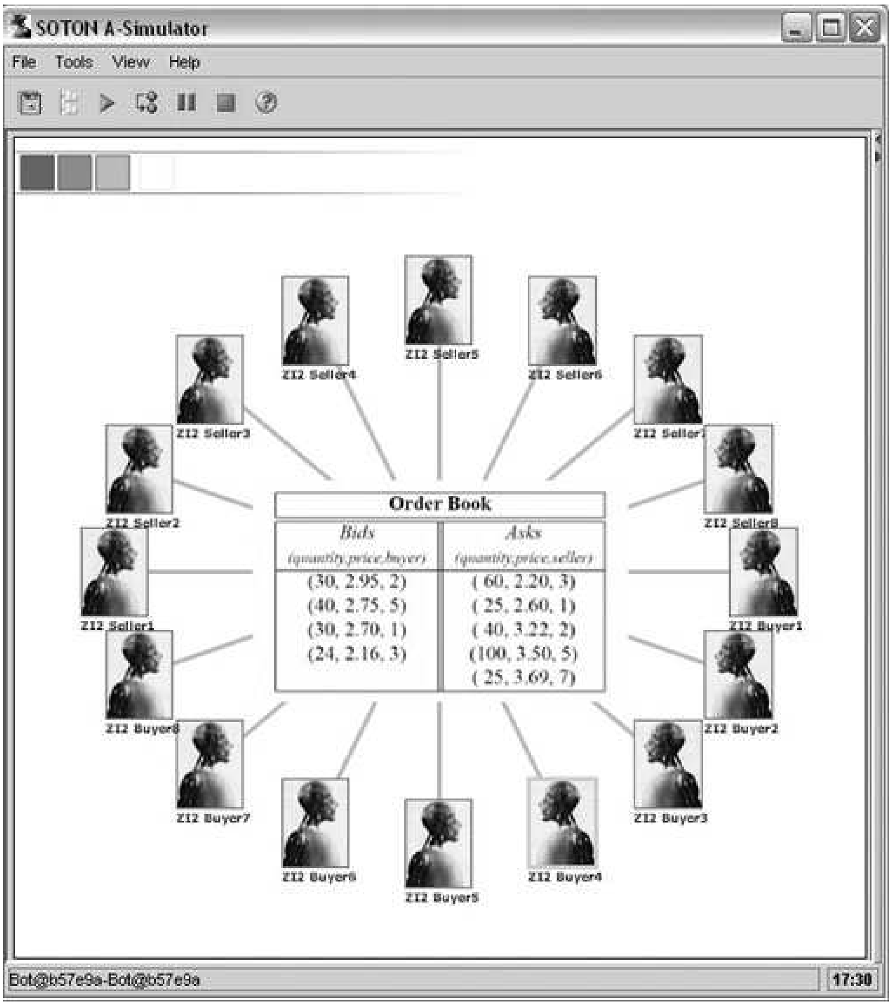


FIGURE 4.4: The multi-unit CDA simulator

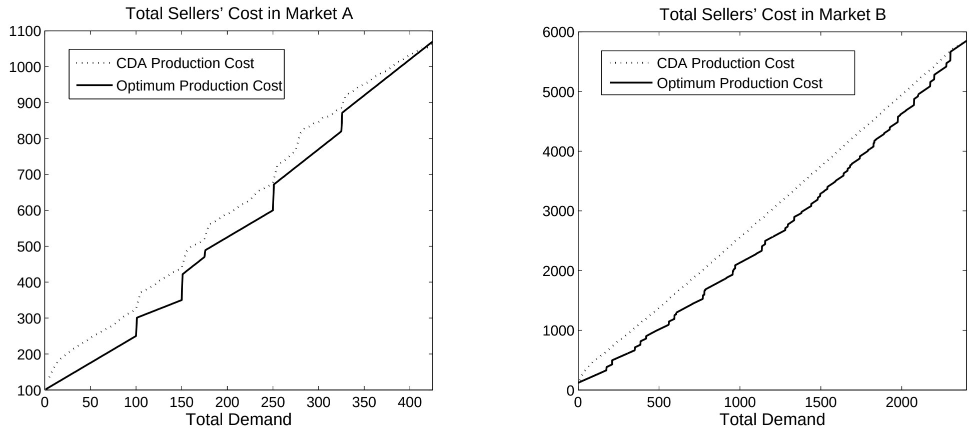
4.6.3 Empirical Evaluation

In order to perform empirical evaluations, we have developed an implementation of this distributed mechanism[12] (shown in figure 4.4) based on the protocol and strategies described here. As the experimental setup, we ran the simulations over 2000 rounds[13] for two different markets, more specifically a small market with 3 buyers and 3 sellers (*market A*) and a larger market with 15 buyers and 15 sellers (*market B*). We consider both the small and large markets so as to demonstrate the scaleability of our mechanism.

In each market, each seller was given a production function (supply for market A is given in table 4.1), while each buyer was required to procure an exact quantity of units with a relatively *high* limit price. We ran different simulations for each market, with different total demands ranging from 1 to the maximum production quantity. The total demand, D, was distributed

[12] Available at `http://www.ecs.soton.ac.uk/~rkd02r/simulator`

[13] The results were validated using a students t-test with two samples of 2000 runs, assuming equal variance with means $\mu_1 = 0.7198$ and $\mu_2 = 0.7218$ and p-value $p = 0.3660$. This means that the difference between the means is not significant and thus 2000 runs are sufficient for statistical significance at a confidence level of 95%.



(a) 3 buyers and 3 sellers (b) 15 buyers and 15 sellers

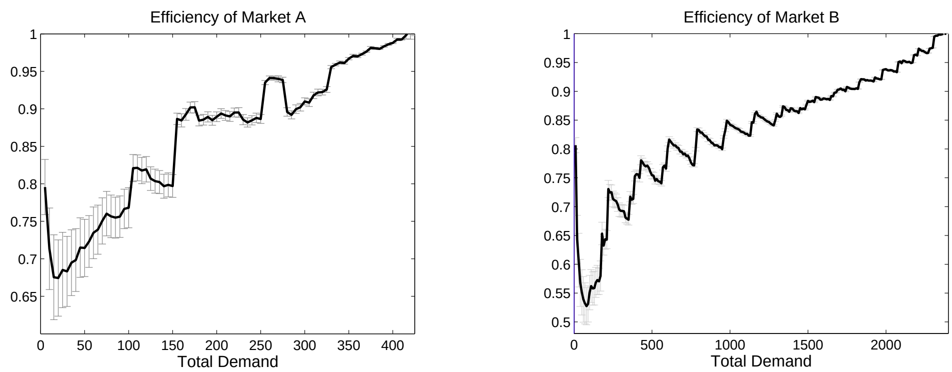
FIGURE 4.5: Optimal and CDA production cost

among the buyers (see table 4.4 for the demand in market A, where $D = \sum_i q_i, D \in [1, 425]$ given the sellers' production functions in table 4.1). Thus, the total demand in market A was varied from 1 to 425 (the maximum supply quantity of market A), while in market B the total demand ranged from 1 to 2400.

Buyers' Demand

B_i	B_1	B_2	B_3
allocation 1	100	150	50
...	...	...	...
allocation n	q_1	q_2	q_3

TABLE 4.4: A set of three buyers with different demands.



(a) 3 buyers and 3 sellers (b) 15 buyers and 15 sellers

FIGURE 4.6: Average market efficiency

In order to empirically evaluate the efficiency of the mechanism, in terms of minimising the total cost of production, we measure this property and compare it to the optimal solution found in the centralised mechanism. Given each total demand, the mean efficiency of the market (averaged over 2000 independent rounds) is shown in figure 4.6, where the optimal production cost is normalised to 1, while the total production cost of the centralised and the decentralised mechanisms are shown in figure 4.5. As can be seen, the mechanism is efficient with an average efficiency of 83% (and a minimum efficiency of 53% when demand is relatively low) for market B and an average efficiency of 86% (and a minimum efficiency of 67%) for market A. In both cases, the minimum efficiency case occurs when the demand is split amongst many more suppliers than are actually needed (with respect to the optimal allocation). This increases the overall cost of supply as a result of the fixed cost of the extraneous suppliers. However, in the typical CDA, the worst-case analysis considers the average efficiency of ZI agents [Gode and Sunder, 1993]. This is because although it is theoretically possible for an allocation of very low efficiency to occur, in almost every run (higher than 99% of the time), the CDA implemented with agents employing the ZI strategy has a high efficiency. Thus, it is the zero-intelligence nature of the strategy which provides a lower bound on measuring efficiency and, we expect the average efficiency with a more informed strategy to be better [Cliff and Bruten, 1997; He et al., 2003; Vytelingum et al., 2004]. We thererefore adopt this approach in discussing the inherent efficiency of our CDA mechanism.

In the experiments with each market, we observe an increasing trend whereby the market efficiency increases as total demand approaches the maximum capacity of the sellers. It can also be seen that there is a high variance when the total demand is relatively low. Considering specifically the set of experiments with market A, the intuitions behind these observations are as follows. The variance of the market efficiency is generally higher when the total demand is low. This is because the optimal allocation for a total demand of 100 is completely covered by seller 1 (with a marginal cost of 1.5 and a startup cost of 100). However, our market mechanism does not ensure that only seller 1 will trade and, thus, sellers 2 and 3 may also be part of this allocation for the total demand of 100. The high variance is principally an artifact of the additional startup costs if more than one seller were to trade. As the total demand increases past 175, the optimal allocation is covered by at least two sellers. Again, the variance past the demand of 175 is the result of sellers supplying different numbers of units at different marginal costs, with at most one additional startup cost. When the total demand is very high, close to the total capacity, all the sellers participate in the allocation, and the small variance is solely due to the sellers providing different numbers of units (a difference which is relatively low compared to the total startup cost). The observations in the set of experiments with market B can also be explained by the same reasoning, with the higher variance occurring when demand that can be covered by a single seller is distributed among multiple sellers.

Furthermore, we can explain the increasing trend of the market efficiency seen in figure 4.5. Considering market A, a demand of up to 175 can be provided by only 1 seller. The jumps in figure 4.5 correspond to the optimal allocation changing between a combination of one to three sellers. For example, jumps at 100 and 150 correspond to the optimal allocation starting with seller 1, changing to seller 2 and finally to seller 3. The increase in efficiency as total demand increases is the result of the number of sellers involved in the optimal allocation, changing from

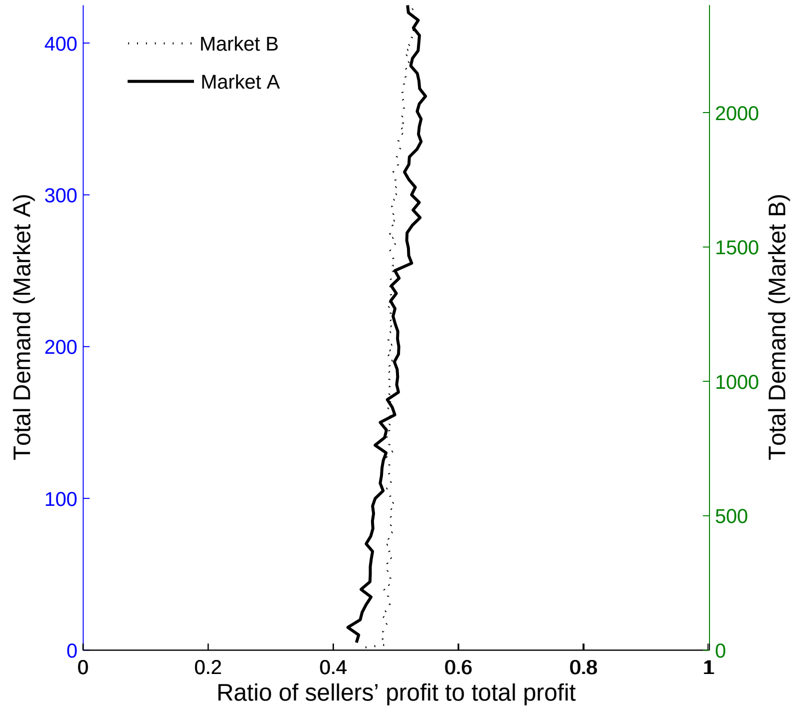


FIGURE 4.7: The sellers' total profit given different demands (for market A with 3 buyers and 3 sellers and market B with 15 buyers and 15 sellers.

a single seller (up to a total demand of 175) to three sellers (past a total demand of 325 which is the highest demand any two sellers can cover). However, in our market, any number of sellers can trade at any time. Thus, as total demand increases, the loss in efficiency that arises from the extra startup costs (compared to the optimal allocation) decreases which in turn explains the generally increasing trend. In the simulations with market B, a similar trend can be observed, with a lower efficiency when demand is lower than the minimum sellers' capacity (210). As in market A, there are more inefficient allocations that can arise when demand is low (and can be satisfied by a single seller), which would decrease the average efficiency much more than it would given a smaller number of inefficient allocations. Here, we use the same reasoning as in market A to explain the jumps, which are larger in number given the larger number of participants.

As well as being efficient, the simulation results in figure 4.7 show that, broadly, the sellers and buyers do indeed equally share the market profits (the ratio of sellers' profits to total market profit is approximately equal to 0.5 in both cases). This fair division of profits arises from the design of the clearing rule (see section 4.6.1). This is important because this profitability means that the agents are incentivised to enter the market which means our distributed mechanism can be viewed as being individually rational.

Having analysed two different markets (A and B) in detail, we now examine how the efficiency

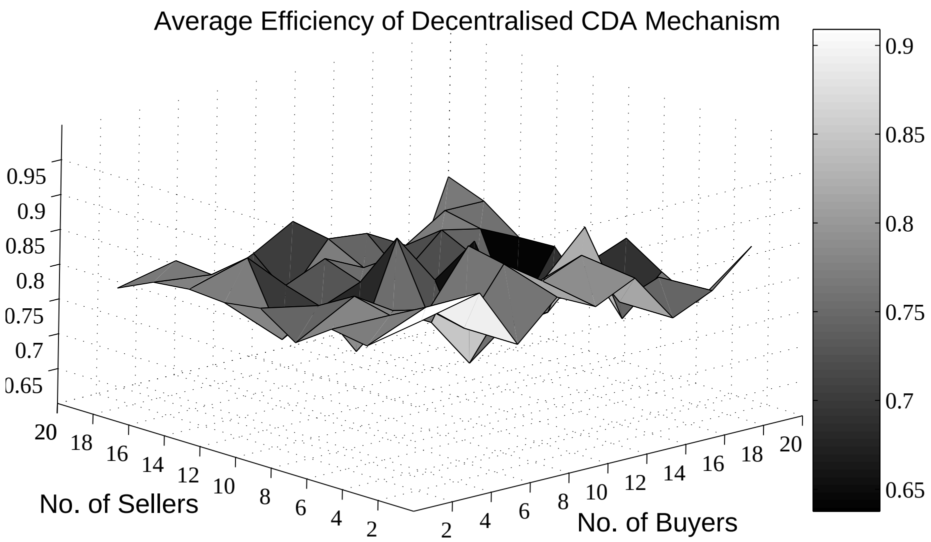


FIGURE 4.8: Performance of decentralised mechanism in different markets with different number of buyers and sellers.

of our mechanism scales up over different markets (see figure 4.8). In order to do so, we find the average efficiency of markets as the number of buyers and the number of sellers are respectively varied from 2[14] to 20. We run the auctions over 500 iterations with sellers randomly allocated their supply and buyers having a demand ranging from 1 upto the total supply divided by number of buyers. As can be seen, the average efficiency of the mechanism is maintained as the size of the market increases. The average efficiency ranges between 0.65 and 0.95 with no correlation to the market size, which implies that it is unaffected by the size of the market. In short, this means that the market scales.

4.7 Summary

In this chapter, we have presented our work on the development of two complementary mechanisms for task allocation. We considered a scenario where production costs are characterised by a cost function composed of a fixed cost, a constant marginal cost and a limited capacity and where we were seeking the minimal total production cost that satisfies demand.

Specifically, in the first mechanism we extend the standard VCG mechanism to our problem domain in order to incentivise selfish agents to report truthfully about their types thereby enabling the mechanism to find the efficient allocation. This required a novel penalty scheme to ensure that the mechanism is strategyproof for agents misreporting both their cost and their capacities. Individual rationality is conserved under this new mechanism and we show how this mechanism

[14] A minimum of two sellers and two buyers is required for a double auction.

is robust to uncertainties in the capacities of the agents. We then presented a dynamic programming algorithm, that solves the task allocation problem of the centre in pseudo-polynomial time.

In the second mechanism, we extend the standard format of a CDA so as to develop a decentralised mechanism for resource allocation in the same context. We find that this mechanism has a fairly high inherent average efficiency (over 65% in the examples we study) by testing it with a variant of the ZI strategy.

When taken together, we find that these mechanisms represent a trade-off in terms of efficiency and the decentralisation of a mechanism (in the examples we consider, the loss in efficiency can range from 0% to 50% depending on the demand and number of buyers and sellers in the market). However, both mechanisms still ensure that the participants derive a profit by joining the mechanism, thereby justifying their use with selfish agents.

In sum, in this chapter, we have designed two mechanisms for addressing requirement 2 in the list detailed in chapter 1(namely the requirement for mechanisms that deal with constrained capacity). Furthermore, the distributed CDA mechanism addresses requirement 1 in that it is a distributed allocation mechanism. This chapter concludes part I of this thesis. We have found that whilst there are numerous advantages to implementing distributed allocation mechanisms (see Chapter 1) there is usually an efficiency cost associated with distributed allocation mechanisms. Chapter 3 showed that in the case of rational agents, this efficiency is lower bounded at $1 - \frac{1}{e}$. In this chapter, we showed that the average efficiency of a mechanism based on the CDA can drop to around 65% when agents within the system are employing a zero-intelligence strategy. Thus, in the context of MSNs, it will be imperative to judge whether these distributed mechanisms justify their efficiency cost. The next part of this thesis will now consider distributed information mechanisms. In these mechanisms, the agents do not form their valuation or cost solely their privately observed type (as considered so far) but also on those of other agents within the system.

Part II

Distributed Information Mechanisms

The previous part of this thesis (Chapters 3 and 4) considered issues associated with mechanisms that enable distributed allocations. In this part of this thesis, we switch the focus to distributed information which is another core challenge within distributed mechanisms (as depicted in figure II.1). Specifically, this challenge considers how to design mechanisms when the agents determine their valuations of goods within a market from distributed pieces of information that are privately known by other agents within the system.

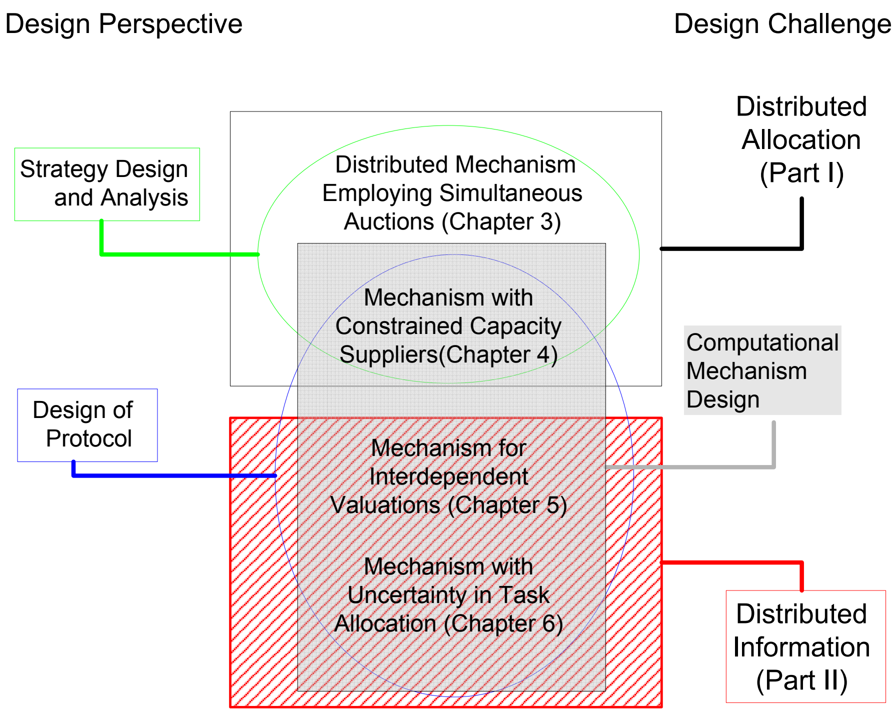


FIGURE II.1: The challenges addressed and the design perspective of part II of the thesis.

Now, it could be argued that traditional auction mechanisms already aggregate distributed pieces of information from different agents in order to determine the outcome of the mechanism. However, these mechanisms only deal with private signals, where an agent can formulate its valuation of a good or service once it is aware of its own signal. This is, therefore, only a very limited form of distributed information whereby only the centre requires these distributed pieces of information so as to determine the outcome of the mechanism (i.e. the allocation of resources and transfers of money). A more general form of distributed information occurs when every agent within the system is potentially reliant on the signals observed by other agents in order to formulate their valuations of goods or services. In this case, the agents cannot determine their valuations until they know the signals observed by the other agents. However, since the agents are selfish, they would not share this information unless they have an incentive to do so. Given this, this part of the thesis considers how to provide these incentives to the agents within the system, whilst still preserving certain desirable system properties such as efficiency and individual rationality.

In more detail, Chapter 5 reports on the design of an efficient and individually rational protocol for allocating multiple items to buyers who have interdependent valuations. Here the distributed information occurs since the agents require the private observations of other agents in order to formulate their interdependent valuation. We then consider in Chapter 6 a specific type of interdependent valuation which arises out of the uncertainty that agents have concerning the success rate of other agents within the system. In this case, the distributed information is of the form of the reports an agent gathers from other agents within the system in order to form perceptions about the success rates of agents within the system.

Chapter 5

Mechanisms for Interdependent Valuations

This chapter is the first of two that concentrates on designing a centralised protocol for scenarios where the agents form their valuations from distributed information. Specifically, we consider the case where agents form their valuations based on the private signals that they observe, as well as those observed by other agents within the system (which is the problem that requirement 3 seeks to tackle). This particular case is termed an interdependent valuations environment and arises in our MSN scenario when sensors fuse different observations about the same event. Specifically, we use an information-theoretic measure to derive this interdependent valuation function. Now, as we later show in this chapter, the VCG mechanism does not conserve its desirable economic properties of being incentive compatible, efficient and individually rational in this case. We therefore address requirement 3 by designing a new mechanism for interdependent valuations that does exhibit these properties. Furthermore, we show that the new mechanism does not add any additional computational burden over and above that of the VCG.

The remainder of the chapter is organized as follows: Section 5.1 introduces the research on efficient protocols for interdependent valuations. We then explain how interdependent valuations are relevant to the running MSN scenario in Section 5.2. This research is then put into the context of general MAS settings in Section 5.3 where related work is discussed. In Section 5.4, we demonstrate how values can be assigned to the measured data using information theoretic principles. We then discuss certain assumptions that are critical for any interdependent valuation mechanism to be efficient. Section 5.6 goes on to presents an efficient, incentive-compatible, individually-rational mechanism when buyers wish to have only a single good. This mechanism is then generalised for the case where buyers requires more than one item in Section 5.7. We then prove the economic properties of the mechanism and discuss its computational properties in Section 5.8. The main contributions of this Chapter are summarised in Section 5.9.

5.1 Introduction

We have demonstrated in Chapter 4 how auction mechanisms are valuable for task and resource allocation problems in MASs consisting of selfish agents. However, a key shortcoming of traditional mechanisms is that they are based on the assumption of private independent valuations to achieve these desirable properties. Such private valuations arise when an agent forms its valuation of the goods or services based *solely* on its own observation or signal (e.g. the value of a particular car to an agent depends solely on the agent's own perception of the car's use and is not dependent on the valuations of other bidders). Thus the private observations of agent i are usually encompassed by θ_i, the type of agent i (refer to chapter 2 for more details). However, the more general case is that valuations are actually *interdependent* (e.g. if the agents' valuations were to consider not only the car's use, but also its potential re-sale value, the valuation would clearly be dependent on the valuations of other bidders). In this case, the bidders form their perception of the value of an item based on the distributed information gathered by other agents within the system. Now, as we show in section 5.7, the desirable properties of the VCG mechanism no longer hold in this case and the auction is not guaranteed to be efficient when agents have interdependent valuations. To rectify this, we develop a new mechanism that is.

In more detail, interdependent valuations occur most commonly within MASs when agents have noisy or uncertain estimates of the true value of a good. For example, consider the case of agents bidding for a service in some form of computational economy (as is found, with web services or grid computing). In such cases, the value of a service to an agent is often dependent on the time of response between submitting a request and receiving the desired service. However, in many such cases, the dynamic and open nature of most of these systems means that each agent is only likely to have limited previous experience of a given service and thus it will only have an imprecise estimate of its expected response time. Now, if the agent knew the response time of other agents that have used this service (e.g. by asking them about their previous experience or by deducing it from their bidding behaviour), it would be able to form a more accurate estimate of the future response time (by cross-correlating from a broader set of experiences). Hence each agent's valuation is dependent on the signals (in this case, the response time) observed by the other agents bidding for the service and thus we again have interdependent valuations. Another instance where interdependent valuations have been documented is in the FCC spectrum auctions [Cramton, 1997] where it was found that bidders formed their valuations based around the beliefs and actions of other bidders. In these auctions, each bidder wanted to infer from the bidding actions of the other bidders how much they valued the spectrum licenses that were being offered. Thus, whilst each bidder had carried out independent research to gauge the market profitability of these spectrum licenses (i.e. how much money can an agent potentially make by using the license if it wins it), they wanted to use the information gained by the other bidders as well.

To overcome the independent valuation limitation, a number of researchers have developed efficient auctions for interdependent valuation scenarios where a single item is allocated (see Sections 5.3 and 5.6 for more details). However, in this work we are interested in the case of multiple items being allocated (i.e where agents may be interested in combinations of items such as a bundle of services). This extension also allows us to consider the important case of combinatorial allocations. These allocations deal with items exhibiting complementarities and substitutabilities and are known to be more efficient than multiple concurrent auctions of single goods (as shown in chapter 3). Such allocations occur in many real world scenarios such as the grid services and FCC spectrum auctions we mentioned earlier.

Now, as we discuss in Section 5.2, such distributed information also needs to be catered for within MSNs when data fusion needs to be carried out. In order to address this problem, we first formulate a function that characterises the value that an agent places on a particular piece of data originating from other agents. We then develop, for the first time, a direct mechanism that can allocate multiple items in an interdependent valuation scenario where each agent receives a single-dimensional signal (for example, a time of response in the computational economy or market profitability in the case of the FCC spectrum). We restrict our attention to single-dimensional signals because in an interdependent valuation scenario it is not possible to develop an efficient auction for multi-dimensional signals [Jehiel and Moldovanu, 2001][1]. Moreover, the single-dimensionality of the signal is not overly restrictive because in many cases the necessary information can be encompassed into a representative single-dimensional signal. In fact, we demonstrate in section 5.2, that this is indeed the case for the MSN scenario we study. In developing this mechanism and studying its application, we advance the state of the art in the following ways:

1. We formulate a novel valuation function based around the information form of the Kalman filter [Manyika and Durrant-Whyte, 1997] since this is the simplest and most elegant way of fusing different measurements of the same observation. This function equates the valuation to the expected gain in information when data from a number of sources is fused.

2. We extend the standard VCG mechanism to deal with interdependent valuations in the case of multiple goods in which agents receive a single-dimensional signal.

3. We prove the economic properties of our mechanism. In particular, we show that it is incentive-compatible, individually rational and efficient.

4. We analyse the computational properties of our mechanism and show that it does not impose any additional computational load on the agents compared to an independent valuation scenario. However, there is a corresponding increase in the centre's computational load.

[1] However Mezzetti [2003] shows that if we adopt a two-stage approach to the auction design, we can then achieve efficiency and incentive-compatibility in certain cases.

5.2 Interdependent Valuations within the MSN Scenario

We now discuss how sensors can have interdependent valuations within the running MSN scenario that we consider (as highlighted in the red rectangles in figure 5.1). To this end, consider multiple selfish sensors that are monitoring a particular area under the constraint of limited communication bandwidth between the sensors. These sensors are interested in obtaining the data gathered by other sensors and, as a result, are willing to pay for this data. Now, in contrast to chapters 3 and 4, the sensors in this case cannot place a value on the data before the sensors communicate their signals to the centre. Instead, they can only provide a function describing how the signals from other sensors would affect their valuation. Such a case arises in our MSN when sensors fuse uncertain information about target estimates in order to obtain a more precise measurement. Then, the knowledge about a particular measurement's precision affects how much value another agent places on it. However this knowledge is only known to the agent that has carried out the measurement. This results in an interdependent valuation scenario where the sensors can only provide a function stating how much value they will place on the data given the signals from the other sensors.

In section 5.4, we shall derive an information-theoretic valuation function which prescribes the value that sensors should place on a piece of data when the sensors wish to fuse target information. In so doing, we also generalise the target-detection scenario to the case where the different sensors have different regions of interest (as opposed to the single region of interest considered so far).

5.3 Related Work

The derivation of valuation functions for MSNs has recently become an active area of interest since it enables a host of cooperative and competitive task allocation mechanism to be employed [Chu et al., 2002; Zhao et al., 2002; Lesser et al., 2003; Iyengar and Brooks, 2005]. On one hand, in [Lesser et al., 2003], the valuation functions do not have an information-theoretic basis. Rather they are specified by the system designer and take into account subjective measures such as how many readings of a target is enough and what is the most important target. On the other hand, [Chu et al., 2002; Zhao et al., 2002] provide guidelines for adopting information-theoretic valuation functions for collaborative target-detection and tracking. Here we adopt this latter approach and use one of these measures, namely the entropy measure, and develop it for the target-detection scenario we consider. Finally, in [Iyengar and Brooks, 2005], it is assumed that there is some information measure which guides the decision-making in the sensors. However, in contrast to our work there are no detail about which information measure should actually be used.

Auctions for interdependent valuations have also been considered by a number of researchers [Krishna, 2002; Dasgupta and Maskin, 2000; Jehiel and Moldovanu, 2001]. In particular, there

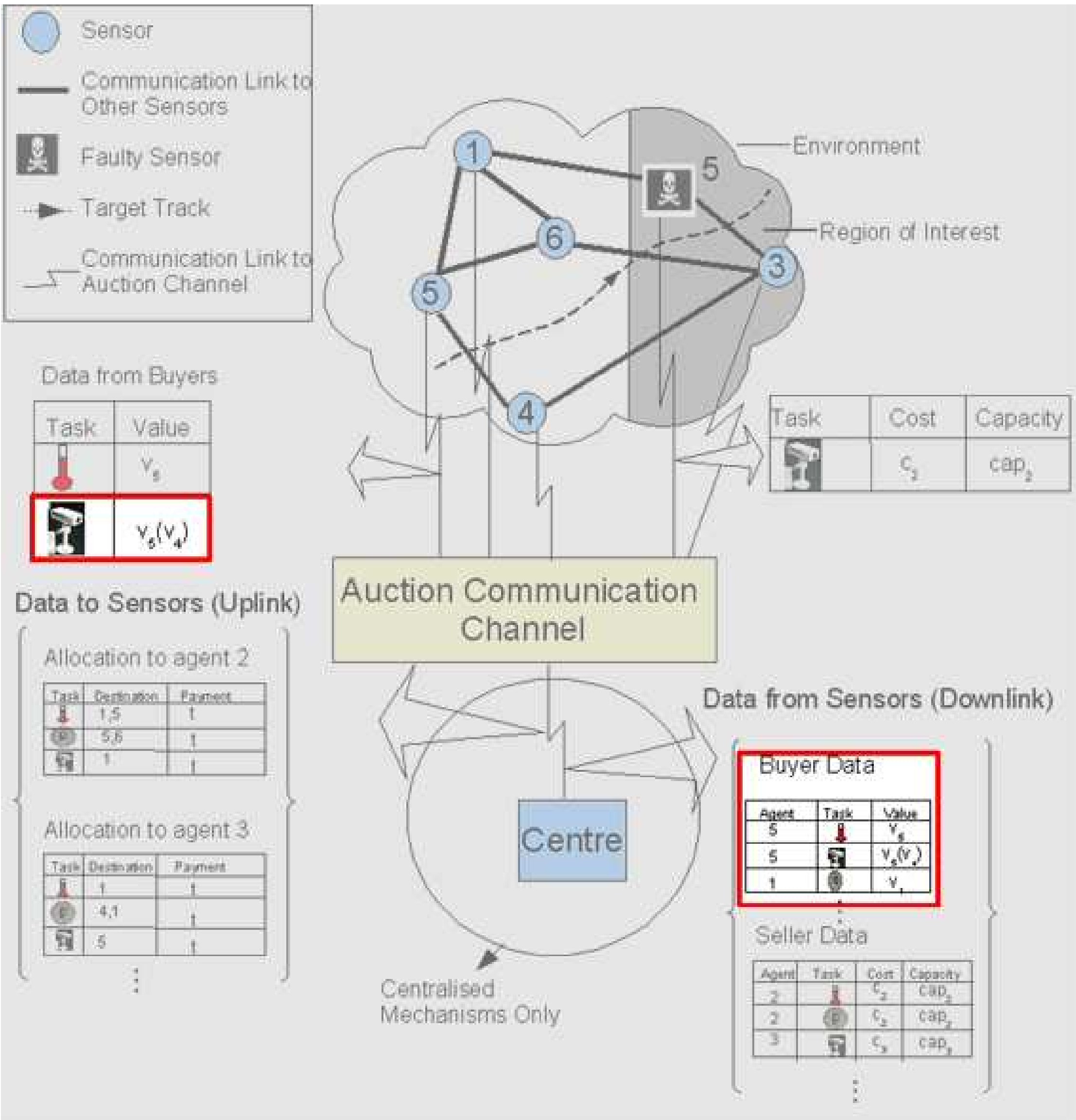


FIGURE 5.1: Figure of the MSN scenario highlighting the interdependent valuation of buyers considered within this chapter

are currently two main approaches to finding an efficient mechanism for the allocation of items with interdependent valuations. Krishna considers a direct mechanism for efficient allocations for multi-unit single items with single-dimensional signals. In this case, agents submit their interdependent valuation functions, as well as their signals, to a central auctioneer who then decides on the efficient allocation. The payment scheme was then devised so that the agents are incentivised to reveal their signals truthfully. Dasgupta and Maskin have also developed an efficient mechanism for the case of two non-identical items, again with single-dimensional signals. In their case, agents make contingent bids rather than submitting their valuation functions and observed signals (i.e. agent 1 submits a range of bids that describes its bid when agent 2 bids a particular value and vice versa). Thus the bidding is more complex than in Krishna's mechanism because the agents have to submit bids based on what other agents might bid, rather than just revealing their valuation function and signals. This bidding becomes even more complex in the

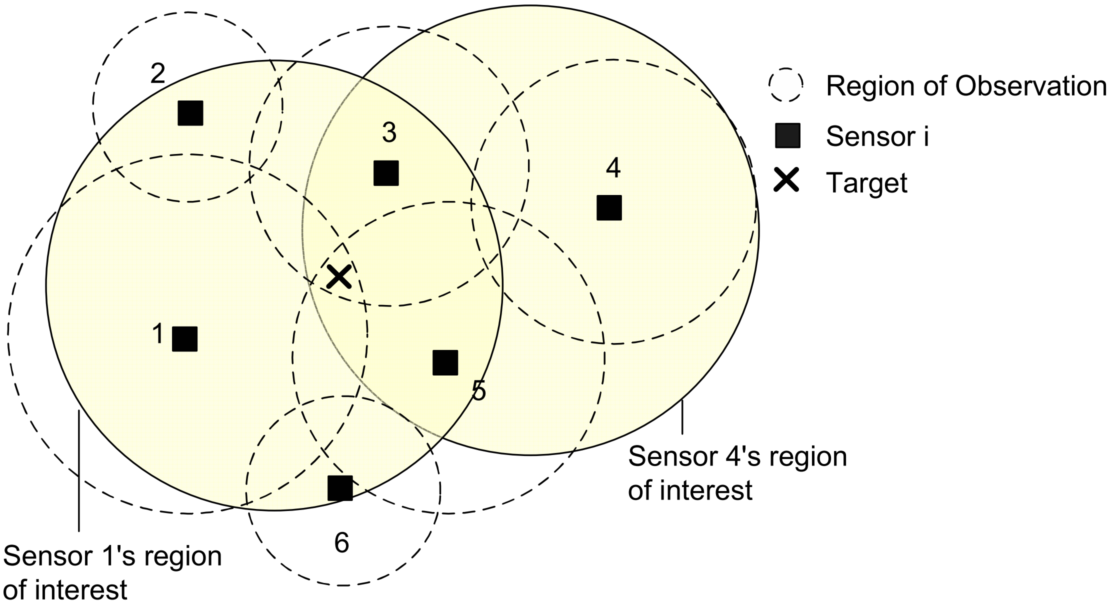


FIGURE 5.2: Figure showing a MSN detecting a target which falls in the region of interest of two sensors and region of observation of three sensors.

indirect mechanism they have developed for the case where multiple items need to be allocated.

Given this, in this chapter, we adopt the approach by Krishna, since the bidding is more straightforward for the agents. Specifically, we develop a direct mechanism in order to deal with the allocation of multiple items where each agent receives a single-dimensional signal. A naïve extension of the VCG mechanism is known not to work in this case [Krishna, 2002] and given this we show how to change the payment scheme in order to achieve the desirable economic properties of the VCG. We should note here that we do not concern ourselves with the problem of multi-dimensionality of these signals since it is known that allowing for multi-dimensionality of signals leads to inefficient allocations in direct mechanisms [Jehiel and Moldovanu, 2001]. If the agents can observe the outcome of their reports, then an efficient allocation with multi-dimensional types is possible [Mezzetti, 2003]. However, we believe that this is impractical in many cases because an agent might not be able to observe the outcome from a report (see [Mezzetti, 2003] for an example). Thus, in this chapter we consider direct mechanism where the agents can report on their types only once.

5.4 An Information-Theoretic Valuation Function

We now develop our valuation function based on the information form of the Kalman filter [Manyika and Durrant-Whyte, 1997]. To this end, we demonstrate how the distributed information filter can be used to fuse different pieces of information together. We then show how sensors in our scenario (depicted in greater detail in figure 5.2) can value the data held by other sensors according to the information gain they receive when obtaining the data.

To recap, the Kalman filter is an efficient recursive filter that estimates the state of a dynamic system from a series of incomplete and noisy measurements [Kalman, 1960]. The observations within a Kalman filter are of the form:

$$z(t) = H(t)y(t) + n(t)$$

where $y(t)$ is the state of the system at time t, $H(t)$ is the linear observation model and $n(t)$ is a zero mean random variable drawn from a normal distribution with variance R. The covariance update component (which measures how the uncertainty in the measurement varies as more data is collected), $P^{-1}(t \mid t)$, of the information form of the Kalman filter for N observations is:

$$P^{-1}(t \mid t) = P^{-1}(t \mid t-1) + \sum_{j=1}^{N} H^T(j)R^{-1}(j)H(j)$$

The summation in the above expression represents the decrease in covariance and thus the gain in information at time t when all the N observations are fused. In the case of our problem the value of receiving data from another agent can thus be represented by the gain in information this observation engenders.

In order to achieve an efficient allocation, this gain in information must be calculated from the measure of the data accuracy prior to actually fusing the data. Thus, we can represent the measure of accuracy of a data point, θ_j (which becomes an agent's type), as its covariance which is calculated from the covariance of its observation, $R(j)$:

$$\theta_j = H^T(j)R^{-1}(j)H(j) \tag{5.1}$$

Thus the gain in information of agent i when all relevant data is transmitted to it and fused, can be expressed as a sum of this measure of accuracy provided by each of the other agents:

$$v_i(\boldsymbol{\theta}) = \theta_i + \sum_{j \in -i} \theta_j \tag{5.2}$$

where $-i = \mathcal{I} \setminus i$.

Equations 5.1 and 5.2 thus cast our valuation function in the Kalman filter form. However, we need to modify this so as to incorporate the characteristics of our scenario. In particular, in our scenario each of the sensors has a region of observation which is the area it can sense and a region of interest which is the area it wishes to gather information about (as shown in figure 5.2). As a result, all observations may not fall in an agent's region of observation (as depicted in figure 5.2). Furthermore, an agent may not be able to receive all the data due to the bandwidth constraints of the communication network. Defining α_{ij} as the probability that the data observed by agent j is relevant to agent i and a vector K describing the allocation of the flow of data in the network, then the expected valuation is:

$$\overline{v_i}(\boldsymbol{\theta}, K) = \theta_i + \sum_{j \in -i} f_{ij}\alpha_{ij}\theta_j \tag{5.3}$$

By slight abuse of notation, we shall hereafter refer to the expected valuation $\overline{v_i}(.)$ as $v_i(.)$. From the valuation function, we can observe that the valuation of an agent i depends on θ_j, which are signals measured by other agents. This firmly puts us in the realm of interdependent valuations. We next describe the mechanism developed by Krishna for such valuations and single items, before detailing our mechanism for the multiple good scenario. However, before presenting the interdependent mechanisms, we shall discuss the assumptions that are critical for the auctions to be efficient.

5.5 Assumptions in Mechanisms with Interdependent Valuations

In this section, we discuss the assumptions that are required so that the mechanisms developed for interdependent valuations are efficient. In fact, it has been shown that when these assumptions are violated, then no efficient direct mechanism can be developed [Krishna, 2002; Jehiel and Moldovanu, 2001]. We shall also demonstrate how these assumptions are not overly restrictive for our sensor network scenario and are naturally satisfied by the MSN scenario we discuss.

Recapitulating, in this scenario, each agent i, $i \in \mathcal{I}$, observes a signal $\theta_i \in \Theta_i$ and forms its valuation $v_i(.)$ based on the vector of signals observed by all agents (i.e. $\boldsymbol{\theta}$) and the particular allocation $K \in \mathcal{K}$ being implemented. Thus, $v_i : \boldsymbol{\Theta} \times \mathcal{K} \rightarrow \Re_+$. The mechanism, $(\mathcal{M}, \boldsymbol{r})$, then consists of an allocation rule $\mathcal{M} : \boldsymbol{\Theta} \rightarrow \mathcal{K}$ that chooses the allocations and a payment rule $\boldsymbol{r} : \boldsymbol{\Theta} \rightarrow \Re_+^{|\mathcal{I}|}$ that determines the payments r_i from each agent, both being based on the reports of the signal values $\boldsymbol{\theta}$. Finally, we shall denote allocations induced by the true report of θ_i (all other agents $-i$ being truthful) as K_0^*. As θ_i is decreased, it is quite natural to expect that the allocation which is deemed efficient will change because the valuations of each allocation by the agents would also change. These allocations will be denoted by K_l^i with l being the index of each successive induced allocation as θ_i is decreased. Mirroring this, as θ_i is increased, the successive efficient allocations are denoted by K_{-l}^i.

Assumption 5.1. $\frac{\partial v_i}{\partial \theta_i} > 0$.

> This implies that higher values of the signal lead to higher valuations for the agent. This restricts the signal of the agent to vary in one direction only, thereby making it impossible for an agent to have the same valuation of a particular allocation for two different signal values. For example, in the case of a computational economy, this would imply that the valuation always increases with rapidity of service (which is the measured signal θ_i). In the case of the multi-sensor network scenario, this condition is automatically satisfied since new data cannot decrease information.

Assumption 5.2. $\frac{\partial v_i}{\partial \theta_i} > \frac{\partial v_j}{\partial \theta_i} \;\; \forall i, j \in \mathcal{I}, i \neq j.$

This implies that an agent's signal affects its own valuation more than it affects the valuation of any other agent. This assumption is the single-crossing condition analogue in the interdependent scenario [Krishna, 2002; Mirrlees, 1971]. Without this condition, no efficient mechanism can exist. In the case of a computational economy, this implies that the agent puts more credence on the rapidity of service it measured, as opposed to the one observed by other agents. In our scenario, this assumption implies that the region of observation of any sensor is not a subset of the region of observation of any other sensor (i.e. no agent is redundant in this system).

Assumption 5.3. $\frac{\partial v_i}{\partial \theta_i}(., K^i_p) \geq \frac{\partial v_i}{\partial \theta_i}(., K^i_q)$ if $p < q$

This implies that if a higher value of θ_i induces an allocation K^i_p, then agent i's value changes more rapidly in this new allocation than in the previous one K^i_q. This implies that on receiving a higher θ_i, the centre allocates a set of goods to i in the new allocation K^i_p where i's valuation changes more rapidly, than in the previous set K^i_{p+1}. To better explain this assumption, consider a situation where there are two services to be allocated and an agent has a complementary valuation of them. Suppose that the agent is allocated a particular service when $\theta_i = \alpha$. Now, if θ_i is increased, there will come a point $\theta_i = \beta > \alpha$ when it will be efficient to allocate both services to the agent (since from assumption 5.2, its valuations will increase more rapidly than that of other agents). This assumption then implies that the rate of change of the valuation with respect to θ_i is greater in this new allocation than in the previous one. Consider, for example, two agents bidding for two pieces of data in our MSN scenario. Then suppose that as θ_i is increased, it first becomes more efficient to allocate one piece of data (denote this allocation as K^i_{-1}) and then both pieces of data to agent i (denote this allocation as K^i_{-2}). Then this assumption implies that $\frac{\partial v_i}{\partial \theta_i}(\theta_i, \theta_{-i}, K^i_{-2}) \geq \frac{\partial v_i}{\partial \theta_i}(\theta_i, \theta_{-i}, K^i_{-1})$ (i.e. agent i's valuation increases more rapidly with θ_i when it is allocated both pieces of data rather than only one).

In the next section, we present the mechanism developed by Krishna for efficient allocation in single good scenarios.

5.6 A Mechanism for Single Goods

Having discussed the assumptions that are required for an efficient mechanism, we now provide an exposition of an efficient, individually rational and incentive-compatible mechanism developed in [Krishna, 2002]. Though this mechanism is limited to single good allocations, it provides an introduction to the design of efficient interdependent mechanisms.

In more detail the mechanism proceeds as follows:

1. Each agent i transmits to the centre its valuation function $v_i(\boldsymbol{\theta})$[2].

2. Each agent i also transmits its observed signal $\widehat{\theta}_i$.

3. The centre then allocates the item to the buyer that has the highest value it[3]:

$$K_i^* = \begin{cases} 1 & if v_i(\widehat{\boldsymbol{\theta}}) > max_{j \neq i} v_j(\widehat{\boldsymbol{\theta}}) \\ 0 & if v_i(\widehat{\boldsymbol{\theta}}) < max_{j \neq i} v_j(\widehat{\boldsymbol{\theta}}) \end{cases}$$

4. The centre also calculates the payment r_i made by agent i if it wins the allocation (i.e. $K_i^* = 1$) as:

$$r_i = v_i(z_i(\widehat{\boldsymbol{\theta}_{-i}}), x_{-i})$$

where

$$z_i(\boldsymbol{\theta_{-i}}) = \inf \{y_i : v_i(y_i, \boldsymbol{\theta_{-i}}) \geq max_{j \neq i} v_j(y_i, \boldsymbol{\theta_{-i}})\}$$

The buyers, j ($j \in \mathcal{I} \setminus i$), for whom $K_j = 0$, pay nothing (i.e. $r_j = 0$)

Thus, the signal $z_i(\boldsymbol{\theta_{-i}})$ is the smallest value of θ_i that i could report and still receive the item, given the reports of the other agents $\widehat{\boldsymbol{\theta}_{-i}}$. In more detail, figure 5.3 demonstrates how the payment is calculated when allocating a good when there are two agents i and j wishing to have that good. The value of θ_i which has been observed by i implies that it should be awarded the object. The payment is calculated at the point in Θ_i when $v_i > v_j$, keeping θ_{-i} constant.

Krishna proves this mechanism to be incentive-compatible, individually-rational and efficient in an ex-post Nash equilibrium. Having thus detailed the mechanism for the single-good scenario, we now develop our mechanism for multiple goods that builds upon it.

5.7 A Mechanism for Multiple Goods

In this section, we extend Krishna's approach in order to develop a direct mechanism that is incentive-compatible, efficient and individually-rational for the case of multiple goods with single-dimensional signals.

Specifically, the mechanism we have developed proceeds as follows:

1. The centre announces the set of items M that are to be auctioned off.

2. Each agent i transmits to the centre its valuation function $v_i(K, \boldsymbol{\theta})$ for all the possible allocations $K \in \mathcal{K}$.

[2]We refer to the valuation as $v_i(\boldsymbol{\theta})$ meaning $v_i(K, \boldsymbol{\theta}) = v_i(\boldsymbol{\theta})$ if $K_i = 1$ and is 0 otherwise.

[3]Ties are decided by a random function assigning equal probability of winning to each of the agents in the tie.

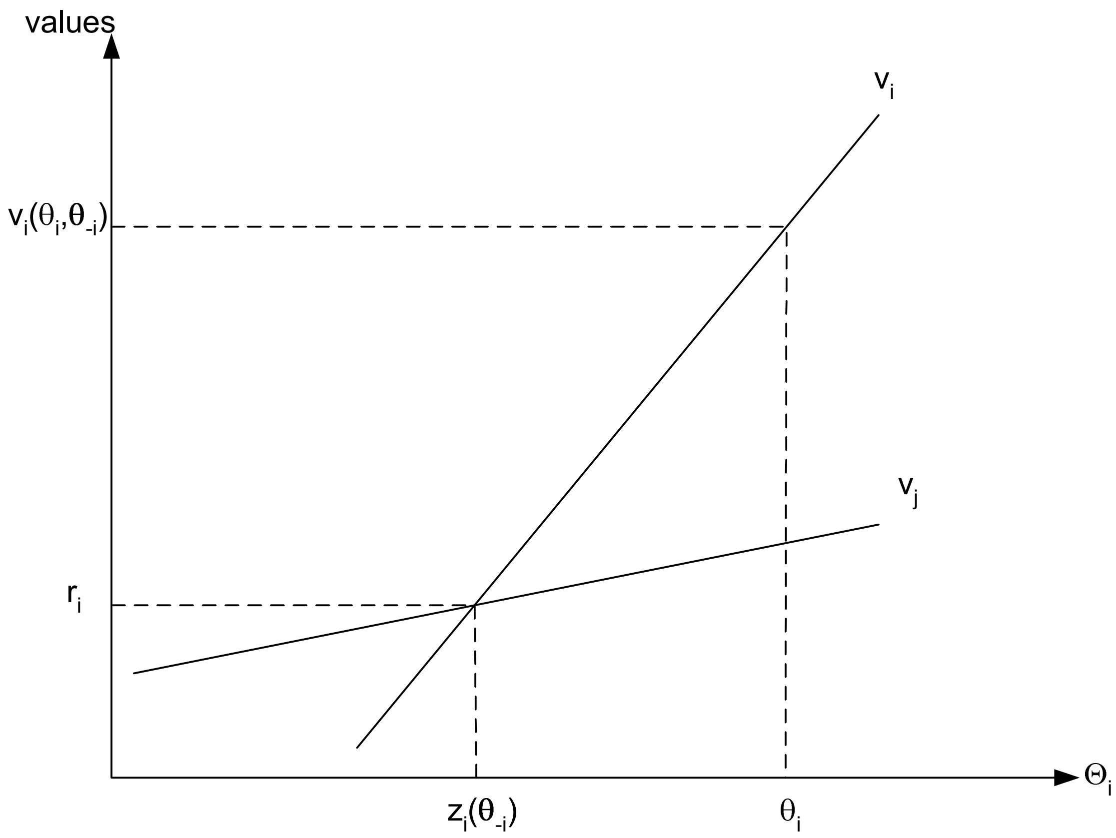


FIGURE 5.3: Figure demonstrating how payments are calculated in Krishna's mechanism for single good and interdependent valuations.

3. Each agent i also transmits its observed signal $\widehat{\theta}_i$. [4]

4. The centre then computes the optimal allocation K_0^* which is calculated as:

$$K_0^* = \arg\max_{K \in \mathcal{K}} \left(\sum_{i \in \mathcal{I}} v_i(K, \widehat{\boldsymbol{\theta}}) \right) \tag{5.4}$$

5. The centre also calculates the payment r_i made by each agent i. To do this, the centre first finds the m next best allocations as the reported signal $\widehat{\theta}_i$ is decreased, until the presence of i makes no difference to the allocations. That is, find allocations $K_1^i \ldots K_m^i$ and the signal values z_i^l such that:

$$z_i^l = \inf\left\{ y_i : \sum_{i \in \mathcal{I}} v_i(K_l^i, y_i, \boldsymbol{\theta_{-i}}) = \sum_{i \in \mathcal{I}} v_i(K_{l+1}^i, y_i, \boldsymbol{\theta_{-i}}) \right\} \tag{5.5}$$

(where each allocation K_l^i is different) until:

$$z_i^m = \inf\left\{ y_i : \sum_{i \in \mathcal{I}} v_i(K_{m-1}^i, y_i, \boldsymbol{\theta_{-i}}) = \sum_{i \in \mathcal{I}} v_i(K_m^i, y_i, \boldsymbol{\theta_{-i}}) \right\} \tag{5.6}$$

[4] Of course, $\widehat{\theta}_i$ may not be equal to θ_i. However, we prove in section 5.8 that it is a best strategy for the agent to set $\widehat{\theta}_i = \theta_i$.

where the allocation K_m^i is the optimal allocation when i does not exist:

$$K_m^i = \arg\max_{K \in \mathcal{K}} \sum_{j \in -i} v_j(K, \boldsymbol{\theta})$$

Then the transfer [5] to buyer i is:

$$r_i = \sum_{l=0}^{m-1} \Big[\sum_{j \in -i} v_j(K_l^i, z_i^l, \boldsymbol{\theta_{-i}}) - \sum_{j \in -i} v_j(K_{l+1}^i, z_i^l, \boldsymbol{\theta_{-i}}) \Big] \tag{5.7}$$

The above scheme rests upon making an agent derive a utility equal to the marginal contribution that its presence makes to the whole system of agents (which is the same intuition as used in the VCG). Thus the additional part of this mechanism is to take into account the effect that an agent's signal θ_i has on the overall utility of the system.

This mechanism is general and is shown (below) to reduce to the well-known multiple-good private value model if we take the case of independent valuations (i.e when $(v_i(\boldsymbol{\theta}, .) = v_i(\theta_i, .))$. Then the optimal allocation (from equation 5.4) is:

$$K_0^* = \arg\max_{K \in \mathcal{K}} \left(\sum_{i \in \mathcal{I}} v_i(K, \widehat{\theta}_i) \right)$$

To calculate the payment scheme, we first note that with independent valuations θ_i only affects $v_i(.)$. Thus repeatedly decreasing θ_i, until the stopping condition on equation 5.6, does not change the valuation of the other agents $-i$ on the different allocations. This then implies that in the payment (as computed by equation 5.7) all the terms cancel each other, except for the first and last, leading to a payment of:

$$r_i = \sum_{j \in \mathcal{I} \backslash i} v_j(K_0^*, \widehat{\theta}_j) - \sum_{j \in \mathcal{I} \backslash i} v_j(K_m^i, \widehat{\theta}_j) \tag{5.8}$$

This is exactly the payment scheme for the multiple-good private values model which we discussed in section 2.4.1. Thus, this shows that the classical VCG mechanism is an instance of the generalised mechanism developed here. Furthermore, notice that assumption 5.2 is automatically satisfied in this independent valuation scenario, since $\frac{\partial v_j}{\partial \theta_i} = 0$ in such a scenario. Also, since an increase in θ_i would only increase $v_i(., \theta_i)$, any increase in θ_i that induces a new allocation would imply that the rate of change of $v_i(., \theta_i)$ with respect to θ_i is higher in the new allocation than in the previous one. Thus, assumption 5.3 is also automatically satisfied in the independent valuation scenario.

[5] If the transfer is negative, it implies that buyer i pays to the centre.

TABLE 5.1: Valuations of the players with each allocation

Allocation	$v_1(K, \boldsymbol{\theta})$	$v_2(K, \boldsymbol{\theta})$	$v_{\mathcal{I}}(K, \boldsymbol{\theta})$
$(AB, \varnothing)$	$4\theta_1 + 2x_2$	0	$4\theta_1 + 2\theta_2$
(A, B)	$2\theta_1 + \theta_2$	$\theta_1 + 2\theta_2$	$3\theta_1 + 3\theta_2$
(B, A)	$\theta_1 + \theta_2$	$0.5\theta_1 + 2\theta_2$	$1.5\theta_1 + 3\theta_2$
$(\varnothing, AB)$	0	$\theta_1 + 4\theta_2$	$\theta_1 + 4\theta_2$

5.7.1 Example of Interdependent Valuations

In order to further explain how the mechanism operates to achieve efficiency and incentive-compatibility we present an example that demonstrates how it computes the efficient allocation and the payments. We show why a straightforward implementation of the VCG mechanism would fail in this case. We will also consider the assumptions that we made in section 5.7 and show how the mechanism fails when these do not hold.

We consider a very simple case, namely that with two agents, 1 and 2, bidding for two different spectrum licenses A and B. The set of possible allocations consists of four members, which are $\mathcal{K} = \{(AB, \varnothing), (A, B), (B, A), (\varnothing, AB)\}$. In this case, each agent perceives a particular signal θ_i that determines the market profitability of the spectrum licenses. Table 5.1 shows the valuations of players 1 and 2 for each allocation, as well as the sum of their valuations.

We shall now consider how agent 1 views the mechanism as it reports its signal θ_1. The explanation for agent 2 is the same and is therefore omitted. Figure 5.4 shows how the value of each allocation varies for agents 1, 2 and the set of agents $\mathcal{I}$, as agent 1's reported signal θ_1 is increased. We denote agent 1 by i and agent 2 by $-i$ to demonstrate how this works in cases of more than two agents. Suppose that agent 1 has observed $\theta_1 = 1.5$ and agent 2 has observed a value of $\theta_2 = 2$. Then from the figure, we see that the efficient allocation in this case is $K_0^* = (A, B)$ (the efficient allocation is the one that maximises the value of $\mathcal{I}$). Furthermore, the values of θ_i at which it becomes more efficient to implement allocations $K_1^i = (\varnothing, AB)$ and $K_{-1}^i = (AB, \varnothing)$ are $z_i^0 = 1$ and $z_i^{-1} = 2$ respectively (shown in figure 5.4).

Hence we can calculate the overall utility that agent 1 derives from reporting truthfully, which from equation 5.7, is $v_i(K_0^*, \boldsymbol{\theta}) + v_{-i}(K_0^*, z_i^0\boldsymbol{\theta_{-i}}) - v_{-i}(K_1^i, z_i^0\boldsymbol{\theta_{-i}}) = 5 + 5 - 9 = 1$. Now, any report in the range $1 \leq \theta_i \leq 2$ will induce the same allocation and transfer and thus agent 1 has no incentive to report θ_i in this range different from the truthful value. If agent 1 reports $\theta_i > 2$, it will then derive a utility of $v_i(K_{-1}^i, \boldsymbol{\theta}) + v_{-i}(K_{-1}^i, z_i^{-1}\boldsymbol{\theta_{-i}}) - v_{-i}(K_0^*, z_i^{-1}\boldsymbol{\theta_{-i}}) + v_{-i}(K_0^*, z_i^0\boldsymbol{\theta_{-i}}) - v_{-i}(K_1^i, z_i^0\boldsymbol{\theta_{-i}}) = 10 + 0 - 6 + 5 - 9 = 0$, which is less than what it would derive from truthful reporting. Thus agent 1 would not over-report its observed value. The reason why this occurs is because, as shown in figure 5.4, $v_i(K_{-1}^i, \boldsymbol{\theta}) - v_i(K_0^*, \boldsymbol{\theta})$ is always less than $v_{-i}(K_0^*, z_i^{-1}\boldsymbol{\theta_{-i}}) - v_{-i}(K_1^i, z_i^{-1}\boldsymbol{\theta_{-i}})$ when the true value of θ_i is in the range $1 \leq \theta_i \leq 2$. If, on the other hand, the agent reports $\theta_i < 1$, it would then derive a utility of $v_i(K_1^i, \boldsymbol{\theta}) = 0$ which is again less than what it would derive from truthful reporting. We have thus demonstrated how an agent finds it in its best interest to report truthfully (see section 5.8 for a more general proof).

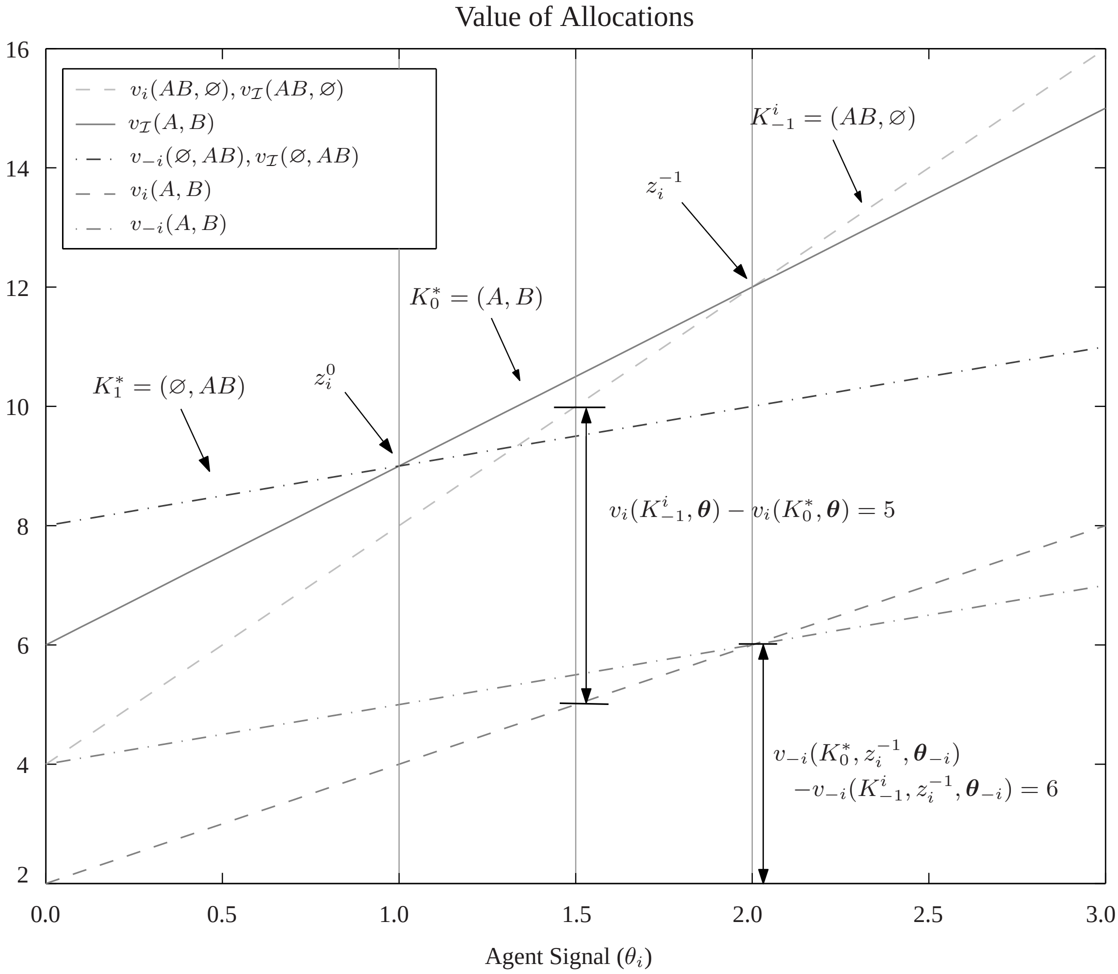


FIGURE 5.4: Valuations of $1, 2$ and $\mathcal{I}$ for each bundle as θ_1 is increased

Now consider applying a traditional VCG mechanism (as presented in section 2.4.1) to the above example. We shall assume that if ever an agent is not present in the system, then its related observation is zero. Then, the efficient allocation when $\boldsymbol{\theta} = (1.5, 2)$ is again $K_0^* = (A, B)$. However the payments from each agent in this case will differ. In the case of truthful reporting, agent 1 will pay (from equation 2.8) $\max_{K\in\mathcal{K}} v_{-i}(K, (\theta_1 = 0, \theta_2)) - v_{-i}(K_0^*, \boldsymbol{\theta})$ which is $8 - 5.5 = 2.5$, thereby deriving a utility of $5 - 2.5 = 2.5$. However, the agent can lie and report, for instance $\theta_1 = 1.8$ and obtain a utility of $5 - (8 - 5.8) = 2.8$. The incentive to lie is present because the traditional VCG does not take into consideration the effect that agent i's *signal* has on the valuation of other agents. Of course, in a private valuation scenario, this effect is by definition non-existent and thus the VCG exhibits its desirable properties in such scenarios. However, these properties are no longer conserved in an interdependent valuation scenario, as presented here.

The mechanism is guaranteed to work in the above example because the valuations satisfy the assumptions presented in section 5.7. We will now show how this mechanism would fail if ever any one of these assumptions does not hold.

In order to show what happens when assumption 5.1 fails, consider only the single good A. Suppose that agent 1 has a valuation of $(\theta_1 - 2)^2 + \theta_2$ for A and agent 2 still has the same valuation of $0.5\theta_1 + 2\theta_2$. Then the auctioneer in this case has to decide only between two allocations, namely $\mathcal{K} = \{(A, \varnothing), (\varnothing, A)\}$. With these valuations, it is efficient to allocate A to agent 2 when $2.25 - \sqrt{\left[(2.25)^2 - (4 - \theta_2)\right]} \leq \theta_1 \leq 2.25 + \sqrt{\left[(2.25)^2 - (4 - \theta_2)\right]}$. If $\theta_1 \leq 2.25 - \sqrt{\left[(2.25)^2 - (4 - \theta_2)\right]}$ agent 1 obtains the good and pays $2\theta_2$ according to equation 5.7. If $\theta_1 \geq 2.25 - \sqrt{\left[(2.25)^2 - (4 - \theta_2)\right]}$, then agent 1 again obtains A, but this time, it pays 6 (again using equation 5.7). Thus, it is always in the interest of agent 1 to state that its signal is in the lower range if its signal happens to occur in either of these ranges. Although assumption 5.1 may seem to be required only for our mechanism to work, this is not so, as it is required for *any* efficient, incentive-compatible mechanism [Mirrlees, 1971].

Now consider that the valuations of the good A are such that $v_1((A, \varnothing), \boldsymbol{\theta}) = 2\theta_1 + \theta_2$ and $v_2((\varnothing, A), \boldsymbol{\theta}) = 3\theta_1 + \theta_2 - 6$ (thus assumption 5.2 is not satisfied). In this case, it is efficient to allocate A to agent 1 when $\theta_1 < 6$ and to agent 2 otherwise. However, it is not possible to achieve an efficient mechanism in this case, since agent 1 will always state $\theta_1 < 6$ no matter what the real value of θ_1 is. In the case of our mechanism, agent 1 pays $\theta_2 - 6$ if it allocated the good. Since $v_1((A, \varnothing), \boldsymbol{\theta})$ is always higher than this, agent 1 will thus lie and always state a value of $\theta_1 < 6$. This problem can again be shown to extend to be symptomatic of any mechanism rather than our mechanism [Dasgupta and Maskin, 2000]. Notice that with the original valuations in table 5.7.1, such a situation would not arise.

We next consider valuations that break assumption 5.3. Here the valuations of agents 1 and 2 for the allocation $K = (AB, \varnothing)$ are $v_1'((AB, \varnothing), \boldsymbol{\theta}) = 0.5\theta_1 + 2\theta_2$ and $v_2'((AB, \varnothing), \boldsymbol{\theta}) = 3.5\theta_1$ as shown in figure 5.5[6]. Since $v_{\mathcal{I}}$ remains the same for all the allocations, then z_i^{-1} is still the same as shown in figure 5.5. Using these modified valuations, agent 1 derives a higher utility of 1.75 (using equation 5.7 and the valuation function) if it reports $\theta_i > 2$ thereby leading to the mechanism no longer being incentive-compatible. The reason this occurs is because if assumption 5.3 is broken we then have that $v_i(K_{-1}^i, \boldsymbol{\theta}) - v_i(K_0^*, \boldsymbol{\theta}) > v_{-i}(K_{-1}^i, z_i^{-1}\boldsymbol{\theta_{-i}}) - v_{-i}(K_0^*, z_i^{-1}\boldsymbol{\theta_{-i}})$ as shown in figure 5.5. As a result, the agent has an incentive to lie and quote a higher value than z_i^{-1}. Notice that this did not occur with the original valuations. Again this assumption is required in order to find an efficient, incentive-compatible mechanism and is thus not idiosyncratic to our mechanism [Dasgupta and Maskin, 2000].

Having thus illustrated the working of our mechanism and the necessity of the assumptions via the use of an example, we now turn to formally proving the properties of our mechanism.

5.8 Properties of the Mechanism

We next prove the properties of our mechanism. We first consider the economic properties; namely that it is incentive-compatible, efficient and strategy proof, whilst intuitively explaining

[6]Of course, in practice, agent 2 having a valuation for nothing is highly unlikely to occur. However, we need to use this particular valuation in this case due to the simplicity of our example in order to demonstrate what happens when one of the assumptions fails.

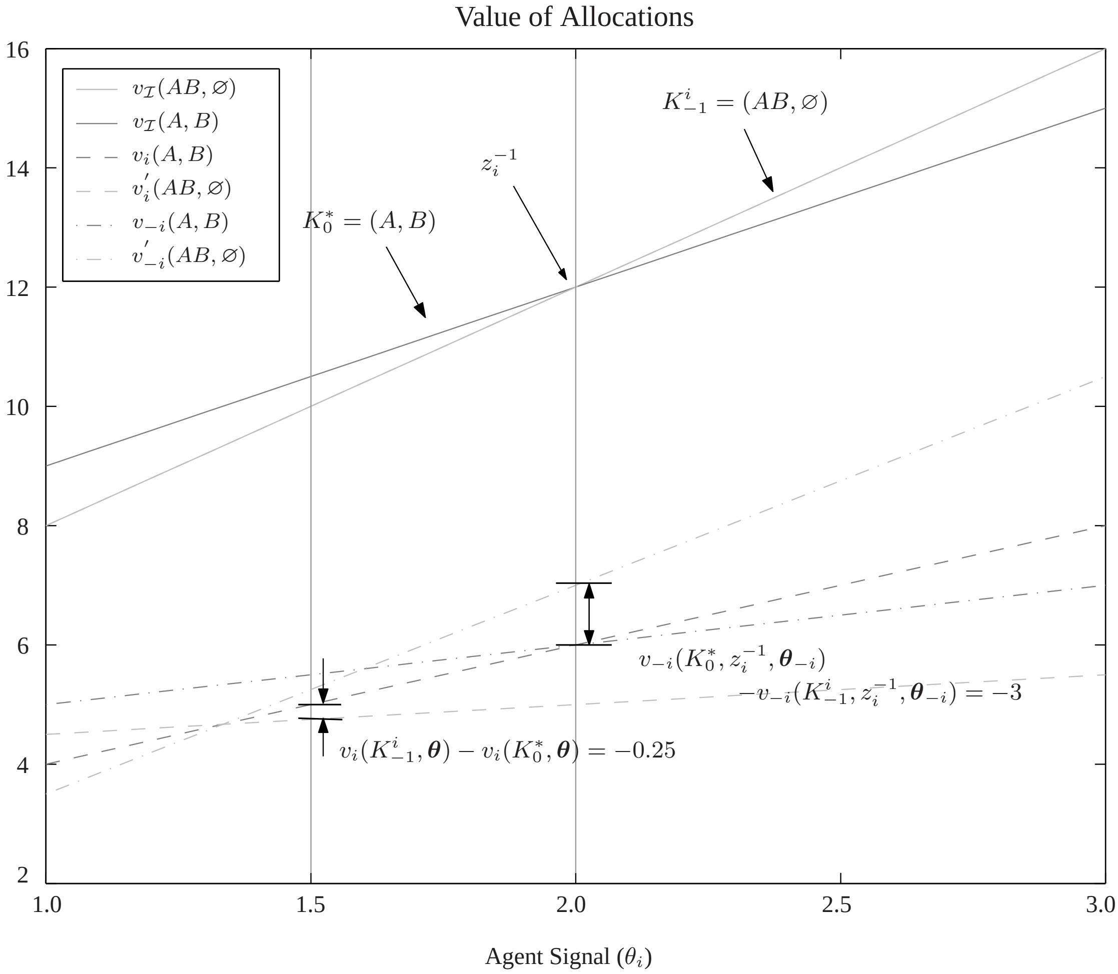


FIGURE 5.5: Modified valuations of $1, 2$ and $\mathcal{I}$ for allocations $(AB, \varnothing)$ and (A,B) as θ_1 is increased

why the mechanism has the aforementioned properties. We then consider the computational properties, showing that the mechanism does not impose any added computational burden on the agents' bidding process (compared to what it would already face in an independent value scenario). However, it does increase the amount of computation required in the calculation of the payment, a computational load borne by the centre.

5.8.1 Economic Properties

Proposition 5.1. *The mechanism is incentive-compatible in ex-post Nash Equilibrium.*

Proof. Let $v_{-i}(.) = \sum_{j \in -i}(v_j(.))$ and $v_{\mathcal{I}}(.) = \sum_{i \in \mathcal{I}}(v_i(.))$. Suppose now that all players except i report their signals truthfully (i.e. $\widehat{\boldsymbol{\theta}}_{-i} = \boldsymbol{\theta}_{-i}$). Let the optimal allocation when i reports truthfully be K_0^*. We can then analyse the utility $u_i(.)$ that agent i derives by reporting a certain $\widehat{\theta}_i$. There are two cases that should be analysed, namely when $\widehat{\theta}_i < \theta_i$ and $\widehat{\theta}_i > \theta_i$. The utility of an agent on reporting $\widehat{\theta}_i = \theta_i$ is:

$$u_i(K_0^*, \boldsymbol{\theta}) = v_i(K_0^*, \boldsymbol{\theta}) + \sum_{l=0}^{m-1} \Big(v_{-i}(K_l^i, z_i^l, \boldsymbol{\theta_{-i}}) - v_{-i}(K_{l+1}^i, z_i^l, \boldsymbol{\theta_{-i}}) \Big) \tag{5.9}$$

Now suppose an agent reports $\widehat{\theta}_i \neq \theta_i$ but this does not change the optimal allocation K_0^* implemented. Then, $u_i(K_0^*, \boldsymbol{\theta}) = u_i(K_0^*, \widehat{\theta}_i, \boldsymbol{\theta_{-i}})$. This is because the agent will derive the same value $v_i(K_0^*, \boldsymbol{\theta})$ if the allocation does not change and the payment will be the same as the signals $z_i^0 \dots z_i^m$ computed by the centre. Now consider the case that an agent reports $\widehat{\theta}_i < \theta_i$ such that this changes the allocation. Then some other optimal allocation, which is necessarily one of the allocations $K_1^i, \dots, K_m^i$, is implemented. Denoting the resulting allocation when $\widehat{\theta}_i < \theta_i$ as K_n^i (i.e. $z_i^n < \widehat{\theta}_i \leq z_i^{n-1}$), the utility that the agent gets from this new allocation is then:

$$u_i(K_n^i, \boldsymbol{\theta}) = v_i(K_n^i, \boldsymbol{\theta}) + \sum_{l=n}^{m-1} \Big(v_{-i}(K_l^i, z_i^l, \boldsymbol{\theta_{-i}}) - v_{-i}(K_{l+1}^i, z_i^l, \boldsymbol{\theta_{-i}}) \Big) \tag{5.10}$$

The difference, $D_n = u_i(K_0^i, \boldsymbol{\theta}) - u_i(K_n^i, \boldsymbol{\theta})$ between truthful reporting and under reporting (as given by equations 5.9 and 5.10 respectively) is:

$$\begin{aligned} D_n &= v_i(K_0^*, \boldsymbol{\theta}) - v_i(K_n^i, \boldsymbol{\theta}) + \sum_{l=0}^{n-1} \Big(v_{-i}(K_l^i, z_i^l, \boldsymbol{\theta_{-i}}) - v_{-i}(K_{l+1}^i, z_i^l, \boldsymbol{\theta_{-i}}) \Big) \\ &= v_i(K_0^*, \boldsymbol{\theta}) + v_{-i}(K_0^*, z_i^0, \boldsymbol{\theta_{-i}}) - v_{-i}(K_n^i, z_i^n, \boldsymbol{\theta_{-i}}) - v_i(K_n^i, \boldsymbol{\theta}) \\ &\quad + \sum_{l=1}^{n} \Big(v_{-i}(K_l^i, z_i^l, \boldsymbol{\theta_{-i}}) - v_{-i}(K_l^i, z_i^{l+1}, \boldsymbol{\theta_{-i}}) \Big) \end{aligned}$$

Since $\frac{\partial v_{-i}(K_l^i, \boldsymbol{\theta})}{\partial \theta_i} \geq 0$, we thus have:

$$D_n > v_i(K_0^*, \boldsymbol{\theta}) + v_{-i}(K_0^*, z_i^0, \boldsymbol{\theta_{-i}}) - v_{-i}(K_n^i, z_i^n, \boldsymbol{\theta_{-i}}) - v_i(K_n^i, \boldsymbol{\theta})$$

Now, we can recast the above as:

$$D_n > v_i(K_0^*, \boldsymbol{\theta}) - v_i(K_0^*, z_i^0, \boldsymbol{\theta_{-i}}) - v_i(K_n^i, \boldsymbol{\theta}) + v_i(K_n^i, z_i^n, \boldsymbol{\theta_{-i}}) + v_{\mathcal{I}}(K_0^*, z_i^0, \boldsymbol{\theta_{-i}}) - v_{\mathcal{I}}(K_n^i, z_i^n, \boldsymbol{\theta_{-i}})$$

However, by construction we know that $v_{\mathcal{I}}(K_0^*, z_i^0, \boldsymbol{\theta_{-i}}) > v_{\mathcal{I}}(K_n^i, z_i^n, \boldsymbol{\theta_{-i}})$ and from assumption 5.3 we also know that $v_i(K_0^*, \boldsymbol{\theta}) - v_i(K_0^*, z_i^0, \boldsymbol{\theta_{-i}}) > v_i(K_n^i, \boldsymbol{\theta}) - v_i(K_n^i, z_i^n, \boldsymbol{\theta_{-i}})$. We thus have $D_n \geq 0$.

On the other hand, if an agent reports $\widehat{\theta}_i > \theta_i$ and this induces an allocation K_{-n}^i, then the utility it derives is:

$$u_i(K_{-n}^i, \boldsymbol{\theta},) = v_i(K_{-n}^i, \boldsymbol{\theta},) + \sum_{l=-n}^{m-1} \Big(v_{-i}(K_l^i, z_i^l, \boldsymbol{\theta_{-i}},) - v_{-i}(K_{l+1}^i, z_i^l, \boldsymbol{\theta_{-i}},) \Big) \tag{5.11}$$

The difference, $D_{-n} = u_i(K_0^*, \boldsymbol{\theta}) - u_i(K_{-n}^i, \boldsymbol{\theta})$ between truthful reporting and under reporting (as given by equations 5.9 and 5.10 respectively) is:

$$
\begin{aligned}
D_{-n} &= v_i(K_0^*, \boldsymbol{\theta}) - v_i(K_{-n}^i, \boldsymbol{\theta}) - \sum_{l=-n}^{-1} \Big(v_{-i}(K_l^i, z_i^l, \boldsymbol{\theta_{-i}}) - v_{-i}(K_{l+1}^i, z_i^l, \boldsymbol{\theta_{-i}}) \Big) \\
&= v_i(K_0^*, \boldsymbol{\theta}) - v_i(K_{-n}^i, \boldsymbol{\theta}) - \sum_{l=-n}^{-1} \Big(v_{\mathcal{I}}(K_l^i, z_i^l, \boldsymbol{\theta_{-i}}) - v_{\mathcal{I}}(K_{l+1}^i, z_i^l, \boldsymbol{\theta_{-i}}) \Big) \\
&\quad + \sum_{l=-n}^{-1} \Big(v_i(K_l^i, z_i^l, \boldsymbol{\theta_{-i}}) - v_i(K_{l+1}^i, z_i^l, \boldsymbol{\theta_{-i}}) \Big)
\end{aligned}
$$

Thus:

$$
\begin{aligned}
D_{-n} &= v_i(K_0^*, \boldsymbol{\theta}) - v_i(K_{-n}^i, \boldsymbol{\theta}) + \sum_{l=-n}^{-1} \Big(v_i(K_l^i, z_i^l, \boldsymbol{\theta_{-i}}) - v_i(K_{l+1}^i, z_i^l, \boldsymbol{\theta_{-i}}) \Big) \\
&= v_i(K_{-n}^i, z_i^{-n} \boldsymbol{\theta_{-i}}) - v_i(K_{-n}^i, \boldsymbol{\theta}) - v_i(K_0^*, z_i^{-1} \boldsymbol{\theta_{-i}}) + v_i(K_0^*, \boldsymbol{\theta}) \\
&\quad - \sum_{l=-n+1}^{-1} \Big(v_i(K_l^i, z_i^{l-1}, \boldsymbol{\theta_{-i}}) - v_i(K_l^i, z_i^l, \boldsymbol{\theta_{-i}}) \Big)
\end{aligned}
$$

Using assumption 5.3 implies that $D_{-n} \geq 0$. We thus see that i derives highest utility when reporting $\widehat{\theta}_i = \theta_i$. □

Proposition 5.2. *The mechanism is efficient.*

This implies that the centre finds the outcome such that:

$$
K^* = \arg\max_K \sum_{i \in \mathcal{I}} v_i(K, \boldsymbol{\theta}) \tag{5.12}
$$

Note that this is different from equation 2.10 in that in this case, we allow the valuations of the agents to depend on the vector of all types $\boldsymbol{\theta}$, as opposed to only the type observed privately by an agent θ_i.

Proof. The above is a result of the incentive-compatibility of the mechanism. Since the goal of the centre is to achieve efficiency, then given truthful reports, the centre will achieve efficiency. □

Proposition 5.3. *The mechanism is individually rational (as defined in chapter 2).*

We begin by assuming that the utility an agent derives from not joining the mechanism is 0. Then, we need to prove that the utility an agent derives in the mechanism is always ≥ 0.

Proof. Given that the agents are incentivized to report truthfully, an agent i derives utility:

$$\begin{aligned} u_i(K_0^*, \boldsymbol{\theta}) &= v_i(K_0^*, \boldsymbol{\theta}) + \sum_{l=0}^{m-1} \Big(v_{-i}(K_l^i, z_i^l, \boldsymbol{\theta_{-i}}) - v_{-i}(K_{l+1}^i, z_i^l, \boldsymbol{\theta_{-i}} \Big) \\ &= v_i(K_0^*, \boldsymbol{\theta}) + \sum_{l=0}^{m-1} \Big(v_{\mathcal{I}}(K_l^i, z_i^l, \boldsymbol{\theta_{-i}}) - v_{\mathcal{I}}(K_{l+1}^i, z_i^l, \boldsymbol{\theta_{-i}}) \Big) \\ &\quad - \sum_{l=0}^{m-1} \Big(v_i(K_l^i, z_i^l, \boldsymbol{\theta_{-i}}) - v_i(K_{l+1}^i, z_i^l, \boldsymbol{\theta_{-i}}) \Big) \end{aligned} \tag{5.13}$$

Since $v_{\mathcal{I}}(K_l^i, z_i^l, \boldsymbol{\theta_{-i}}) = v_{\mathcal{I}}(K_{l+1}^i, z_i^l, \boldsymbol{\theta_{-i}})$ (from equation 5.5):

$$\begin{aligned} u_i(K_0^*, \boldsymbol{\theta}) &= v_i(K_0^*, \boldsymbol{\theta}) - \sum_{l=0}^{m-1} \Big(v_i(K_l^i, z_i^l, \boldsymbol{\theta_{-i}}) - v_i(K_{l+1}^i, z_i^l, \boldsymbol{\theta_{-i}}) \Big) \\ &= v_i(K_0^*, \boldsymbol{\theta}) - v_i(K_0^*, z_i^0, \boldsymbol{\theta_{-i}}) + v_i(K_m^i, z_i^m, \boldsymbol{\theta_{-i}}) \\ &\quad + \sum_{l=1}^{m-1} \Big(v_i(K_l^i, z_i^l, \boldsymbol{\theta_{-i}}) - v_i(K_l^i, z_i^{l+1}, \boldsymbol{\theta_{-i}}) \Big) \end{aligned} \tag{5.14}$$

From equation 5.6, $v_i(K_m^i, z_i^m, \boldsymbol{\theta_{-i}}) = 0$. Now, since $\frac{\partial v_i(K, \boldsymbol{\theta})}{\partial \theta_i} \geq 0$, $u_i(K_0^*, \boldsymbol{\theta}) > 0$. □

5.8.2 Computational Properties

In order for a mechanism to be of use in real world scenarios, we must not only consider its economic properties, but also its computational porperty. An important distinction which was pointed out in chapter 1 is to differentiate between the computational load that is imposed on the agents within the auction and that imposed on the auctioneer or centre. We will analyse the computational properties of the mechanism as opposed to that faced by agents in a VCG mechanism. In so doing, we aim to quantify the computational cost that the added richness of this mechanism (namely the ability to express interdependent valuation) imposes.

Outcome Determination. In our mechanism, the centre will need to solve equation 5.4 as opposed to 2.8 in the VCG mechanism in order to determine the efficient allocation. In both cases the computation involves solving the combinatorial allocation problem which is a NP-hard combinatorial optimisation [Rothkopf et al., 1998]. In fact, the size of the set over which the optimisation is carried out is the same in both cases since this is determined by the number of items $|M|$. Thus our mechanism imposes no additional computational load in terms of the centre calculating the allocation.

However, in terms of calculating the payments to the agents, our mechanism does impose a much larger computational load. In the case of the VCG mechanism, calculating the payment involves performing the winner determination problem $|\mathcal{I}|$ times over the reduced set of agents $\mathcal{I} \setminus i$ (see equation 2.8). However in our case, the centre needs to successively reduce the value of the report from each agent (and calculate the optimal allocation at each stage) until it reaches an allocation which is the optimal one for the reduced set of agents $\mathcal{I} \setminus i$ (see equation 5.7). In the worst case scenario, we have to traverse through all possible allocations (except the efficient one) when calculating the different z_i^l for each agent $i \in \mathcal{I}$. For m goods in a combinatorial auction, this requires $2^m - 1$ calculations and is thus exponential in complexity. However, typically the number of allocations that need to be traversed (i.e the K_l^i) will be less than 2^m and there is some redundancy between the calculation of the K_l^i in between the agents in $\mathcal{I}$. We will exploit this redundancy in future work so as to reduce the computational load on the centre.

Preference Formulation. In the case of a direct mechanism such as the VCG or our mechanism, the agents do not have additional computational load in formulating their preferences over all possible outcomes. This is because the agents transmit their observed signal θ_i to the centre and thus do not actually compute $v_i(K, \boldsymbol{\theta})$ over all $K \in \mathcal{K}$. Rather, it is the centre which performs this calculation for each agent when solving the winner determination problem. Thus, our mechanism in this case does not add any computational load on the agents.

Strategy Selection. In the VCG mechanism the agent knows *a priori* that it has a dominant strategy, and thus this computational problem does not arise. In our case, an agent has an *ex-post* Nash strategy. Thus if all the agents are behaving rationally, there is no computational load on the agent in this particular case. However, if it becomes common knowledge that some agent is not playing its best-response strategy (i.e. some agent is not rational) then the agents will have to search through their space of strategies again to find their best-response.

Thus, we can observe that there is no additional computational load on the agents when compared to a standard VCG mechanism and thus we can use the computationally efficient bidding languages developed for VCG mechanisms [Parkes, 2001; Nisan, 2000]. This is important since it is conceivable that while the centre in a multi-sensor network may have enough computational power, this is not necessarily so for the individual sensors that will typically be much more limited.

5.9 Summary

In this chapter, we have first developed a utility function for sensors in our MSN scenario based on the information form of the Kalman filter. Since these utility function exhibit interdependence, we could not use standard resource allocation mechanisms. We thus developed a generic mechanism for interdependent valuations which significantly extends the standard VCG mechanism and proved that the ensuing mechanism has the ideal economic properties of being efficient, incentive compatible and individually rational. Our mechanism is general and reduces to the VCG mechanism whenever there are independent valuations (as seen in section 5.7). Thus, we can visualise our mechanism being used even in MAS where the designer is unsure whether the valuations are interdependent or not. Finally, we analysed the computational complexity of implementing the mechanism and compared it to the complexity of implementing its closest equivalent (in the private value case), namely the VCG mechanism.

Whilst we have presented our mechanism in terms of resource allocation, it can be easily converted into a task allocation scenario. In such a case, agents will first submit cost functions instead of valuation functions. Then, we need to perform a minimisation instead of a maximisation in equations 5.4, 5.5 and 5.6 and take supremums instead of infimums in equations 5.5 and 5.6. With these changes, the mechanism still conserves both its computational and economic properties in the task allocation scenario.

In sum, this chapter has considered an important class of auctions in which the bidders have interdependent valuations (based on a single dimensional signal measured by each bidder) and bid for multiple goods. In so doing, requirement 3 from Chapter 1 (namely the requirement for mechanisms that deal with distributed information) has been addressed. In the next chapter, we shall consider a particular type of distributed information that arises when agents depend on other agents' reports in order to gauge the success rate of task providers.

Chapter 6

Mechanisms with Uncertainty in Task Completion

In this chapter, we incorporate the uncertainty that an agent may face in completing its assigned task into the design of the task allocation mechanism (which is the problem that requirement 4 seeks to address). Such uncertainty was briefly discussed in Chapter 4, when an agent was only aware of the probability distribution over its capacity. However, this chapter goes further in exploring uncertainties that can occur when an agent's success is dependent upon the task to which it is assigned. In more detail, for each set of tasks, task performers have an associated *Probability of Success* (POS) which determines the probability that it successfully completes the task. Now, the agents may have differing views on how successful a particular agent is in offering a certain task. This differing view is termed the *trust* that an agent has on another agent. Furthermore, agents communicate between themselves their perceptions of the POS of the tasks offered by other agents. This allows an agent to form its own perception about the POS of other agents from its own experience, as well as from reports from other agents. Hence, we have a distributed information environment since the information required to formulate an agent's expected valuation (what it expects to obtain before the task is attempted) is distributed amongst potentially many agents. Hence, this chapter also addresses requirement 1 which is to develop mechanisms that deal with distributed information.

The remainder of this chapter is structured as follows. Section 6.1 introduces the research on uncertainty in task completion and explains its importance in general MAS settings. We then put this research into the context of the running MSN scenario in Section 6.2. Section 6.3 discusses related work in the areas of trust and mechanism design. In Section 6.4, we explain the general task allocation problem that we seek to tackle. Section 6.5 then explain the generic properties that such a trust model should incorporate so as to lead to efficient allocations in our setting. We also demonstrate via an example why the VCG mechanism fails when considering uncertainty in task allocation. We then present our mechanism and prove its properties in Section 6.6. In

Section 6.7, we demonstrate the generality of our mechanism. We then discuss the computational aspects of implementing our mechanism in Section 6.8 and then go on and empirically evaluate it in Section 6.9. Finally, the main contributions of this Chapter are summarised in Section 6.10.

6.1 Introduction

In this chapter, we challenge the assumptions made in traditional MD that an agent always completes every task it starts or it does not default on payment for a good. The result of this assumption is that an agent chooses to interact with partners based on their costs or valuations only. However, cheapest is not always best and these agents may ultimately not be the most successful. For example, in the MSN scenario we study in more detail in section 6.2, sensors may decide to pay more for a service from a sensor which is more likely to provide a good quality of data more reliably. Thus, in many practical situations, the choice of interaction partners is motivated by an agent's individual model of its counterparts, as well as by information gathered from its environment about them. For example, on eBay, buyers determine the credibility of particular sellers by considering their own interaction experiences with them (if they have any) and by referring to the historic evaluation information provided by other buyers. To capture this phenomenon, we exploit the notion of *trust* to represent an agent's perception of another agent's *probability of success* (POS) in completing a task [Dasgupta, 1998]. This, in turn, leads us to propose the area of *trust-based mechanism design* (TBMD) as an extension of traditional MD that adds trust as an additional factor to costs and valuations in decision making.

In more detail, the trust in an agent is generally defined as the expectation that it will fulfill what it agrees to do, given its observable actions and information gathered from other agents about it [Dasgupta, 1998] [1]. By their very nature, different agents are likely to hold different opinions about the trust of a particular agent depending on their experiences and the specifics of the trust model they use [Ramchurn et al., 2004]. As a result, we cannot simply extend the conventional MD solutions (e.g. the VCG mechanism) to encompass the notion of trust because such work is predicated on the fact that agents have *private and independent* information which determines their choice over outcomes. Trust, on the other hand, implies *public and interdependent information* (in the sense discussed in Chapter 5). For example, our trust in a seller in a market would result from other agents in the market telling us about the seller's output quality (efficiency), combined with our own notion of the seller's output quality. A high degree of trust in the seller's efficiency would mean that we believe that the seller is highly efficient, while a low value indicates that we believe it will not be efficient.

In this work, we specifically consider MD in the context of task allocation (where it has often been applied [Sandholm, 2003; Rosenschein and Zlotkin, 1994]). Specifically, in our scenario,

[1] The term "trust"' has also been used in connection with the dependability of information about other agents [Ramchurn et al., 2004].

agents may have different probabilities of success in completing a task assigned to them (e.g. it may be believed that a particular builder has a 95% chance of making a roof in five days, while another one may be believed to have a 75% chance of doing so). Moreover, an agent may assign different weights to the reports of other agents depending on the similarity of their types. For example, consider a "repair engine" task assigned to a garage. In this case, two agents owning a Ferrari would be likely to assign higher weights to each other's report about the POS of the garage than they would to the report of another agent which owns a Robin Reliant.

Against this background, this chapter develops and evaluates the notion of trust-based mechanism design. In doing so, we advance the state of the art in the following ways:

1. We first define the general properties that trust models must exhibit to allow a trust-based mechanism (TBM) to generate an optimal allocation of tasks.
2. We extend the standard VCG mechanism in order to deal with uncertainties in task completion.
3. We prove the economic properties of our TBM and show that it is incentive-compatible, efficient and individually-rational.
4. We study the computational properties of our mechanism. Specifically we show that the task allocation problem is $\mathcal{NP} - complete$ and develop algorithms based on dynamic programming for the generation of possible allocations and pruning of the search space.
5. We also empirically evaluate our mechanism when faced with seller's bias (i.e. the seller is biased concerning its POS) and show that our mechanism achieves the efficient allocation in the long run.

We now detail the MSN scenario from which the requirement of addressing uncertainty in task completion is inspired.

6.2 Uncertainty in Task Completion Within the MSN Scenario

This section discusses how the research question we address in this chapter, namely uncertainty in task completion, can arise in the MSN scenario (as highlighted in red in figure 6.1). To this end, consider the multiple selfish sensors which can be tasked with monitoring a certain region which is of interest to other sensors within the MSN. Now, these sensors may provide different qualities of service depending on a number of factors such as the hardware on which they are based, the immediate environment in which they are situated and the state of their hardware (e.g. whether it is faulty or not). Furthermore, the sensors may have different costs in actually completing their tasks which may be due to the different hardware they utilise or the different amount of time they spend in fulfilling the service. As a result, the sensors requesting the tasks may choose to pay a premium for a better POS of the task being completed.

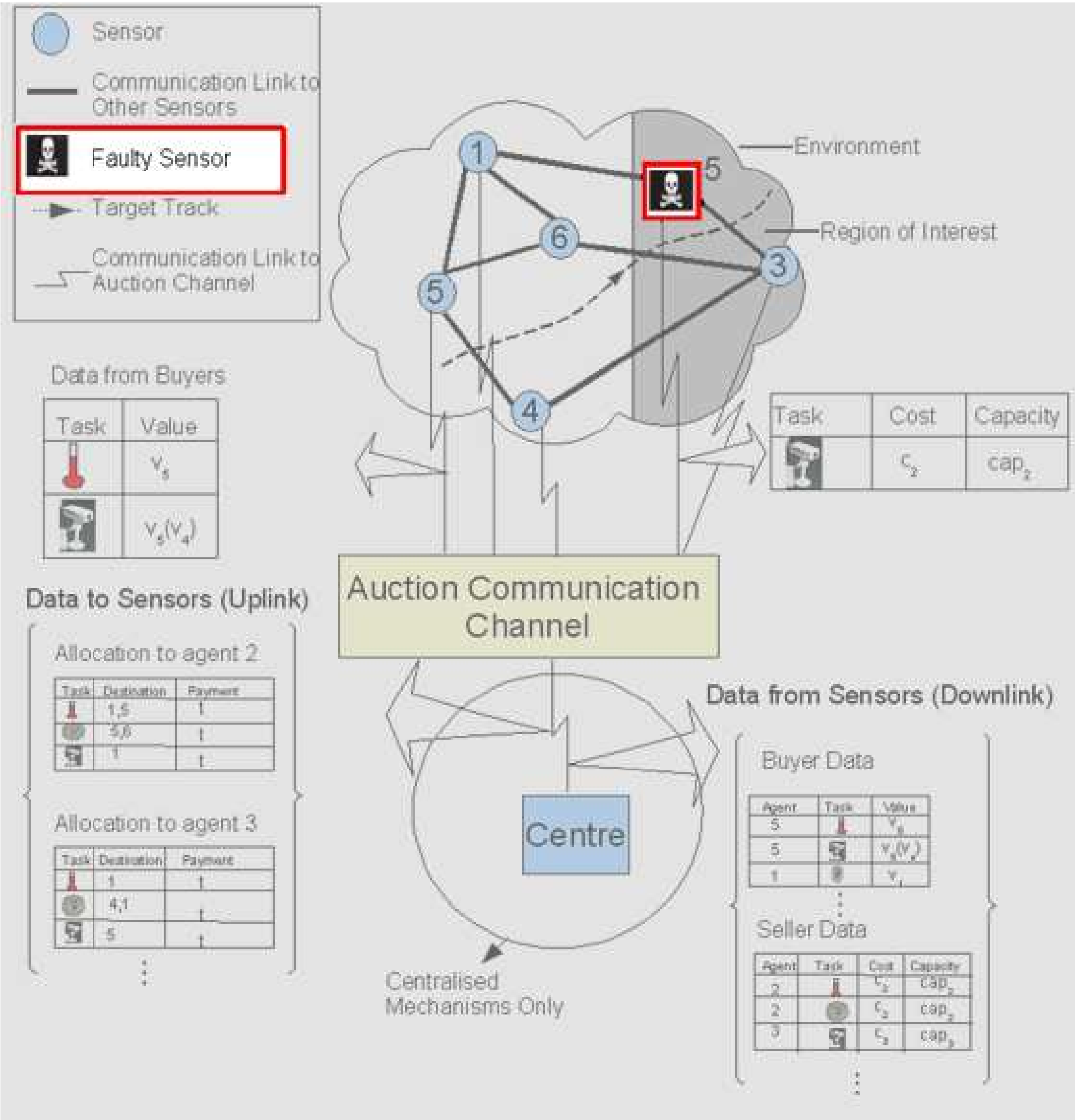


FIGURE 6.1: Figure of MSN scenario highlighting faulty sensors which are considered within this chapter

Now, the sensor may not be aware of the POS of another sensor, since there may not have been sufficient interactions between them for the sensor to learn the POS of the other sensor. Then, it may query other sensors that have had previous experience with the particular service provider so as to gauge its POS. However, different sensors may require data for different reasons and thus place different ratings on the POS provided by a provider. For example, a sensor interested in environment monitoring may impose less restrictive quality levels on the visual data provided than one which is involved in target tracking. As a result, it may rate the POS of a particular sensor at a much higher level than another sensor. Therefore the sensors will also have to learn to judge the levels of importance to place on the ratings and experiences provided by other sensors (i.e. it will have to develop a *trust model*).

6.3 Related Work

In associating trust to mechanism design, we build upon work in both areas. In the area of trust and reputation, a number of computational models have been developed (see [Ramchurn et al., 2004] for a review). While these models can help in choosing the most successful agents, they are not shown to generate efficient outcomes in any given mechanism. An exception to this is the work on reputation mechanisms [Dellarocas, 2002; Jurca and Faltings, 2003]. However, these mechanisms only produce efficient outcomes in very constrained scenarios and under strict assumptions (e.g. in [Dellarocas, 2002] sellers are monopolists and each buyer interacts at most once with a seller and in [Jurca and Faltings, 2003] the majority of agents must already be truthful for the mechanism to work[2]).

In the case of MD, there has been comparatively little work on achieving efficient, incentive-compatible and individually-rational mechanisms that take into account *uncertainty* in general. An exception to this rule is the dAGVA mechanism (see section 2.4) which considers the case when the types of agents are unknown to themselves, but are drawn from a probability distribution of types which is common knowledge to all agents. However, in our case, the agents know their types and these incorporates their uncertainty related to fulfilling a task. Porter et al. [2002] have also considered this case and their mechanism is the one that is most closely related to ours. However, they limit themselves to the case where agents can only report on their own POS. This is a drawback because it assumes the agents can measure their own POS accurately and it does not consider the case where this measure may be biased (i.e. different agents perceive the success of the same event differently). Thus our mechanism is a generalisation of theirs (see section 6.7).

Finally, our work may also seem to be a case of interdependent, multidimensional allocation schemes [Dasgupta and Maskin, 2000] where there is an important impossibility result of not being able to achieve efficiency when considering interdependent, multidimensional signals (see Chapter 5 for a more detailed discussion on interdependent valuations). However, we circumvent this by first relating the trust values to a probability that an allocation is completed, rather than to an absolute valuation or cost signal and second by achieving an ex-ante equilibrium rather than the stronger ex-post equilibrium.

6.4 The Allocation Problem

We now discuss in more detail the problem structure that we consider in the remainder of this chapter. The system that we wish to control consists of a set of agents $\mathcal{I} = \{1, \ldots, I\}$ that are requesting tasks from a set of atomic tasks $\mathcal{T} = \{\tau_1 \ldots, \tau_n\}$ to be performed for them.

[2]This is different from a best-response Nash strategy of truthful reporting since the majority of agents are known a priori to be truthful and thus they can be counted upon to report truthfully even if it is not rational for them to do so.

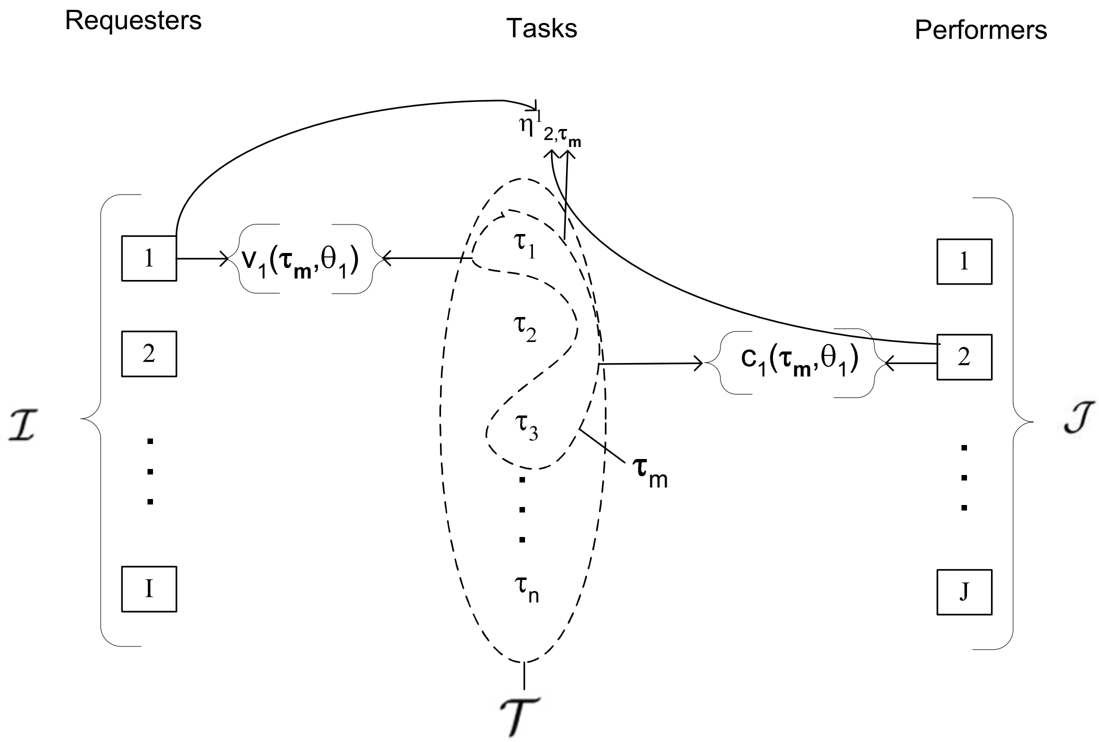


FIGURE 6.2: Graphical depiction of the allocation problem studied with this chapter.

We shall call these agents *task requesters*. Furthermore, there is another set of agents, called *task providers*, $\mathcal{J} = \{1, \ldots, J\}$ that can perform these tasks[3]. Now, define the set $2^{\mathcal{T}} = \{\boldsymbol{\tau}_1, \ldots, \boldsymbol{\tau}_m, \ldots, \boldsymbol{\tau}_{2^n-1}\}$ as the power set of $\mathcal{T}$. Then, a task performer j would have a cost $c_j(\boldsymbol{\tau}_m, \theta_j)$ for performing the set of tasks $\boldsymbol{\tau}_m$. Furthermore, dependent upon its capabilities and constraints, task provider j would perform the set of tasks $\boldsymbol{\tau_m}$ requested by agent i to a certain POS level. Let $\gamma^i_{j,\boldsymbol{\tau_m}}$ be an indicator function denoting whether the task $\boldsymbol{\tau_m}$ requested by agent i and performed by agent i has been deemed successful by agent i. Thus:

$$\gamma^i_{j,\boldsymbol{\tau_m}} = \begin{cases} 1 & \text{if } \boldsymbol{\tau_m} \text{ is evaluated by agent } i \text{ as successfully completed} \\ 0 & \text{otherwise} \end{cases}$$

We shall assume that the cost incurred by a task provider is independent of whether it has been successful or not in completing the task. Furthermore, a task requester i would have a valuation $v_i(\boldsymbol{\tau}_m, \theta_i)$ for a set of tasks $\boldsymbol{\tau_m}$ when those tasks are successfully performed. Otherwise, it derives a value of 0. Figure 6.2 shows a graphical depiction of the problem structure we deal with in this chapter.

In more detail, each task provider has a cost vector $\boldsymbol{c}_j$ that specifies the cost it incurs for different sets of tasks. Similarly, each task requester has a valuation vector $\boldsymbol{v}_i$ that specifies the value it derives for different sets of tasks when these tasks are performed for it at a POS of 1. Then, given

[3]It is naturally possible for task requesters to be task performers as well (i.e. $\mathcal{I} \cap \mathcal{J} \neq \emptyset$). However, we shall present them as different sets since this clarifies the explanation.

the set of values, $V = \{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_I\}$, and costs, $C = \{\boldsymbol{c}_1, \ldots, \boldsymbol{c}_J\}$, an allocation K (amongst the set of possible allocations $\mathcal{K}$) matches task requesters to the task providers by specifying which requested tasks are performed and which task providers perform them (i.e. $\mathcal{K} : \mathcal{I} \times \mathcal{T} \rightarrow \mathcal{J}$). Once the tasks have been completed and the POS levels have been determined, we can then calculate the overall value (i.e system value), $U(K, \theta, \boldsymbol{\gamma})$, of an allocation as:

$$U(K, \theta, \boldsymbol{\gamma}) = \sum_{i=1}^{I} v_i(K, \theta, \boldsymbol{\gamma^i}) - \sum_{j=1}^{J} c_j(K, \theta_j) \tag{6.1}$$

where $v_i(K, \theta, \boldsymbol{\gamma^i})$ is the value that task requester i derives when the set of tasks specified by K are completed to the POS level given by $\boldsymbol{\gamma^i}$ and $c_j(K, \theta_j)$ is the cost incurred by task performer j when it performs the tasks specified by K. Once a certain allocation K has been decided, an agent i is then asked to pay for the task(s) it requested (if they are implemented in K), whereas an agent j receives payment for the task(s) it has performed. Let the overall transfer of money to a particular agent i be denoted by $r_i \in \Re$. As is common in this domain, we assume that an agent is rational (expected utility maximiser) and has a quasi-linear utility function (see Chapter 2 for more details). Then a task requester has a utility given by:

$$u_i(K, r_i, \theta_i, \boldsymbol{\gamma^i}) = v_i(K, \theta_i, \boldsymbol{\gamma^i}) + r_i \tag{6.2}$$

and a task performer has a utility given by :

$$u_i(K, r_j, \theta_j) = c_j(K, \theta_j) + r_j \tag{6.3}$$

The problem at hand is then to find a mechanism that fulfills the following commonly sought objectives in MD (as discussed in Chapter 2):

- *Efficiency*: an allocation K^* that maximises the total utility of all the agents in the system.
- *Individual Rationality*: an allocation scheme that ensures agents are willing to participate rather than opt out (i.e. $u_i \geq 0$).

Now, in the traditional case, $\gamma^i_{j,\boldsymbol{\tau_m}}$ is always assumed to be equal to 1. In this case, one can use the VCG mechanism described in chapter 2 for the task allocation problem. In effect, our task allocation problem is then reduced to the following protocol which is shown in figure 6.3[4]:

[4] We should here note that in the above scheme, an agent is reporting its cost and valuation rather than its type. Though technically it is thus not a direct mechanism (since the agents do not report their types), it can easily be converted into a direct scheme by having the agents report the two dimensions of its type instead of the cost and valuation.

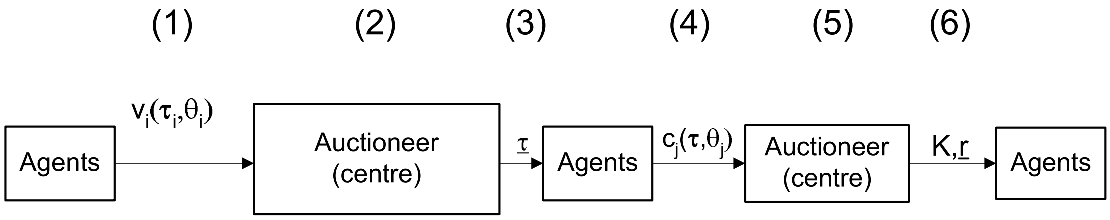


FIGURE 6.3: Task allocation model without uncertainty in task completion.

1. The centre receives the set of tasks $\boldsymbol{\tau_m}$ to be allocated from the task requesters along with their *reported* valuations $\widehat{v}_i(\boldsymbol{\tau_m}, \theta_i)$ for each set of tasks, they are requesting (step 1 in figure 6.3).

2. The centre then posts these tasks in the vector $\boldsymbol{\tau}$ (step 3). Each task performer j then reports its cost $\widehat{c}_j(\tau_m, \theta_j)$ for completing the sets of tasks (step 4).

3. The centre then solves the following standard VCG auction equation (step 5):

$$K^* = \arg\max_{K \in \mathcal{K}} \left[\sum_{i \in \mathcal{I}} \widehat{v}_i(K, \theta_i) - \sum_{j \in \mathcal{J}} \widehat{c}_j(K, \theta_j) \right] \tag{6.4}$$

   and computes each transfer r_i in the vector $\boldsymbol{r}$ for task requesters as:

$$r_i = \sum_{l \in -i} \widehat{v}_l(K^*, \theta_j) - \sum_{j \in \mathcal{J}} \widehat{c}_j(K^*, \theta_j) - \max_{K \in \mathcal{K}} \left(\sum_{l \in -i} \widehat{v}_l(K, \theta_l) - \sum_{j \in \mathcal{J}} \widehat{c}_j(K, \theta_j) \right) \tag{6.5}$$

   and for task performers as:

$$r_j = \sum_{i \in \mathcal{I}} \widehat{v}_i(K^*, \theta_i) - \sum_{l \in -j} \widehat{c}_l(K^*, \theta_l) - \max_{K \in \mathcal{K}} \left(\sum_{i \in \mathcal{I}} \widehat{v}_i(K, \theta_i) - \sum_{l \in -j} \widehat{c}_l(K, \theta_l) \right) \tag{6.6}$$

4. The centre allocates the tasks according to the optimal allocation K^* and implements the transfers r_i (step 6).

The VCG mechanism described above thus receives bids and asks from agents and implements the allocation K^* that maximises $\sum_i v_i(K^*, \theta_i) - c_i(K, \theta_i)$. Each task requester makes a payment $v_i(K^*, \theta_i) - (U(\mathcal{I} \cup \mathcal{J}) - U(\mathcal{I} \cup \mathcal{J} \setminus i))$ where $U(\mathcal{I} \cup \mathcal{J})$ is the total utility of K^* and $U(\mathcal{I} \cup \mathcal{J} \setminus i)$ is the total utility of the choice that would be implemented without agent i. Similarly, each task performer would receive a payment (as r_j will be negative) $c_j(K^*, \theta_i) - (U(\mathcal{I} \cup \mathcal{J}) - U(\mathcal{I} \cup \mathcal{J} \setminus j))$ where $U(\mathcal{I} \cup \mathcal{J} \setminus j)$ is the total utility of the choice that would be implemented without agent j. In equilibrium, each agent receives as utility the marginal value that it contributes to the system. This is why the VCG mechanism will be incentive-compatible (as argued in chapter 2) and thus lead to an efficient mechanism. Furthermore, assuming that the utility derived from opting out of the system is zero then it can also be deduced that the VCG is also individually-rational.

Thus in the traditional setting, the VCG mechanism can provide an efficient, individually rational and incentive compatible allocation since the centre can determine the overall utility of an allocation *before* the allocation is actually carried out. In contrast, in our problem setting, the POS is not known *a priori* since the tasks which have been allocated under K are evaluated by the respective task requesters *after* they have been performed. Thus, the ex-post value of an allocation cannot be determined whilst deciding upon the allocation. This implies that one cannot achieve ex-post efficiency in this setting. Instead, the efficiency we aim to achieve is ex-ante efficiency where the *expected* utility is maximised. The expected utility of an allocation is then calculated based upon the perception that an agent has about another agent's POS at fulfilling a certain task. We shall term this perception the *trust* that an agent has in another agent fulfilling a task.

Definition 6.1. Trust. A task requester i has a trust $t^i_{j,\boldsymbol{\tau_m}}$ in agent j if it believes that agent j will fulfill the set of task $\boldsymbol{\tau_m}$ with a POS given by $t^i_{j,\boldsymbol{\tau_m}}$.

Now, as argued in section 6.1, an agent can formulate its trust based on reports from different agents within the system. This is especially relevant when the POS of an agent is viewed differently by the agents within the system and needs to be learned within the system. Thus, typically, the trust of an agent in a particular task provider will be calculated as an amalgamation of the experience that the agent has had with the task provider, as well as reports from other agents about their experience. The variable encapsulating the experience of a particular agent shall be termed the POS measure. Specifically the POS measure is defined as:

Definition 6.2. POS Measure. The POS measure of a task performer j, $\eta^i_{j,\boldsymbol{\tau_m}}$, as measured by a task requester i with respect to the set of tasks $\boldsymbol{\tau_m}$, is the frequency with which agent j has successfully completed $\boldsymbol{\tau_m}$ when it was allocated to perform this task for agent i

Having thus explained the allocation problem that we tackle in this chapter, in the next subsection, we will now expand on the properties that a generic trust model needs to satisfy for an efficient mechanism. We will also show how to augment the task allocation problem to encompass trust measures and demonstrate via an example why a simple extension of the VCG mechanism cannot guarantee efficiency.

6.5 Trust Model Requirements

Many computational trust models have been developed to allow agents to choose their most trustworthy interaction partners (as discussed in section 6.3). However, at their most fundamental level, these models can be viewed as alternative approaches for achieving the following properties[5]:

[5]Note that we do not focus on a particular trust model. This is because trust models implement the above properties in their own ways and in different contexts. Therefore, we concentrate on these abstract properties to keep the focus on the relationship between trust and the design of an efficient mechanism. In so doing, we ensure that the properties of our mechanism are independent of any specific trust model.

1. The trust measure of an agent i in an agent j depends both on i's perception of j's POS and on the perception of other agents on j's POS. This latter point encapsulates the concept of *reputation* whereby the society of agents generally attributes some characteristic to one of its members by aggregating some/all the opinions of its other members about that member. Thus, each agent considers this societal view on other members when building up its own measure of trust in its counterparts [Dasgupta, 1998]. The trust of agent i in its counterpart j with respect to a certain set of tasks $\boldsymbol{\tau_m}$, $t^i_{j,\boldsymbol{\tau_m}} \in [0,1]$, is given by a function, $g : [0,1]^{|\mathcal{I}|} \rightarrow [0,1]$, (which, in the simplest case, is a weighted sum) of all POS measures sent by other agents to agent i about agent j as shown below:

$$t^i_{j,\boldsymbol{\tau_m}} = g(\{\eta^1_{j,\boldsymbol{\tau_m}}, \ldots, \eta^i_{j,\boldsymbol{\tau_m}}, \ldots, \eta^I_{j,\boldsymbol{\tau_m}}\}) \tag{6.7}$$

   where $\eta^i_{j,\boldsymbol{\tau_m}} \in [0,1]$ is the POS of agent j as perceived by agent i with respect to task $\boldsymbol{\tau_m}$ and g is the function that combines both personal measures of POS and other agents' measures. In general, trust models compute the POS measures over multiple interactions. Thus, the level of success recorded in each interaction is normally averaged to give a representative value (see [Ramchurn et al., 2004] for a general discussion on trust metrics).

2. Trust results from an analysis of an agent's POS in performing a given task. The more successful, the more trustworthy the agent is. Thus, the models assume that trust is monotonic increasing with POS. Therefore, the relationship between trust and POS is expressed as: $\frac{\partial t^i_{j,\boldsymbol{\tau_m}}}{\partial \eta_{j,\boldsymbol{\tau_m}}} > 0$.

Given the above, agents can update the trust rating for another agent each time they interact (both by recording their view of the success of their counterpart and by gathering new reports from other agents about it). Thus, if an agent's POS does not change, the trust measure in it should become more precise as more observations are made and received from other agents. Moreover, having the trust monotonic increasing with POS ensures the condition given by Mirrlees [1971] regarding fixed points in allocation schemes is satisfied (this is a necessary condition for the mechanism to be efficient).

6.5.1 Augmenting the Task Allocation Scenario

In this section we show how trust is to be calculated and taken into account in the task allocation problem we described in section 6.4. Here, any trust model satisfying the properties discussed in the previous section can be used when actually building the system. The following changes are made (as shown in figure 6.4):

- Each task requester i and each task requester j reports to the centre their POS vector (i.e. $\widehat{\boldsymbol{\eta}}^{\boldsymbol{i}} = [\widehat{\eta}^i_1 \ldots \widehat{\eta}^i_J]$ and $\widehat{\boldsymbol{\eta}}^{\boldsymbol{j}} = [\widehat{\eta}^j_1 \ldots \widehat{\eta}^j_J]$ (step 1)). This is the POS that an agent has observed about the task performers. This vector may not be complete if agents have not experienced

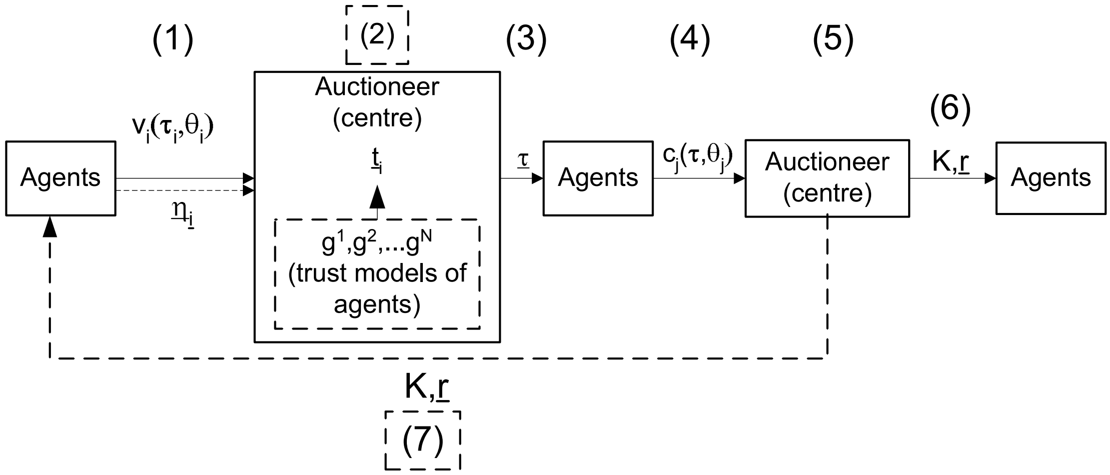


FIGURE 6.4: Trust-based task allocation model. The dotted lines represent the modifications we make to the mechanism when using trust in the feedback loop.

any past interactions with other agents. However, this does not affect the properties of the mechanism since the centre will only pick those POSs that are relevant (and calculate trust according to these).

- The agents must also submit their respective trust calculation function (equation 6.7) that applies over the vector of all (or part of) other agents' reported POSs (i.e. $\widehat{\boldsymbol{\eta}}$), $\boldsymbol{t^i} = g(\widehat{\boldsymbol{\eta}})$[6], to the centre before the allocation of tasks (step 2). This allows the centre to compute the trust of agent i in all other agents (given i's own perception, as well as other agents' perceptions of the task performer's POS). Given that the trust t^i only affects the allocation of tasks being requested by agent i, the latter has no incentive to lie about its trust function to the centre (otherwise it could result in i's task not being allocated to the agent deemed most trustworthy by i).

The trust function $g(.)$ may assign different weights to the reports of different agents depending on the level of similarity between the types of agents i and $-i$. Thus, given the trust functions and reports of POS of each agent, we now require the centre to maximise the overall *expected* valuation of the allocation (in step 5), as opposed to the valuation of the allocation independent of trust (i.e. which the standard VCG does). This is because an agent has a certain probability of completing the task to a degree of success which may be less than one. Thus, the expected value of an allocation is then $\left(E_{[\boldsymbol{\gamma^i}|K,\boldsymbol{t^i}]}\left[\sum_{i\in\mathcal{I}}\widehat{v}_i(K,\theta_i,\boldsymbol{\gamma^i})\right] - \sum_{i\in\mathcal{I}}\widehat{c}_i(K,\theta_i)\right)$ given the trust vector $\boldsymbol{t^i}$. This captures the fact that the agent i, that allocated the task, determines the value of $\boldsymbol{\gamma^i}$ for all tasks which it has requested and been allocated in K. This effectively means that the valuations are non-deterministic, while the costs are deterministic. The centre thus determines the efficient allocation K^* (step 7) such that the value of the efficient allocation is maximised.

[6]We drop the task subscript of the trust and POS variable when the task is not relevant in the explanation.

TABLE 6.1: A set of four agents in which agent 4 has proposed a task.

Agent i	c_i	η_1^i	η_2^i	η_3^i	t_i^4
1	40	0.4	1.0	0.8	0.5
2	80	0.6	1.0	0.8	1.0
3	50	0.5	1.0	0.9	0.86
4	∞	0.525	1.0	0.95	na

Having shown how to fit trust into the process of determining the value of allocations, in the next subsection we provide a simple example to show why the VCG solution presented in section 6.4 is not incentive compatible (and thus not efficient) even when we modify it to consider expected valuations. This then motivates the search for a mechanism that is.

6.5.2 Failure of the VCG Solution

Consider a system of four agents where agent 4 has asked for a task τ to be allocated and its valuation of this task is $v_4(\tau, \theta_4) = 210$. Each agent i has a cost c_i to perform the task proposed by 4 (agent 4 has infinite cost to perform the task by itself) and does not derive any value from the task being performed. Now, suppose that the trust function of agent 4 is a weighed sum of the POS reports by the agents (i.e. $t_i^4 = \alpha \centerdot \widehat{\boldsymbol{\eta}}_{\boldsymbol{i}}$ where $\alpha = [0.3\ \ 0.2\ \ 0.1\ \ 0.4]$). Note that we do not concern ourselves with the reports η_4^i since the task is proposed by agent 4 itself. Table 6.1 shows the cost c_i of attempting the task, and the observed POS value of each agent, $\boldsymbol{\eta^i}$, as well as the trust computed by agent 4, t_i^4, if each agent reports truthfully on its $\boldsymbol{\eta^i}$.

The VCG solution of section 6.4 determines the allocation and payments based only on cost and valuations. However, this would clearly fail to find an efficient allocation since agent 1 would be allocated the task despite being the least trusted and hence most likely to fail. If we instead implemented the VCG mechanism with the *expected* valuations (taking into account the trust and POS reports), we then have $K^* = [0010]$ (i.e agent 3 is allocated the task), $r_1 = r_2 = 0$ and $r_3 = 210\gamma - 130$. Thus, agent 3 will then derive an average payment of $0.87 \times 210 - 130 = 52.7$. However, this scheme is not incentive-compatible because agent 2 can lie about η_3^2 by reporting $\widehat{\eta}_3^2 \leq 0.7357$ which will then lead to it being allocated the task and deriving a positive utility from this allocation. Note that this scheme is exactly that of Porter et al. for a single-task scenario (see section 6.7).

As can be seen, the VCG mechanism needs to be extended to circumvent this problem. Specifically, we require a mechanism that is efficient given the reports of the agents on their costs and valuations of allocations, as well as their observed POS vector (since the VCG is affected by false reports of POS). In effect, we need to change the payment scheme so as to make the truthful-reporting of POSs an optimal strategy for the agent again. Once this is achieved, the centre can then choose the efficient allocation based on expected utilities. The difficulty with designing such a mechanism is that the centre cannot check on the validity of POS reports of agents because it is based on a private observation carried out by the agent. Thus two agents may legitimately differ in their observed POS of another agent due to their different interaction histories with that agent.

6.6 The Trust-Based Mechanism

Before presenting our trust-based mechanism (TBM), we first introduce some notation. Let the expected utility before the allocation is carried out be $\overline{U}(K, \boldsymbol{\theta}, \gamma) = E_{[\gamma|K,\boldsymbol{t^i}]}\left[U(K, \boldsymbol{\theta}, \gamma)\right]$ where $\boldsymbol{\theta}$ is the vector containing all agent types. We also denote the marginal contribution of the agent i to the system given an efficient allocation $\widehat{K}^*$ as $mc_i = \overline{U}_{-i}(\widehat{K}^*, \boldsymbol{\theta}, \gamma) - \max_{K \in \mathcal{K}} \left[\overline{U}_{-i}(K, \boldsymbol{\theta_{-i}}, \gamma)\right]$ where $\max_{K \in \mathcal{K}} \left[\overline{U}_{-i}(K, \boldsymbol{\theta_{-i}}, \gamma)\right]$ is the overall expected utility of the efficient allocation that would have resulted if agent i were not present in the system. Now, we can detail TBM:

1. Find the efficient allocation $\widehat{K}^*$ such that:

$$\widehat{K}^* = \arg\max_{K \in \mathcal{K}} \overline{U}(K, \boldsymbol{\theta}, \gamma) \tag{6.8}$$

   This finds the best allocation; that is, the one that maximises the sum of *expected utilities* of the agents, conditional on the reports of the agents. We note here that we do not take into consideration the reward functions of the agents when calculating the overall utility since these rewards are from one agent to another and therefore do not make a difference when calculating the overall utility of the agents.

2. We now calculate the efficient allocation that would have resulted if an agent i's report is taken out:

$$K^*_{-i} = \arg\max_{K \in \mathcal{K}} E_{[\gamma|K,\boldsymbol{\tilde{t}^i}]}\left[U(K, \boldsymbol{\theta}, \gamma)\right]$$

   where $\boldsymbol{\tilde{t}^i} = g(\boldsymbol{\widehat{\eta}} \setminus \boldsymbol{\widehat{\eta}^i})$. This computes how $\boldsymbol{\widehat{\eta}^i}$ affects which allocation is deemed efficient.

3. We now find the effect that an agent's $\boldsymbol{\widehat{\eta}^i}$ has had on its marginal contribution. Thus, find:

$$D_i = \overline{U}(\widehat{K}^*, .) - \overline{U}(K^*_{-i}, .)$$

   This distils the effect of an agent's $\boldsymbol{\widehat{\eta}^i}$ reports.

4. Given K^*, the payment r_i made to the agent i is then [7]:

$$r_i = mc_i - D_i \tag{6.9}$$

   Naturally, if r_i is negative it implies that i makes a payment to the centre. The first part of the payment scheme, mc_i, calculates the effect that an agent's *presence* has had on the

[7] The calculation is the same for a task provider j and is thus omitted.

overall expected utility of the system. We also subtract D_i to take into account the effect that an agent's POS report has on the chosen allocation. This is in line with the intuition behind VCG mechanisms in which an agent's report affects the allocation, but not the payment it receives or gives.

We will now prove each of the properties of TBM in turn, whilst intuitively explaining why the mechanism has the aforementioned properties.

6.6.1 Properties of the Mechanism

Proposition 6.3. *TBM is incentive-compatible in ex-ante Nash Equilibrium.*

Proof. We first need to calculate the expected utility, $E_{[\gamma|K,\boldsymbol{t^i}]}\left[u_i(K,\theta_i,\boldsymbol{\gamma})\right]$, that an agent derives from TBM because the goal of a rational agent is to maximise its expected utility. We note here that we are assuming that the agent is myopic in that it is only concerned with its current expected utility given the cost vector, $\boldsymbol{c}(K,\boldsymbol{\theta})$, the value vector, $\boldsymbol{v}(K,\boldsymbol{\theta})$, and the trust vector $\boldsymbol{t}$. The expected utility that an agent (since the proofs are identical for the task providers and requesters, we shall refer to an agent $i \in \mathcal{I} \cup \mathcal{J}$), $\overline{u}_i(\widehat{K}^*,\theta_i,\boldsymbol{\gamma})$, derives from an efficient allocation, as calculated from equation 6.8, given the reports of all agents in the system is:

$$\begin{aligned}\overline{u}_i(\widehat{K}^*,\theta_i,\boldsymbol{\gamma}) = {} & E_{[\boldsymbol{\gamma}|\widehat{K}^*,\boldsymbol{t^i}]}\big[v_i(\widehat{K}^*,\theta_i,\boldsymbol{\gamma})\big] - c_i(\widehat{K}^*,\theta_i) \\ & + mc_i(\widehat{K}^*,\theta_i,\boldsymbol{\gamma}) - D_i \\ = {} & E_{[\boldsymbol{\gamma}|\widehat{K}^*,\boldsymbol{t^i}]}\big[v_i(\widehat{K}^*,\theta_i,\boldsymbol{\gamma}) - \widehat{v}_i(\widehat{K}^*,\theta_i,\boldsymbol{\gamma})\big] \\ & - \Big(c_i(\widehat{K}^*,\theta_i) - \widehat{c}_i(\widehat{K}^*,\theta_i)\Big) + \\ & \overline{U}(K^*_{-i},\boldsymbol{\theta},\boldsymbol{\gamma}) - \max_{K\in\mathcal{K}}\big[\overline{U}_{-i}(K,\boldsymbol{\theta_{-i}},\boldsymbol{\gamma})\big]\end{aligned} \tag{6.10}$$

From 6.10 we will firstly prove the following lemma:

Lemma 6.4. *An agent has an equilibrium strategy to reveal its observed POS values.*

Proof. We consider how $\widehat{\boldsymbol{\eta}}^{\boldsymbol{i}}$ affects $\overline{u}_i(\widehat{K}^*,\theta_i,\boldsymbol{\gamma})$. From equation 6.10 we observe that $\widehat{\boldsymbol{\eta}}^{\boldsymbol{i}}$ cannot affect $\overline{U}(K_{-i},\boldsymbol{\theta},\boldsymbol{\gamma}) - \max_{K\in\mathcal{K}}\big[\overline{U}_{-i}(K,\boldsymbol{\theta_{-i}},\boldsymbol{\gamma})\big]$. Thus, an agent only has an incentive to lie so that $\widehat{K}^*$ is selected such that $E_{[\boldsymbol{\gamma}|\widehat{K}^*,\boldsymbol{t^i}]}\big[v_i(\widehat{K}^*,\theta_i,\boldsymbol{\gamma}) - \widehat{v}_i(\widehat{K}^*,\theta_i,\boldsymbol{\gamma})\big] - \Big(c_i(\widehat{K}^*,\theta_i) - \widehat{c}_i(\widehat{K}^*,\theta_i)\Big)$ is maximised. If an agent reveals its cost and valuation truthfully, i.e. $\widehat{v}(.) = v(.)$ and $\widehat{c}(.) = c$, we then have the term as zero. Then an agent cannot gain from an untruthful reporting of $\widehat{\boldsymbol{\eta}}_{\boldsymbol{i}}$. If, however, an agent is to gain from such an untruthful reporting, it needs to set either $\widehat{v}(.) < v(.)$ and $\widehat{c}(.) > c$ or both. However, doing so would decrease the chance of i successfully allocating a task or winning an allocation. Therefore, i would not reveal untruthful values for $\widehat{c}(.)$ and $\widehat{v}(.)$. Moreover, i will actually report truthfully its $\widehat{\boldsymbol{\eta}}^{\boldsymbol{i}}$ since this allows the centre to choose those agents that i deems to have a high POS (as well as helping other agents choose i as having a perception close to theirs). Thus, reporting $\widehat{\boldsymbol{\eta}}^{\boldsymbol{i}} = \boldsymbol{\eta}^{\boldsymbol{i}}$ is an ex-ante Nash equilibrium strategy. $\square$

Given lemma 6.4, we can now show that TBM is incentive compatible. Suppose that an agent is truthful about $\widehat{v}(.)$ and $\widehat{c}(.)$. Then its utility is $\overline{U}(K^*_{-i}, \boldsymbol{\theta}, \boldsymbol{\gamma}) - \max_{K \in \mathcal{K}} \left[\overline{U}_{-i}(K, \boldsymbol{\theta_{-i}}, \boldsymbol{\gamma})\right]$. Now assume that the agent lies about $\widehat{v}(.)$ and $\widehat{c}(.)$ so as to increase its utility. This then means that $E_{[\boldsymbol{\gamma}|\widehat{K}^*, \boldsymbol{t_i}]}\left[v_i(\widehat{K}^*, \theta_i, \boldsymbol{\gamma}) - \widehat{v}_i(\widehat{K}^*, \theta_i, \boldsymbol{\gamma})\right] - \Big(c_i(\widehat{K}^*, \theta_i) - \widehat{c}_i(\widehat{K}^*, \theta_i)\Big) + \overline{U}(K'_{-i}, \boldsymbol{\theta}, \boldsymbol{\gamma}) > \overline{U}(K^*_{-i}, \boldsymbol{\theta}, \boldsymbol{\gamma})$ where K'_{-i} is the efficient allocation found with $\widehat{c}(.)$ and $\widehat{v}(.)$ without the report of $\boldsymbol{\eta}^{\boldsymbol{i}}$. However, as argued earlier, an agent would not report a lower value or a higher cost. Thus $E_{[\boldsymbol{\gamma}|K, \boldsymbol{t}^{\boldsymbol{i}}]}\left[v_i(\widehat{K}^*, \theta_i, \boldsymbol{\gamma}) - \widehat{v}_i(\widehat{K}^*, \theta_i, \boldsymbol{\gamma})\right] - \Big(c_i(\widehat{K}^*, \theta_i) - \widehat{c}_i(\widehat{K}^*, \theta_i)\Big) \leq 0$. Furthermore, by the maximisation of step 2 of TBM, $\overline{U}(K'_{-i}, \boldsymbol{\theta}, \boldsymbol{\gamma}) < \overline{U}(K^*_{-i}, \boldsymbol{\theta}, \boldsymbol{\gamma})$ if all other agents report truthfully. Thus, TBM is incentive-compatible in a Nash equilibrium. $\square$

Proposition 6.5. *TBM is efficient.*

Proof. Given that the agents are incentivised to report truthfully (proposition 6.3), the centre will calculate the efficient allocation according to equation 6.8 (i.e. $\widehat{K}^* = K^*$). $\square$

Proposition 6.6. *TBM is individually-rational in expected utility (as defined in section 2).*

Proof. We need to show that the expected utility of any agent from an efficient allocation K^* is greater than if the agent were not in the scheme (i.e. $\overline{u}_i(K^*, \theta_i, \boldsymbol{\gamma}) \geq 0$). As a result of the inherent uncertainty in the completion of tasks, we cannot guarantee that the mechanism will be ex-post individually-rational for an agent. Rather, we prove that the mechanism is individually-rational for an agent if we consider expected utility. Given truthful reports, the utility of an agent from equation 6.10 is $\overline{U}(K^*_{-i}, \boldsymbol{\theta}, \boldsymbol{\gamma}) - \max_{K \in \mathcal{K}} \left[\overline{U}_{-i}(K, \boldsymbol{\theta_{-i}}, \boldsymbol{\gamma})\right]$. The first maximisation is carried out without the reports $\boldsymbol{\eta}^{\boldsymbol{i}}_{\boldsymbol{-i}}$,whereas the second one is carried out over the set of agents $\mathcal{I} \setminus i$. Thus, the second maximisation is carried out over a smaller set than the first one. As a result, $\max_{K \in \mathcal{K}} \left[\overline{U}_{-i}(K, \boldsymbol{\theta_{-i}}, \boldsymbol{\gamma})\right] \geq \overline{U}(K^*_{-i}, \boldsymbol{\theta}, \boldsymbol{\gamma})$ such that $\overline{u}_i(K^*, \theta_i, \boldsymbol{\gamma}) \geq 0$. $\square$

6.7 Instances of Trust-Based Mechanism

Having thus discussed the computational properties of TBM, we now demonstrate its generality. Specifically, TBM can be viewed as a generalised version of both the VCG mechanism and the mechanism by Porter et al. This is because in TBM, there exists uncertainties about whether a set of agents will carry out an allocation and about the relevance of reports of POS by agents. In this section, we demonstrate its generality by analysing two specific instances of the mechanism. We first show that TBM reduces to Porter et al.'s fault-tolerant mechanism (FTM) and then to the VCG mechanism described in section 6.4.

6.7.1 Self-POS Reports Only

The mechanism developed in [Porter et al., 2002] is a special case of TBM. Specifically, agents only report on their own POS (i.e. $\widehat{\boldsymbol{\eta}}^{\boldsymbol{i}} = \widehat{\eta}_i^i$) and agents assign a relevance of 1 to reports by all other agents. However, since in their model there is no notion of varying perceptions of success, we need to introduce the notion of a *report agent* that has $v(K,.) = 0$ and $c(K,.) = \infty$. This acts as a proxy to agents reporting the ex-post POS to the centre. This also caters for the problem of single POS reports ($\overline{U}(K^*_{-i},.)$ is undefined) as there is then no measure of t^i_j once j's report is removed. The centre then calculates the efficient allocation as: $K^* = \arg\max_{K \in \mathcal{K}} \left[\overline{U}(\widehat{K}^*, \boldsymbol{\theta}, \gamma)\right]$ and the payment to the agent i is $r_i = mc_i - D_i = mc_i$. The term $D_i = 0$ since, as a result of the report agent, $\overline{U}(\widehat{K}^*,.) = \overline{U}(K^*_{-i},.)$ (because $\boldsymbol{t}$ is equal in both cases).

6.7.2 Efficiency Independent Scenario

In this case we do not consider the reports of efficiency. Thereby, trust in the allocation and payment schemes are equivalent to setting the trust to be constant at 1 for every agent. Thus, from equation 6.8, the efficient allocation is:

$$\begin{aligned} K^* &= \arg\max_{K \in \mathcal{K}} \left[E_{[\gamma|K, \boldsymbol{t}^i = \mathbf{1}]} \left[\sum_{i \in \mathcal{I}} \widehat{v}_i(K, \theta_i) \right] - \sum_{i \in \mathcal{I}} \widehat{c}_i(K, \theta_i) \right] \\ &= \arg\max_{K \in \mathcal{K}} \left[\sum_{i \in \mathcal{I}} \widehat{v}_i(K, \theta_i) - \sum_{i \in \mathcal{I}} \widehat{c}_i(K, \theta_i) \right] \end{aligned}$$

The payment scheme is:

$$\begin{aligned} r_i &= U_{-i}(\widehat{K}^*, \gamma, c_{-i}) - E_{[\gamma|\widehat{\widehat{K}}^*, \boldsymbol{t_{-i}}]}(U(\widehat{\widehat{K}}^*, \boldsymbol{\theta}_{-i})) - D_i \\ &= U_{-i}(\widehat{K}^*, \gamma, \boldsymbol{\theta}_{-i}) - (U(\widehat{\widehat{K}}^*, \boldsymbol{\theta}_{-i})) \end{aligned}$$

since $D_i(.)$ becomes irrelevant and $E_{[\gamma|\widehat{\widehat{K}}^*, \boldsymbol{t_{-i}}]}\left[U(\widehat{\widehat{K}}^*, \boldsymbol{\theta}_{-i})\right] = U(\widehat{\widehat{K}}^*, \boldsymbol{\theta}_{-i})$. We thus have both the allocation and payment scheme the same as the VCG mechanism presented in section 6.4.

6.8 Implementing the Trust-Based Mechanism

Having explained the economic properties of the TBM, we now consider its computational properties. In more detail, we present the implementation of TBM as an optimisation process that combines integer programming (IP) and dynamic programming (DP) thereby greatly reducing the computational load. We first describe an optimisation model based on IP and discuss the various constraints that must be applied to take into account expected valuations for all possible allocations. Given this, we then show how the set of allocations to be considered can be reduced using an algorithm based on DP. Therefore, our approach combines a preprocessing stage (along the lines of [Sandholm, 2002a]) with an optimisation stage (along the lines of [Andersson et al., 2000]) to produce an implementation that captures the model presented in section 6.6.

6.8.1 The Optimisation Model

In order to conserve the economic properties of the TBM discussed in section 6.6, it is imperative that the algorithms used to determine the allocation produce the allocation that maximises the expected utility (i.e. the algorithm needs to solve the optimisation problem presented in equation 6.8 exactly). This restricts our scope since we cannot use approximate algorithms such as those developed by [Parkes and Schneidman, 2004; Lehmann et al., 2002].

The search space of the optimisation problem we seek to solve can be represented graphically as shown in figure 6.5. Thus, as compared to figure 6.2, the task column has been expanded in order to represent possible allocations of tasks from task requesters to task performers. Before explaining this mapping in more detail, we will first introduce some useful graph theory notations.

Notice that in figure 6.5, each node in the valuation column, $\mathbf{v}$, is potentially related to multiple nodes in the expanded task column $\mathcal{T}$(as shown by the dotted red sets) and each node in $\mathbf{c}$ is potentially linked to multiple nodes in $\mathcal{T}$ (as shown by the dotted black sets). $\mathcal{T}$ contains decomposed tasks from the bids denoting the bidder, one particular task, and the set of tasks in the bid from which this particular task originates (e.g. $t_1[1, \{t_1, t_3\}]$ signifies task 1 from agent 1 in the bid placed for tasks $\{t_1, t_3\}$ (as shown by the arrows on the figure)). The relationships between the nodes of $\mathbf{v}$, $\mathbf{c}$, and $\mathcal{T}$ can be thus be regarded as a special type of edge involving several nodes. Hence, the problem we are trying to solve contains hypergraphs.

Specifically, a hypergraph can be defined in the following manner [Berge, 1973]:

Definition 6.7. Hypergraph. Let $X = \{x_1, x_2, \ldots, x_n\}$ be a finite set, and let $\mathcal{E} = \{E_i | i \in I\}$ be a family of subsets of X. The family $\mathcal{E}$ is said to be a hypergraph on X if:

1. $E_i \neq \emptyset \ \ (i \in I)$
2. $\cup_{i \in I} E_i = X$.

The pair $H = (X, \mathcal{E})$ is called a hypergraph. The elements $x_1, x_2, \ldots, x_n$ are called the vertices and the sets $E_1, E_2, \ldots, E_m$ are called the hyperedges.

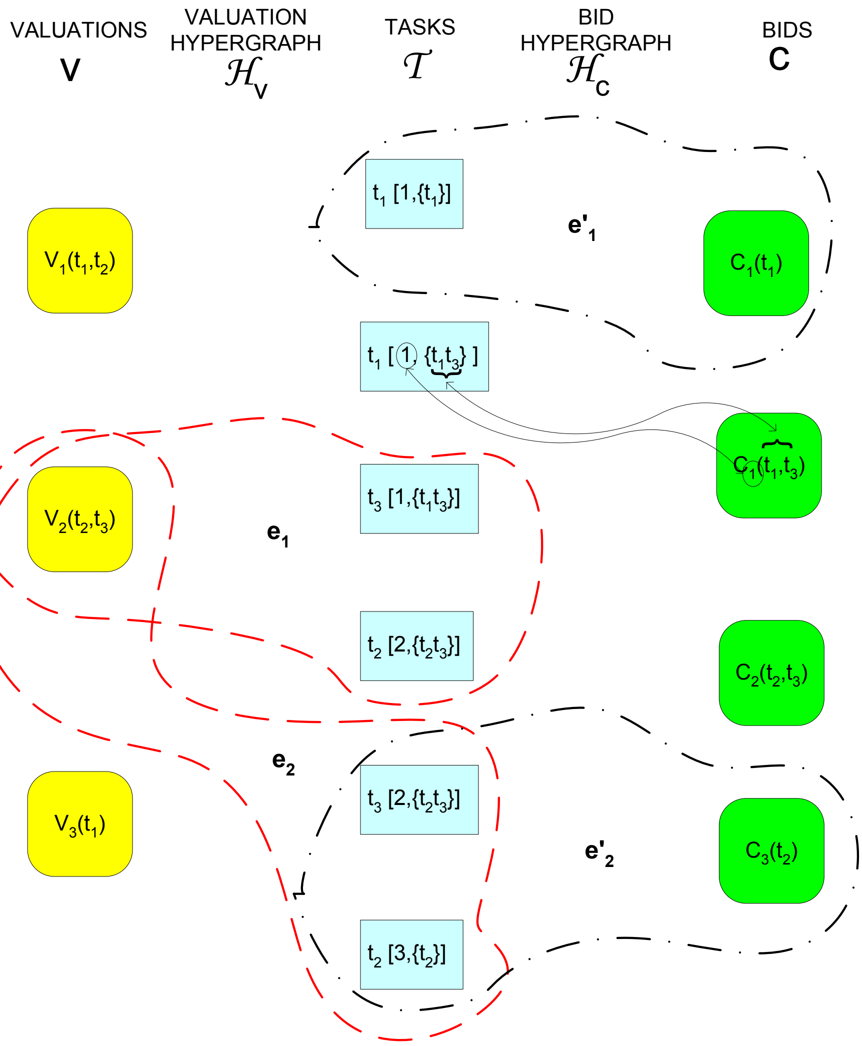


FIGURE 6.5: Graphical representation of the TBM search space.

In a hypergraph, two hyperedges are said to be *adjacent* if their intersection is not empty. Otherwise they are said to be *disjoint*. We say that a hypergraph is *weighted* if we associate to each hyperedge $E \in \mathcal{E}$ a real number, $w(E)$, called the *weight* of E.

From the formal definition of hypergraph, we observe that figure 6.5 results from the overlapping of two separate hypergraphs: (i) the *valuation hypergraph* resulting from linking valuations to task bids and (ii) the *bid hypergraph* resulting from linking each bid to its task bids. In what follows, we formally define both hypergraphs based on valuations and bids.

Let $\boldsymbol{v} = \{v_i(\boldsymbol{\tau}, \theta_i) | v_i(\boldsymbol{\tau}, \theta_i) \in \boldsymbol{v}_i \; i \in I\}$ and $\boldsymbol{c} = \{c_j(\boldsymbol{\tau}, \theta_j) | c_j(\boldsymbol{\tau}, \theta_j) \in \boldsymbol{c}_j \; j \in J\}$ be the sets of all valuations and all bids respectively. Consider now that each bid $c_j(\boldsymbol{\tau}, \theta_j)$ standing for the offer of agent j for a set of tasks $\boldsymbol{\tau}$ can be split into single-task bids for every task in $\boldsymbol{\tau}$ so that $k_j^{\boldsymbol{\tau}}(\tau_{j1}, \theta_j), \ldots, k_j^{\boldsymbol{\tau}}(\tau_{jn}, \theta_j)$ represents the cost of agent j for tasks $\tau_{j1}, \ldots, \tau_{jn}$, and $c_j(\boldsymbol{\tau}, \theta_j) = \sum_{\tau' \in \boldsymbol{\tau}} k_j^{\boldsymbol{\tau}}(\tau', \theta_j)$. Thus, we define $\mathcal{K} = \{k_j^{\boldsymbol{\tau}}(\tau', \theta_j) | c_j(\boldsymbol{\tau}, \theta_j) \in \boldsymbol{c}, \tau' \in \boldsymbol{\tau}\}$ as the set containing the cost per single task for every bid in $\boldsymbol{c}$. Notice though that the splitting of bids into single-task bids is, as shown below, an artefact to help us build our optimisation model. Therefore, single-task bids must be regarded as *dummy single-task bids* since we shall never require sellers to make explicit their values.

Hence, on the one hand, we define the valuation hypergraph as a pair $\mathcal{H}_v = (\boldsymbol{v} \cup \mathcal{K}, \mathcal{E}_v)$. We say that $e \in \mathcal{E}_v$ where $e = \{v_i(\boldsymbol{\tau}, \theta_i)\} \cup K$ and $K \subseteq \mathcal{K}$ iff $\boldsymbol{\tau} = \cup_{k(\tau,\theta)} \tau$. Thus, each hyperedge in $\mathcal{H}_v$ consists of a single valuation vertex corresponding to an element in $\boldsymbol{v}$ along with a complete task allocation out of the dummy single-task bids in $\mathcal{K}$. On the other hand, we define the bid hypergraph as a pair $\mathcal{H}_c = (\mathcal{K} \cup \boldsymbol{c}, \mathcal{E}_c)$. For each bid $c_j(\boldsymbol{\tau}, \theta_j) \in \boldsymbol{c}$ such that it splits into dummy single-task bids $k_j^{\boldsymbol{\tau}}(\tau_{j_1}, \theta_j), \ldots, k_j^{\boldsymbol{\tau}}(\tau_{j_n}, \theta_j)$, there is a hyperedge $e = \{c_j(\boldsymbol{\tau}, \theta_j), k_j^{\boldsymbol{\tau}}(\tau_{j_1}, \theta_j), \ldots, k_j^{\boldsymbol{\tau}}(\tau_{j_n}, \theta_j)\}$. In other words, each hyperedge in $\mathcal{H}_c$ consists of a single bid vertex corresponding to an element in $\boldsymbol{c}$ along with the single-task costs in $\mathcal{K}$ resulting from splitting the bid. Notice that our definitions of valuation and bid hypergraphs ensure that each hyperedge in H_v contains a single valuation from $\boldsymbol{v}$ and each hyperedge in H_c contains a single bid from $\boldsymbol{c}$.

In addition to the definitions above, we shall require some auxiliary functions to operate on the hyperedges of both hypergraphs:

- $\delta(x) = \{e \in \mathcal{H}_v | x \in e\}$ returns all hyperedges in $\mathcal{H}_v$ containing x.
- $\lambda(x) = \{e \in \mathcal{H}_c | x \in e\}$ returns all hyperedges in $\mathcal{H}_c$ containing x.
- $\nu(e)$ returns the valuation vertex in $\boldsymbol{v}$ in hyperedge $e \in \mathcal{H}_v$.
- $\alpha(e)$ returns the bid vertex in $\boldsymbol{c}$ in hyperedge $e \in \mathcal{H}_c$.
- $\beta(e) = \{k \in \mathcal{K} | k \in e\}$ returns the cost vertexes in hyperedge $e \in \mathcal{H}_v$.

Consider now that we turn both $\mathcal{H}_v$ and $\mathcal{H}_c$ into weighted hypergraphs as follows. On the one hand, since each hyperedge $e \in \mathcal{H}_v$ stands for a *local* bid allocation, namely an allocation for valuation vertex $\nu(e)$, we can associate to each hyperedge the expected valuation for $\nu(e)$ given the local bid allocation represented by the cost vertexes in $\beta(e)$ as its weight. We can thus recast our calculation of expected valuations as follows:

$$E[v_i(\boldsymbol{\tau}, \theta_i), K] = \sum_{\boldsymbol{\tau_n} \in 2^{\boldsymbol{\tau}}} \left(v_i(\boldsymbol{\tau_n}, \theta_i) \prod_{\tau_l \in \boldsymbol{\tau_n}, (\boldsymbol{\tau}_l \to j) \in K} t^i_{j,\tau_l} \prod_{\tau_m \in \boldsymbol{\tau}/\boldsymbol{\tau_n}, (\boldsymbol{\tau}_m \to k) \in K} 1 - t^i_{k,\tau_m} \right) \tag{6.11}$$

where i assigns the tasks τ_l or τ_m to a given agent j and k respectively according to the local bid allocation K chosen.

On the other hand, we can associate to each hyperedge $e' \in \mathcal{H}_c$ the value of the bid vertex $\alpha(e') \in \mathcal{C}$. Henceforth, $\omega(e)$ shall stand for the weight of hyperedge $e \in \mathcal{H}_v$, whereas $\kappa(e')$ shall stand for the weight of hyperedge $e' \in \mathcal{H}_c$.

Once both $\mathcal{H}_v$ and $\mathcal{H}_c$ are completely constructed, we can then exploit these structures to obtain an allocation for the TBM model. In order to do so, notice that if two hyperedges in $\mathcal{H}_v$ are adjacent it means that two valuations sharing some tasks would be allocated the very same bid for that task, which turns out to be an unfeasible allocation. Therefore, feasible allocations can be expressed as sets of disjoint hyperedges, which leads to the well-known matching problem in a hypergraph [Gross and Yellen, 1999]:

Definition 6.8. Matching problem. For a hypergraph $H = (X, \mathcal{E})$, a family $\mathcal{E}' \subseteq \mathcal{E}$ is defined to be a *matching* if the hyperedges of $\mathcal{E}'$ are pairwise disjoint.

With respect to a given matching $\mathcal{E}'$, a vertex x_i is said to be *matched* or *covered* if there is a hyperedge in $\mathcal{E}'$ incident to x_i. If a vertex is not matched, it is said to be *unmatched* or *exposed*. A matching that leaves no vertexes exposed is said to be *complete*.

Therefore, our aim is to find a matching for $\mathcal{H}_v$ that is not necessarily complete (the optimal allocation may demand that some valuations remain exposed). However, we are not interested in any matching, but specifically in the one that maximises the sum of the total expected valuations weighting the hyperedges in $\mathcal{H}_v$. This leads us to the well-known *maximum weighted matching* problem [Gondrand and Minoux, 1986] which consists of finding a matching for which the sum of the weights of the hyperedges is maximised.

Nonetheless, we cannot solve the maximum weighted matching problem for $\mathcal{H}_v$ without taking into account $\mathcal{H}_c$. We also require a matching for $\mathcal{H}_c$, but, in this case, a minimum weighted matching so that the total cost of selected bids is minimised. In turn, the matching for $\mathcal{H}_c$ also depends on the matching for $\mathcal{H}_v$: whenever hyperedge $e \in \mathcal{H}_v$ is selected (a valuation is selected along with a set of task costs) we must enforce the fact that the hyperedges in $\mathcal{H}_c$ containing the cost vertexes in e are also selected (and thus, a selected bid is considered along with its task costs). In this way, any matching in $\mathcal{H}_v$ generates an associated matching in $\mathcal{H}_c$. Our aim is to obtain the maximum weighted matching in $\mathcal{H}_v$ that minimises its associated weighted matching in $\mathcal{H}_c$.

Finally, the surplus maximising task allocation in a trust-based scenario results from the solution to the maximisation of the following expression:

$$\sum_{e \in \mathcal{E}v} x_e \cdot \omega(e) - \sum_{e' \in \mathcal{E}_c} y_{e'} \cdot \kappa(e') \tag{6.12}$$

subject to:

1. $\sum_{e \in \delta(k)} x_e = y_{\lambda(k)} \; \forall k \in \mathcal{K}$
2. $y_e + y_{e'} \leq 1 \; \forall e, e'$ iff $c(e) = c_i(\boldsymbol{\tau}_j, \theta_i), c(e') = c_i(\boldsymbol{\tau}_k, \theta_i)$ and $\boldsymbol{\tau}_j \cap \boldsymbol{\tau}_k \neq \emptyset$
3. $x_e + x_{e'} \leq 1 \; \forall e, e'$ iff $v(e) = v_i(\boldsymbol{\tau}_j, \theta_i), v(e') = v_i(\boldsymbol{\tau}_k, \theta_i)$ and $\boldsymbol{\tau}_j \cap \boldsymbol{\tau}_k \neq \emptyset$

where $x_e \in \{0,1\}$ is a binary decision variable representing whether the valuation in hyperedge e is selected or not, and $y_{e'} \in \{0,1\}$ is a binary decision variable representing whether the bid in hyperedge e' is selected or not.

As to the side constraints restricting expression 6.12, constraint (1) ensures at the same time that the very same bid cannot be allocated to the very same task of separate valuations and that a valuation cannot have more than one bid allocation. Constraint (2) enforces the fact that overlapping bids owned by the very same agent are exclusive (XOR bids [Andersson et al., 2000]), and hence they cannot be simultaneously selected. Finally, constraint (3) enforces exclusivity among valuations with overlapping tasks (XOR asks). Notice that our optimisation model, as formalised by equation 6.12, resembles the combinatorial exchange (which is a double auction in which buyers and sellers submit sealed bids) since it consists of both bids and asks. Indeed, we can consider the goods in the exchange to be the dummy tasks in $\mathcal{K}$, the bids the elements in $\boldsymbol{c}$, and the asks the weights of the hyperedges in $\mathcal{H}_v$. Thus, while the number of bids remain the same in the exchange, the number of valuations may significantly increase (since we are considering $\mathcal{H}_v$ instead of $\boldsymbol{v}$). This increased complexity can be attributed to the introduction of trust in our theoretical model which makes the initial valuations (asks) (the elements in $\boldsymbol{v}$), allocation-dependent. Hence, from every single valuation in $\boldsymbol{v}$, several potential asks originate for the exchange when considering the bidder to which each task my be allocated. As shown by Sandholm et al. [2001], the decision problem for a binary single-unit combinatorial exchange winner determination problem is $\mathcal{NP}-complete$ and the optimisation problem cannot be approximated to a ratio $n^{1-\epsilon}$ in polynomial time unless $\mathcal{P} = \mathcal{ZPP}$. Our problem is thus $\mathcal{NP}-complete$.

6.8.2 Preprocessing Bids and Allocations

The previous section has considered the allocation problem; that is, to determine the allocation that maximises the expected utility. However, in order to construct the objective function in equation 6.12 we must first generate for each valuation $v_i(\boldsymbol{\tau})$ its expected valuations considering the task allocations in $\mathcal{K}_{v_i(\boldsymbol{\tau})}$. In this section we offer a dynamic programming approach to this problem since, as detailed below, the problem observes the *principle of optimality* (in the sense proposed in [Skiena, 1998]). Thus, partial solutions can be optimally extended with regard to the *state* after the partial solution, instead of to the partial solution itself. In our particular case, the local task allocations for $v_i(\boldsymbol{\tau})$ can be obtained from the allocations assessed for sets of tasks $\boldsymbol{\tau}' \subseteq \boldsymbol{\tau}$.

To this end, algorithms 1 and 2 formalise our DP approach. Specifically, algorithm 1 calculates the expected valuations for each ask $v_i(\boldsymbol{\tau})$ based on the potential task allocations for the tasks in $\boldsymbol{\tau}$ using the costs in $\boldsymbol{c}$. Task allocations are stored in table A, which is employed by the algorithm as a look-up table indexed by task vectors, whereas expected valuations are stored in E. Notice that the first step in the algorithm (line 1) refers to the following preprocessing actions:

PRE1: *Remove non-competitive bids.* Notice that we regard a bid over a set of tasks $\boldsymbol{\tau}$ as non-competitive if all the valuations for tasks in $\boldsymbol{\tau}$ are lower than the bid. Formally, we remove $c_j(\tau)$ if $\max(v(\boldsymbol{\tau})) < c_j(\boldsymbol{\tau}'), \boldsymbol{\tau}' \subseteq \boldsymbol{\tau}$.

PRE2: *Remove bids that cause free disposal.* At this stage, we prune those bids containing tasks for which no valuations exist. Formally, we remove $c_j(\tau)$ if $\exists \tau \in \boldsymbol{\tau}$ such that $\nexists v_i(\boldsymbol{\tau}')$ and $\tau \in \boldsymbol{\tau}'$.

Having carried out these two preprocessing actions, table A is then filled in by the recursive function *allocation* outlined by algorithm 2. This algorithm differentiates two cases when assessing task allocations, depending on whether the task $\boldsymbol{\tau}$ received as an input is *single* (lines 2 to 4) or *combinatorial* (lines 5 to 8). For the single task case, the algorithm locates all bids in $\boldsymbol{c}$ that contain the task (line 3). For combinatorial tasks, the algorithm generates two recursive calls: one for all the elements in the task vector but the last one (line 6), and another for the last element in the task vector (line 7). At this point the algorithm looks for stored results in A to avoid revisiting the same subproblem. If such stored results do exist, they are retrieved, otherwise the recursive calls proceed. Finally, the task allocations obtained in lines 6 and 7 are combined to provide all possible task allocations (line 8). Notice, therefore, that algorithm 2 is in fact a *memoized* [Cormen et al., 1990] recursive algorithm: it maintains a table, A, with subproblem solutions, but the control structure for filling in the table is recursive. A memoized algorithm is desirable in this context because it only solves those subproblems that are definitely required. For instance, consider a call to *allocation* with combinatorial task $\boldsymbol{\tau} = \langle \tau_1, \tau_2, \tau_3 \rangle$ as input. Such a call is split into two recursive calls with inputs $\langle \tau_1, \tau_2 \rangle$ and $\langle \tau_3 \rangle$.

In order to combine task allocations, in algorithm 2 we use the $\otimes$ operator over sets of task allocations (line 8) that we define as follows: $\Delta \otimes \Delta' = \{(\delta_i, \delta_j) | \delta_i \in \Delta, \delta_j \in \Delta'$ and $\nexists (\tau, c_k(\boldsymbol{\tau})) \in \delta_i \; (\tau', c_l(\boldsymbol{\tau}')) \in \delta_j$ such that $c_k(\boldsymbol{\tau})$ and $c_l(\boldsymbol{\tau}')$ are mutually exclusive. Notice that the $\otimes$ operator is defined as a variation of the Cartesian product that discards task allocations containing XOR bids. Thus, the $\otimes$ operator implements the following pruning actions:

PRE3: *Discard task allocations containing mutually exclusive bids.*
After assessing a given task allocation, algorithm 2 returns the result (line 12) to algorithm 1 so that it is stored in the look-up table (line 5). After that, algorithm 1 carries out the following further preprocessing actions.

PRE4: *Remove task allocations that cause free disposal.* Eventually some sets of tasks (along with their subtasks) may only be asked for by a single agent. In such a case, there is no sense in considering local task allocations with overlapping bids (bids over some common task(s)) because their acceptance would only be possible if we allowed free disposal (line 7).

PRE5: *Remove non-competitive task allocations.* We regard a local task allocation as non-competitive if the total cost of the bids composing the allocation is higher than the expected valuation for the tasks being considered. For each allocation, the expected value is computed using equation 6.11 and stored (line 9) if this value is greater or equal to the cost.

Algorithm 1 *function* task_allocations (V,E,C,$\boldsymbol{t}$)

$C' \leftarrow$ **PRE2**(**PRE1**(V, C));
for $i \in \{1, \ldots, I\}$ **do**
  for $v_i(\boldsymbol{\tau}, \theta) \in \boldsymbol{v}_i$ **do**
    if $A[\boldsymbol{\tau}] = \emptyset$ **then**
      $A[\boldsymbol{\tau}] \leftarrow$ allocation$(\boldsymbol{\tau}, A, C')$;
    end if
    $\mathcal{K}_{v_i(\boldsymbol{\tau})} \leftarrow$ **PRE4**$(A[\boldsymbol{\tau}])$;
    for $\mathcal{A} \in \mathcal{K}_{v_i(\boldsymbol{\tau})}$ **do**
      $E[v_i(\boldsymbol{\tau}), \mathcal{K}] \leftarrow$ **PRE5**$(v_i(\boldsymbol{\tau}), \mathcal{K}, \boldsymbol{t})$;
    end for
  end for
end for
return E

Algorithm 2 *function* allocation ($\boldsymbol{\tau}$,A,C)

if $A[\boldsymbol{\tau}] = \emptyset$ **then**
  case $\boldsymbol{\tau} = \langle \tau_i \rangle$:
    $\alpha \leftarrow \{(\tau_i, c_k(\boldsymbol{\tau}')) | \tau_i \in \boldsymbol{\tau}', 1 \leq k \leq I, \boldsymbol{\tau}' \in 2^{\mathcal{T}}\}$;
  break;
  case $\boldsymbol{\tau} = \langle \tau_{i_1}, \ldots, \tau_{i_{m+1}} \rangle$:
    $\Delta^m \leftarrow allocation(\langle \tau_{i_1}, \ldots, \tau_{i_m} \rangle, A, C)$;
    $\Delta^{m+1} \leftarrow allocation(\langle \tau_{i_{m+1}} \rangle, A, C)$;
    $\alpha \leftarrow \Delta^m \otimes \Delta^{m+1}$;
else
  $\alpha \leftarrow A[\boldsymbol{\tau}]$
end if
return α

Algorithm 1 runs in time $O(n \cdot m^r)$ in the worst case, where m stands for the total number of bids, r stands for the number of tasks, and n stands for the total number of valuations. The worst case occurs when all valuations and bids are combinatorial and over all tasks[8]. Otherwise, the running time of the algorithm is highly dependent on the sparsity of bids and valuations. Thus, the lower the degree of (task) overlapping of bids and valuations, the lower the running time[9]. Besides, the pruning actions included in the algorithm are expected to further reduce the search space, and thus the running time. The results of algorithm 1 are used for building the optimisation model in section 6.8.1, which is solved using ILOG's CPLEX.

Notice that a brute-force approach to our optimisation problem would be extremely expensive. In this case, the number of feasible allocations for a given valuation would amount to $\binom{m}{1}^r$. Thereafter, considering a different valuation combined with the former one, would lead us to consider $\binom{m-1}{1}^r$ feasible allocations to be combined with the feasible allocations obtained so far (and thus $\binom{m}{1}^r \cdot \binom{m-1}{1}^r$ overall). In the general case, when jointly considering n valuations,

[8] This case can be generated in CATS [Leyton-Brown et al., 2000] with the *normal* distributions with the parameters of their testbed set as follows, μ_{goods} to high and σ_{goods} to low.

[9] This can be generated using the dual distributions within the CATS testbed.

the total number of feasible allocations would amount to $\prod_{i=0}^{n}(m-i)^r$. This expression asymptotically converges to $(m!)^r$ if $n \simeq m$ and m^{rn} if $n \ll m$ (which is much larger than the $n \cdot m^r$ of our algorithm). Notice that a brute-force approach to our optimisation problem would be extremely expensive. In this case, the number of feasible allocations for a given valuation would amount to $\binom{m}{1}^r$. Thereafter, considering a different valuation combined with the former one, would lead us to consider $\binom{m-1}{1}^r$ feasible allocations to be combined with the feasible allocations obtained so far (and thus $\binom{m}{1}^r \cdot \binom{m-1}{1}^r$ overall). In the general case, when jointly considering n valuations, the total number of feasible allocations would amount to $\prod_{i=0}^{n}(m-i)^r$. This expression asymptotically converges to $(m!)^r$ if $n \simeq m$ and m^{rn} if $n \ll m$. Thus, very simple problems, for instance with m=10, r=5 and $n = 10$, cannot be solved using a high performance optimiser such as CPLEX as brute-force would explore a search space of cardinality $(10!)^5$. On the other hand algorithm 1 would generate $10 \cdot 10^5$ possible allocations, feeding a branch and cut algorithm. Hoffman and Padberg [1993] report the possibility of solving an instance of a very similar problem in under 25 minutes.

Having thus shown how to reduce the computational load whilst implementing TBM, we next experimentally compare its performance against the mechanism developed in [Porter et al., 2002].

6.9 Experimental Evaluation

Here we empirically evaluate TBM by comparing it with the fault tolerant mechanism (FTM) of Porter et al. (this is chosen because it also deals with the POS of agents as discussed in sections 6.3 and 6.7) and the standard VCG. Here, we refer to task performing agents as contractors in what follows. In our experiments we perform 500 successive allocations, in the scenario described in section 6.5, with six agents each given one task to complete [10]. After each allocation, contractors perform tasks and the level of success is measured and reported to all agents. Each agent can then update its measure of the contractors' POSs, as well as the contractors' trustworthiness as discussed in section 6.5. The valuations and POS of each agent are obtained from a uniform distribution and the costs are the same for all tasks. We iterate the process and average the results (here for 200 iterations). Given the properties of TBM and FTM we postulate the following hypotheses and validate them as shown below:

Proposition 6.9. *TBM always chooses the efficient allocation (K^*) in the long run.*

This hypothesis reflects the fact that we expect agents in TBM to take a number of interactions to model the true POS of their counterparts, using their individual trust models. After this time, however, the mechanism can choose those contractors that are most successful at completing a given task. As can be seen in figure 6.6, the optimal allocation chosen by TBM, K^*TBM,

[10] The general results of this experiment held with a number of setting. We chose this setting at random to display the empirical results.

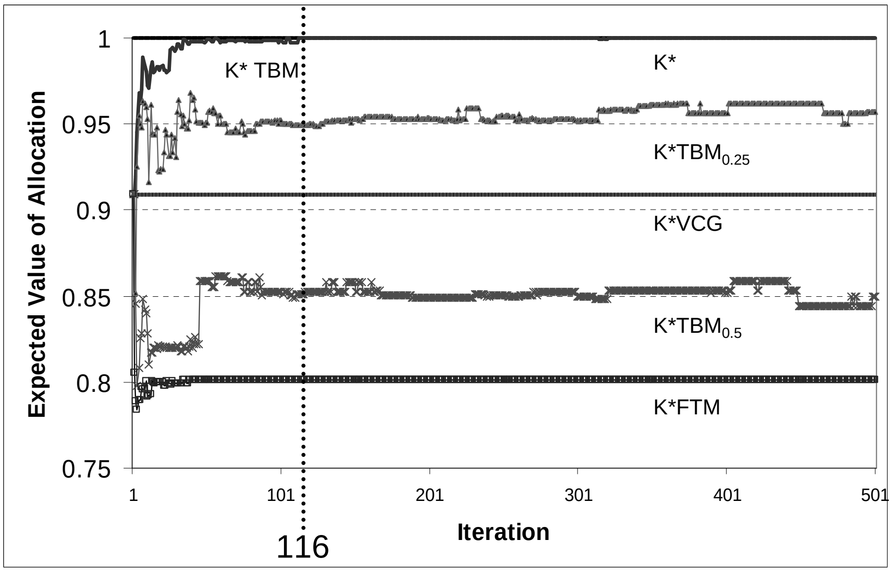


FIGURE 6.6: Expected value of chosen allocations for TBM and FTM

reaches the efficient allocation K^* (given *real* POSs) after 116 interactions[11]. After this, the POS of each contractor is accurately modelled, as is the trust of agents in their contractors. Thus, the most trusted and utility maximising allocation is found by the TBM. This result is observed for all cases where the POSs of contractors are varied.

Proposition 6.10. *TBM finds better allocations than FTM when contractors' own reported POS are biased.*

While FTM only takes into account a contractor's own reports, TBM uses the trust model of the various individual agents (which take into account reports not only from the contractor) to make an allocation. In the particular trust model we use in TBM, an agent can give different weights to reports from different agents (as shown in section 6.5.2). We therefore varied the weight w, assigned to a contractor's report of its own POS in the trust model of an agent. Here we exemplify the cases where $w = 0.5$ (i.e. the contractor's report is given equal weighting to the agent's perceived POS), $w = 0.25$ and $w = 0$ (i.e. no importance is given to the contractor's report).

As can be seen, our hypothesis is validated by the results given in figure 6.6 (with normalised expected values). Note here that K^*VCG is the allocation independent of POSs or if the POSs of agents are all equal. We note as K^*TBM_w the allocation chosen by TBM with a weight w.

In more detail, figure 6.6 depicts the following results: $K^*VCG = 0.909$. At equilibrium, the following ranges are found for the expected value: $K^*TBM = 1, 0.97 > K^*TBM_{0.25} > 0.94$, $0.86 > K^*TBM_{0.5} > 0.84$, and $K^*FTM = 0.8$. Specifically, TBM_0 (i.e. TBM) reaches the optimal allocation K^* (i.e. equivalent to zero bias from the seller) after 116 iterations, while

[11]The results were validated using a Student's t-test with two samples of 100 and 200 iterations assuming equal variances with means $\mu_1 = 0.99999$ and $\mu_2 = 1.0$ and p-value $p = 0.778528$. This means that the difference between the means is not significant.

$TBM_{0.25}$ and $TBM_{0.5}$ settle around a sub-optimal allocation (the expected value of which decreases with increasing w). Moreover, FTM is seen to settle at $K^*FTM = 0.8$ after 82 iterations. In general, it is noted that FTM always settles at $K^*FTM < K^*$ (and sometimes even $K^*FTM < K^*VCG$ as in figure 6.6 depending on the valuations the agents have for the tasks). This result is explained by the fact that the biased reports cause biased trust values to be obtained by the centre which then chooses a sub-optimal allocation (i.e. less than K^* which chooses agents according to their 'real POSs'). $TBM_{0.25}$ and $TBM_{0.5}$ are less affected by biased reports since the weighted trust model reduces the effect of bias on the overall trust values (but still affects the mechanism). In most trust models, however, $w \geq 0.5$ is never given to the contractors' POS report and here it only represents an extreme case [Ramchurn et al., 2004]. Moreover, if the bias is removed, then FTM and the weighted TBMs behave the same as TBM since the agents then perceive the same POS and all achieve K^*. It was also observed that the speed with which TBM and FTM achieve K^* also depends on the difference between the optimum allocation and the other allocations. This is because the smaller the differences, the harder it becomes to differentiate these allocations given imperfect estimations of POSs (i.e. the larger the samples, the more accurate the POSs are, hence the longer the learning rate).

6.10 Summary

In this chapter we have considered the case where uncertainties occur as to whether an agent will complete its allocated task. In order to deal with this problem, we introduced the notion of trust-based mechanism design (TBMD) which generalises the VCG mechanism by using the trust model of individual agents in order to generate efficient allocations. We discussed the properties that a generic trust model should possess in order to ensure efficient allocation. We then developed a trust-based mechanism (TBM) and proved that it is efficient, individually rational, and incentive compatible. We have also considered the computational properties of this mechanism and shown that the allocation problem is $\mathcal{NP} - complete$ by reducing the problem to two linked maximum weighted matching problems. Furthermore, we developed algorithms based on DP so as to speed up the generation of possible allocations. We then demonstrated the generality of the mechanism by reducing it to two known mechanisms, namely the VCG mechanism and the FTM mechanism. We finally empirically evaluated our mechanism against the FTM and showed that it is robust to bias in the system (unlike FTM).

In order to ground the theoretical work described in this chapter, we adopted a MSN scenario where sensors can fail in completing their assigned tasks. Sensors typically fail due to a number of reasons as outlined in section 6.2. These failures hampers the implementation of traditional mechanisms within them, since as we have shown, traditional mechanisms such as the VCG cannot cope with failures. Therefore, the mechanism presented here represent an important advance in that it can deal with such failures within the context of individually-owned sensors.

In sum, this chapter has considered an important class of auctions in which there is an uncertainty in the completion of a task associated with each task performer. In so doing, we have addressed requirement 4 from the list of requirements outlined in chapter 1 (namely that of uncertainty in task completion). Furthermore, we have allowed the task requesters to exchange information about their past experience with task performers. This has addressed requirement 1 since this information is distributed amongst potentially many agents and is not known to the task performer before the allocation is decided. Moreover, we have empirically evaluated TBM and shown that it *always* achieves the optimum allocation in the long run and achieves better allocations than its closest comparison when contractors provide biased reports of POS.

Chapter 7

Conclusions

In this chapter, we will present a global view on what we achieved in terms of analysing and designing distributed mechanisms. Thus, in Section 7.1, we will first summarise the research carried out within each chapter. In doing so we will also explain how we addressed each of the requirements that we initially set out at the beginning of this thesis. In Section 7.2, we use the knowledge gained within this thesis to identify promising areas of future work.

7.1 Summary

Distributed mechanisms are fast becoming imperative for operating networked systems that allow software agents representing distinct stakeholders with different aims and objectives to interact. Such mechanisms are gaining prominence since they are more robust, less prone to bottlenecks, more tractable and more trusted than their centralised counterparts. Now, there are two main points of focus from which the design process within distributed mechanisms can be carried out: 1) the design of optimal strategies for agents given the prevailing protocol and 2) the design of protocols that govern the interactions between the agents. This thesis has reported work from both of these. Using the former perspective, in Chapter 3, we developed an optimal strategy for a bidder in a market consisting of simultaneous Vickrey auctions. Then, using the latter perspective we designed protocols that seek to address the requirements of constrained capacity, interdependent valuations and uncertainty in task allocation (in Chapters 4, 5 and 6 respectively). Within this context, we can segment the work reported in these chapters as addressing two broad issues that are associated with distributed mechanisms; namely that of distributed allocation mechanisms (Chapters 3 and 4) where the allocation is not computed by a centre and distributed information mechanisms (Chapters 5 and 6) where an agent requires information from other agents in the system so as to determine its value (or cost) for an item (or task). In order to ground our work, we have employed a running scenario within this thesis that deals with a multi-sensor network that is composed of individually-owned sensors. Whilst the results presented within this thesis can be generally applied to MASs, the sensor network scenario provides us with a canonical example of how the research problems addressed may arise.

At a general level, this thesis has improved the understanding of distributed mechanisms for multi-agent systems in the following ways:

- **Distributed Allocation:** We studied the simultaneous auctions mechanism in Chapter 3 and designed a modified continuous double auction in Chapter 4. In both of these distributed allocation mechanisms, we find that their efficiency is less than the full efficiency that can be achieved with centralised mechanism. Specifically, we find that there is a lower bound on the efficiency given by $1 - \frac{1}{e}$ within simultaneous auctions, whilst in the continuous double auction, the efficiency ranges between 0.64 and 0.95. From this, we can infer that there is a cost of distributing a mechanism in that we can no longer achieve full efficiency. Nevertheless, this cost may not be overly prohibitive in certain scenarios where the advantages associated with distributed mechanisms are more important.
- **Distributed Information:** We designed distributed information mechanisms in Chapter 5 and 6 since the traditional mechanisms cannot incentivise the agents to choose strategies that lead to desirable outcomes in cases where their valuation are interdependent. In both these lines of research, we were able to achieve efficiency and, in so doing, showed that efficient mechanisms can be designed within distributed information scenarios.

Furthermore, Chapters 4, 5 and 6 have considered the computational aspects of mechanism design. In Chapter 4, a dynamic programming approach was sufficient to ensure that the solution was found in pseudo-polynomial time. We then showed in Chapter 5 that the designed mechanism does not impose additional computational burden on the agents. However, the centre still faced an $\mathcal{NP}$-hard problem since it was carrying out a combinatorial allocation. In Chapter 6, we then considered how to combine a dynamic programming with a linear programming approach in order to speed up the computation. However, the problem remains $\mathcal{NP}$-hard, since as in Chapter 5, an exact solution is required.

Finally, the scenario considered within this thesis is itself novel in that it considers sensor networks in which the sensors are individually owned and that can trade information and services amongst themselves. This scenario gave rise to the four requirements outlined in Chapter 1 which have been addressed by the research reported in each of the chapters.

In more detail, in Chapter 3, we studied utility-maximising strategies for agents participating in multiple, simultaneous second-price auctions. In this context, we find the counter-intuitive result that it is optimal for a global agent to place non-zero bids in all auctions that sell the desired item, even when the bidder only requires a single item and derives no additional benefit from having more. This result holds when the global agent is facing either local agents only or a mixture of global and local agents. For this distributed allocation mechanism, we study the efficiency of the market with and without a global bidder. We first derive a lower efficiency bound for such markets in the absence of global bidders. We then empirically study the efficiency of the market as the number of bidders vary. We show that, if the global bidder can accurately predict the number of local bidders in each auction, the efficiency slightly increases. In contrast, if there is much uncertainty, the efficiency significantly decreases as the number of auctions increases due to the increased probability that a global bidder wins more than two items. These results demonstrate that the way in which the efficiency and, thus the social welfare is affected by a global bidder depends on the information available to that global bidder.

In Chapter 4, we considered the design of both a centralised and a decentralised protocol in a scenario where the production costs are characterised by a cost function composed of a fixed cost, a constant marginal cost and a limited capacity. The centralised mechanism extends the standard VCG mechanism to this problem domain by introducing a novel penalty scheme. This resulted in the mechanism being strategyproof, individual rational, efficient and robust to uncertainties in the capacities of the agents. A dynamic programming algorithm, that solves the task allocation problem of the centre in pseudo-polynomial time, then shows how the mechanism is also computationally efficient. However, the mechanism is centralised. Therefore, in the second mechanism, we extend the standard format of a continuous double auction so as to develop a decentralised mechanism for resource allocation in the same context. We find that this mechanism has a high inherent average efficiency (over 86% in the examples we study) by testing it with a variant of the zero intelligence strategy. Thus, we find that these mechanisms represent a trade-off in terms of efficiency and the decentralisation of a mechanism. However, both mechanisms still ensure that the participants derive a profit by joining the mechanism, thereby justifying their use with selfish agents.

Having dealt with distributed allocation in Chapters 3 and 4, the second part of this thesis (Chapters 5 and 6) considers distributed information mechanisms. In Chapter 5, we first developed a utility function for sensors in our MSN scenario based on the information form of the Kalman filter. Since these utility function exhibit interdependence, we could not use standard resource allocation mechanisms. Thus we developed a generic mechanism for interdependent valuations that significantly extends the standard VCG mechanism and proved that the ensuing mechanism has the ideal economic properties of being efficient, incentive compatible and individually rational. We then showed that this more complex mechanism only increases the centre's computational burden and the bidding for the agents (which are more likely to be computationally constrained) is no more demanding than that for the VCG.

In Chapter 6, we considered the case where agents are uncertain about whether other agents will successfully complete their allocated tasks and have different perceptions about the probability of success of other agents in the system. In order to deal with this problem, we developed a trust-based mechanism and proved that it is efficient, individually rational, and incentive compatible. We then demonstrated the generality of the mechanism by reducing it to two known mechanisms, namely the VCG mechanism and Porter et al.'s fault-tolerant mechanism. We also considered the computational properties of this mechanism and showed that the allocation problem is $\mathcal{NP}-complete$. Furthermore, we developed algorithms based on dynamic programming so as to speed up the generation of possible allocations. Finally, we empirically evaluated our mechanism against the fault-tolerant mechanism and showed that, unlike the fault-tolerant mechanism, it is robust to bias in the system.

Looking back at the research requirements outlined in Chapter 1, the research carried out in this thesis has successfully addressed each of them:

- **Distributed allocation**: We have studied two different distributed allocation mechanisms, namely a simultaneous second-price auction in Chapter 3 and a continuous double auction in Chapter 4. In the former case, in order to study the efficiency of the mechanism, we first derived the optimal strategy of an agent under different market conditions. In the latter case, we modified the standard CDA protocol so as to achieve distributed allocation of tasks when agent have a certain capacity to which they can supply. We then investigated the effect that distribution had on the efficiency of the system.

- **Constrained capacity**: In Chapter 4, we designed both a centralised and a decentralised protocol for the case where agents have a constrained capacity. The former modified the VCG mechanism by introducing a penalty scheme which ensures the economic properties of incentive-compatibility, efficiency and individual rationality are preserved. We also proved the robustness of the mechanism. The latter was based on the CDA.

- **Distributed (Interdependent) valuations**: We developed an efficient, incentive compatible and individually-rational mechanism in Chapter 5 when agents have interdependent valuations and are willing to acquire multiple items. Chapter 6 also studied a form of distributed information by analysing a scenario where agents learn through the distributed experiences of all agents. This results in a distributed information scenario since the agents require information about the distributed experience in order to know their expected valuation. We also developed an efficient, incentive-compatible and individually-rational mechanism in this case.

- **Uncertainty in task completion**: In Chapter 6, we studied the case where there is uncertainty as to whether the agents will actually fulfill their assigned tasks. We developed an efficient, incentive-compatible and individually-rational mechanism for this case. We also considered its computational properties and developed algorithms for speeding up the computation of the task allocation and payments.

When taken together, this thesis has made significant advances in the state of the art of distributed mechanisms for multiagent systems. However, much still remains to be done.

7.2 Future Work

Despite these achievements, there are still many issues that need to be addressed. On a theoretical level, there is a need to unify the different strands of distributed mechanisms so as to improve their applicability in multi-agent systems. Thus, we require a mechanism that achieves distributed allocation and where the agents have distributed information. Furthermore, the distributed allocation mechanisms studied within this thesis suffer from a lack of efficiency. Hence future work should concentrate on ways of achieving distributed allocation, whilst still conserving efficiency. To achieve this goal, it is important to distribute the two tasks carried out by a centre, namely the computation of the optimal allocation and the enforcement of these allocations. One potential area to look for insights is in work on distributed constraint optimisation algorithms [Modi et al., 2003; Mailler and Lesser, 2004] which distribute the computation of optimisation problems over agents within the system. Furthermore, distributed enforcement mechanism (like the one studied in [Blankenburg et al., 2005]) should be investigated in the context of mechanism design. Finally one could investigate hierarchical systems whereby the control of the centre is devolved to multiple centres who each have a subset of agents to control.

On a practical level, we have designed mechanisms that address each of the requirements arising within multi-sensor networks. Future work should concentrate on addressing all these issues simultaneously. Furthermore, the context of the application of the sensor networks will often dictate the valuation/cost that these sensors have. Thus, there is a need to develop a more general valuation function for the sensor networks that will consider the specificities of the context in which they are deployed. Finally, the implementation of the mechanisms can involve a high computational load. Thus, work is needed to develop algorithms that make these mechanisms more tractable (perhaps based on techniques borrowed from linear and integer programming). In cases where this is not possible, then a relaxation of the goals of the mechanism may be required.

Having described future work on a broader plane, we now identify the following promising directions for further research that stem from the specifics of the work discussed in each chapter:

- Chapter 3: Our analysis of simultaneous second price auctions focused on the case where buyers wish to have a single item. Future work can expand this to the case of multiple items and where buyers have combinatorial valuations. Also, optimal equilibrium strategies for purchasing item in markets consisting of different types of auctions still remain to be investigated. These would lead to a more general distributed allocation mechanism and improve the applicability of this research within general multi-agent systems settings.
- Chapter 4: As future work in this chapter, one can extend these mechanisms to deal with iterated allocations (i.e. ones in which new demand continuously appears) since in several of the cases we consider it is conceivable that the agents can observe and learn about the behaviours of other agents in the system. Also a deeper study is required to

formally establish the consequence of requiring robust mechanisms on the efficiency of the resultant mechanism. Finally, we aim to develop more sophisticated strategies for the decentralised mechanism in order to enhance the efficiency of the system, whilst ensuring that these sophisticated strategies derive higher profit than their simpler counterparts. This has been shown to be achievable in simple continuous double auctions. [Cliff and Bruten, 1997; Gjerstad and Dickhaut, 1998; Vytelingum et al., 2004] and we believe it is also achievable in our modified continuous double auction protocol. Such developments will enable us to more effectively find the set of agents that can perform the required task at the lowest cost (i.e. the efficiency will be increased).

- Chapter 5: This chapter developed a valuation function from a relatively simple information theoretic base. An extension to this work can consider more complex information theoretic measures (such as the Kullback-Liebler divergence and the Mahanalobis distance measure) and to also take into account the relative importance of targets. Another line of work could consider the design of a distributed mechanism for choosing the optimal allocation and calculating the payment. To this end, by showing that a centralised mechanism exists, one of the necessary conditions for the existence of a decentralised mechanism has been satisfied. Given this, we intend to explore techniques such as those developed in [Parkes and Shneidman, 2004] in order to develop a distributed form of this mechanism. However, it is important to point out that in our mechanism, as it currently stands, the agents only transmit a representative value to the centre (rather than the data itself). Thus, any distributed data fusion algorithm can conceivably be implemented in our scenario as long as we can formulate such a representative value (which would typically have a much lower bandwidth requirement than the data itself).

- Chapter 6: In this chapter, the focus was on an efficient mechanism which therefore required exact solutions. In future work we aim to find an approximate mechanism that is guaranteed to be efficient within a certain bound. This reduces the extra computational burden involved when taking into account trust in combinatorial exchanges. It will also allow the development of local search algorithms that will further reduce the computation involved in finding the efficient allocation. Furthermore, our current mechanism is incentive-compatible, thus providing no incentive for agents to deviate from truthful behaviour within a single-shot allocation. In future, we aim to investigate iterative mechanisms which prevent agents from strategizing over rounds and induces truthful behaviour across rounds.

Bibliography

E. Adar and B. Huberman. Free Riding on Gnutella. *First Monday*, 5(10), October 2000.

S. Airiau and S. Sen. Strategic Bidding for Multiple Units in Simultaneous and Sequential Auctions. *Group Decision and Negotiation*, 12(5), 2003.

I. F. Akyildiz, W. Su, Y. Sankarasubramanian, and E. Cayirci. A Survey on Sensor Networks. *IEEE Communications Magazine*, 40(8):102–114, 2002.

L. Anderegg and S. Eidenbenz. Ad-hoc VCG: a Truthful and Cost-efficient Routing Protocol for Mobile Ad-hoc Networks with Selfish Agents. In *Proc. Ninth International Conference on Mobile Computing and Networking (MOBICOM '03)*, pages 245–259, 2003.

A. Andersson, M. Tenhunen, and F. Ygge. Integer Programming for Combinatorial Auction Winner Determination. In *Proc. Fourth International Conference on Multiagent Systems (ICMAS '00)*, pages 39–46, Boston,USA, 2000.

P. Anthony and N. R. Jennings. Developing a Bidding Agent for Multiple Heterogeneous Auctions. *ACM Transactions on Internet Technology*, 3(3):185217, 2003.

K. J. Arrow. The Property Rights Doctrine and Demand Revenue Selection under Incomplete Information. In *Economics and Human welfare: Essays in Honor of Tibor Scitovsky*, pages 23–29. New York Academic Press.

Y. Bachrach and J. S. Rosenschein. Achieving Allocatively-Efficient and Strongly Budget-Balanced Mechanisms in the Network Flow Domain for Bounded-Rational Agents. In *Proc. Nineteenth International Joint Conference on Artificial Intelligence*, pages 1653–1654, Edinburgh, Scotland, August 2005.

C. Berge. *Graphs and Hypergraphs*. North-Holland Publishing Company, 1973.

T. Berners-Lee. *Weaving the Web*. Harper, San Francisco, 1999.

M. Bichler and J. Kalagnanam. Configurable offers and Winner Determination in Multi-Attribute Auctions. *European Journal of Operational Research*, 160(2):380–394, 2005.

G. R. Bitran and H. H. Yanasse. Computational Complexity Of The Capacitated Lot Size Problem. *Management Science*, (28):1174–1186, 1982.

B. Blankenburg, R. K. Dash, S. D. Ramchurn, M. Klusch, and N. R. Jennings. Trusted kernel-based coalition formation. In *Proc. Fourth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS '04)*, pages 989–996, Utrecht, Netherlands, 2005.

C. Boutilier. Multiagent Systems: Challenges and Opportunities for Decision-Theoretic Planning. *Al Magazine*, 20(4):35–43, 1999.

J. Bredin, D. Kotz, and D. Rus. Market-based Resource Control for Mobile Agents. In *Proceedings of the Second International Conference on Autonomous Agents (Agents '98)*.

A. Byde, C. Preist, and N.R. Jennings. Decision Procedures for Multiple Auctions. In *Proc. First International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2000)*, page 613620, Bologna, Italy, 2000.

H. Chen, C. Chu, and J. M. Proth. An Improvement of the Lagrangian Relaxation Approach for Job Shop Scheduling: A Dynamic Programming Method. *IEEE Transactions on Robotics and Automation*, 14:786–795, 1998.

M. Chu, H. Haussecker, and F. Zhao. Scalable Information-Driven Sensor Querying and Routing for Ad-Hoc Heterogeneous Sensor Networks. *The International Journal of High Performance Computing Applications*, 16(3):293–314, 2002.

E. H. Clarke. Multipart Pricing of Public Goods. *Public Choice*, 11:17–33, 1971.

S. H. Clearwater, editor. *Market-Based Control- A Paradigm for Distributed Resource Allocation*. World Scientific, 1996.

D. Cliff and J. Bruten. Minimal-Intelligence Agents for Bargaining Behaviors in Market-Based Environments. Technical Report HPL-97-91, 1997.

V. Conitzer and T. Sandholm. Complexity of Mechanism Design. In *Proc. Uncertainty in Artificial Intelligence Conference (UAI '02)*, pages 103–110, Edmonton, Canada, 2002.

T. H. Cormen, C. E. Leiserson, and R. R. Rivest. *Introduction to Algorithms*. The MIT Press, Cambridge, Masachusetts, 1990.

P. Cramton. The FCC Spectrum Auctions: An Early Assessment. *Journal of Economics and Management Strategy*, 6(3):431–495, 1997.

D. Culler, D. Estrin, and M. Srivastava. Overview of Sensor Networks. *IEEE Computer*, 37(8): 41– 49, 2004.

V. D. Dang, R. K. Dash, A. Rogers, and N. R. Jennings. Overlapping coalition formation for efficient data fusion in multi-sensor networks. In *Proc. 21st National Conference on AI (AAAI)*, Boston, USA, 2006.

P. Dasgupta. Trust as a Commodity. In D. Gambetta, editor, *Trust: Making and Breaking Cooperative Relations*, pages 49–72. Blackwell, 1998.

P. Dasgupta and E. Maskin. Efficient Auctions. *Quarterly Journal of Economics*, 115:341–388, 2000.

R. K. Dash, D. C. Parkes, and N. R. Jennings. Computational Mechanism Design: A Call to Arms. *IEEE Intelligent Systems*, 18(6):40–47, 2003.

R. K. Dash, S. D. Ramchurn, and N. R. Jennings. Trust-based mechanism design. In *Proc. Third International Conference on Autonomous Agents and Multi-Agent Systems*, pages 748–755, New York, USA, 2004.

R. K. Dash, A. Rogers, S. Reece, S. Roberts, and N. R. Jennings. Constrained bandwidth allocation in multi-sensor information fusion: a mechanism design approach. In *Proc. Eighth International Conference on Information Fusion*, Philadelphia, USA, 2005.

C. d'Aspremont and L.A. Gerard-Varet. Incentives and incomplete information. *Journal of Public Economics*, 11:25–45, 1979.

E. David, A. Rogers, J. Schiff, S. Kraus, and N. R. Jennings. Optimal design of English auctions with discrete bid levels. In *Proc. of Sixth ACM Conference on Electronic Commerce (EC'05)*, pages 98–107, Vancouver, Canada, 2005.

C. Dellarocas. Goodwill Hunting: An Economically Efficient Online Feedback Mechanism for Environments with Variable Product Quality. In *Proceedings of the the workshop on Agent-Mediated Electronic Commerce*, pages 238–252, Bologna, Italy, 2002.

A. Deshpande, C. Guestrin, S. Madden, J. Hellerstein, and W. Hong. Model-based Approximate Querying in Sensor Networks. *International Journal on Very Large Data Bases*, pages 588–599, 2004.

S. deVries and R. Vohra. Combinatorial Auctions: A Survey. *INFORMS Journal on Computing*, 15(3):284–309, 2003.

E. H. Durfee and V. Lesser. Negotiating task decomposition and allocation using partial global planning. In L. Gasser and M. Huhns, editors, *Distributed Artificial Intelligence*, volume II, pages 229–244. Pitman Publishing, San Mateo, CA, 1989.

R. Engelbrecht-Wiggans and R. Weber. An Example of a Multiobject Auction Game. *Management Science*, 25:1272–1277, 1979.

M. Eso, S. Ghosh, J. R. Kalagnanam, and L. Ladanyi. Bid Evaluation in Procurement Auctions with Piecewise Linear Supply Curves. Technical Report RC 22219, IBM Research, Yorktown Heights, NY, 10598, 2001.

S. S. Fatima, M. Wooldridge, and N. R. Jennings. Sequential Auctions for Objects with Common and Private Values. In *Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-05)*, pages 635–642, 2005.

J. Feigenbaum, C. H. Papadimitriou, R. Sami, and S. Shenker. A BGP-based Mechanism for Lowest-Cost Routing. In *Proceedings of the Twenty-First Symposium on Principles of Distributed Computing, ACM Press*, pages 173–182, New York,USA, 2002.

J. Feigenbaum, C.H. Papadimitriou, and S. Shenker. Sharing the Cost of Multicast Transmissions. *Journal of Computer and System Sciences*, 63(1):21–41, 2001.

M. Florian, J. K. Lenstra, and H. G. Rinnooy Kan. Deterministic production planning: Algorithms and complexity. *Management Science*, 26:669–679, 1980.

I. Foster and C. Kesselman, editors. *The Grid: Blueprint for a New Computing Infrastructure*. Morgan Kaufmann Publishers, Inc., San Francisco, USA, 1999.

D. Friedman and J. Rust, editors. *The Double Auction Market: Institutions, Theories and Evidence*. Addison-Wesley, New York, 1992.

D. Fudenberg and D. K. Levine. Self-Confirming Equilibrium. *Econometrica*, 61(3):523–545, 1993.

Y. Fujishima, K. Leyton-Brown, and Y. Shoham. Taming the Computational Complexity of Combinatorial Auctions: Optimal and Approximate Approaches. In *International Joint Conference on Artificial Intelligence (IJCAI '99)*, pages 548–553, Stockholm, Sweden, 1999.

M. R. Garey and D. S. Johnson. *Computers and Intractability - A Guide to the Theory of NP-Completeness*. Freeman, San Francisco, 1979.

E. H. Gerding, R. K. Dash, D. C. Yuen, and N. R. Jennings. Optimal Bidding Strategies for Simultaneous Vickrey Auctions with Perfect Substitutes. In *Proc. Eighth Trading Agent Design and Analysis/Agent Mediated E-Commerce (TADA/AMEC) joint workshop at AAMAS 2006*, 2006a.

E. H. Gerding, A. Rogers, R. K. Dash, and N. R. Jennings. Competing Sellers in Online Markets: Reserve Prices, Shill Bidding, and Auction Fees. In *Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'06)*, pages 1208–1210, Hakodate, Japan, 2006b.

B. P. Gerkey and M. J. Mataric. A market-based formulation of sensor-actuator network coordination. In *Proceedings of the AAAI Spring Symposium on Intelligent Embedded and Distributed Systems*, pages 21–26, Palo Alto, California, United States, 2002a.

B. P. Gerkey and M. J. Mataric. Sold!: Auction methods for multi-robot coordination. *IEEE Transactions on Robotics and Automation, Special Issue on Multi-robot Systems*, 2002b.

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

A. Giovannucci, J. A. Rodrguez-Aguilar, A. Reyes-Moro, F. X. Noria, and J. Cerquides. Towards automated procurement via agent-aware support. In *Proc. Third International Conference on Autonomous Agents and Multi-Agent Systems*, pages 244–251, New York, USA, 2004.

S. Gjerstad and J. Dickhaut. Price Formation in Double Auctions. *Games and Economic Behavior*, 22:1–29, 1998.

J. Glazer and A. Rubinstein. Motives and Implementation: On the Design of Mechanism to Elicit Opinions. *Journal of Economic Theory*, pages 157–173, 1998.

D. K. Gode and S. Sunder. Allocative efficiency of markets with zero-intelligence traders: Market as a partial substitute for individual rationality. *Journal of Political Economy*, 101(1): 119–137, 1993.

M. Gondrand and M. Minoux. *Graphs and Algorithms*. John Wiley and Sons, 1986.

A. Greenwald, , R.M. Kirby, Jon Reiter, and J. Boyan. Bid Determination in Simultaneous Auctions: A Case Study. In *Proc. of the Third ACM Conference on Electronic Commerce*, pages 115–124, Florida, United States, 2001.

J. Gross and J. Yellen. *Graph Theory and its Applications*. CRC Press, Boca Raton, Florida, USA, 1999.

T. Groves. Incentives in teams. *Econometrica*, 41:617–631, 1973.

F. Gul and E. Stacchetti. The English Auction with Differentiated Commodities. *Journal of Economic Theory*, pages 66–95, 2000.

M. He, N. R. Jennings, and A. Prgel-Bennett. A Heuristic Bidding Strategy for Buying Multiple Goods in Multiple English Auctions. *ACM Trans on Internet Technology*, 2006. To appear.

M. He, H. F. Leung, and N. R. Jennings. A Fuzzy Logic Based Bidding Strategy for Autonomous Agents in Continuous Double Auctions. *IEEE Trans on Knowledge and Data Engineering*, 15 (6):1345–1363, 2003.

H. Heidt, J. Puig-Suari, A. S. Moore, S. Nakasuka, and R. J. Twiggs. CubeSat: A new Generation of Picosatellite for Education and Industry Low-Cost Space Experimentation. In *14th Annual AIAA/USU Conference on Small Satellites*, 2000.

B. H. Hobbs, C. B. Metzler, and J.-H. Pang. Strategy Gaming Analysis for Electric Power Systems: An MPEC approach. *IEEE Transactions on Power Systems*, 15:638645, 2000.

K. L. Hoffman and M. Padberg. Solving Airline Crew Scheduling Problems by Branch-and-Cut. *Management Science*, 39(6):657–682, 1993.

A. C. Huang, B. C. Ling, and S. Ponnekanti. Pervasive Computing: What is it Good for? In *Proceedings of the ACM International Workshop on Data Engineering for Wireless and Mobile Access*, pages 84–91, Seattle, Washington, United States, 1999.

L. Hurwicz. On Informationally Decentralized Systems. In C. McGuire and Roy Radner, editors, *Decision and Organisation : A Volume in Honor of Jacob Marchak*.

S. Iyengar and R. Brooks, editors. *Distributed sensor networks*. Chapman and Hall, 2005.

M. O. Jackson. Mechanism Theory. In *The Encyclopedia of Life Support Systems*. EOLSS Publishers, 2000.

P. Jehiel and B. Moldovanu. Efficient Design with Interdependent Valuations. *Econometrica*, 69(5):1237–59, 2001.

N. R. Jennings. An Agent-Based Approach for Building Complex Software Systems. *Communications of the ACM*, 44(4):35–41, 2001.

N. R. Jennings and S. Bussmann. Agent-Based Control Systems. *IEEE Control Systems Magazine*, 23(3):61–74, 2003.

N. R. Jennings, S. D. Ramchurn, M. Allen-Williams, R. K. Dash, P. S. Dutta, A. Rogers, and I. Vetsikas. The ALADDIN Project: Agent Technology To The Rescue. In *Proc. First AAMAS Workshop on Agent Technology for Disaster Management (ATDM)*, Hakodate, Japan, 2006. To appear.

R. Jurca and B. Faltings. An Incentive Compatible Reputation Mechanism. In *Proc. of the International Joint Conference on Autonomous Agents and Multi-Agent Systems*, pages 1026–1027, 2003.

R.E. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Transaction of the AMSE — Journal of Basic Engineering*, pages 35–45, 1960.

P. Klemperer. What Really Matters in Auction Design. *The Journal of Economic Perspectives*, 16(1):169–189, 2002.

A. Kothari, D. C. Parkes, and S. Suri. Approximately-Strategyproof and Tractable Multi-Unit Auctions. In *Fourth ACM Conf. on Electronic Commerce (EC'03)*, 2003.

S. Kraus. Automated negotiation and decision making in multiagent environments. pages 150–172, 2001.

Sarit Kraus, Katia Sycara, and Amir Evenchik. Reaching Agreements through Argumentation: A Logical Model and Implementation. *Artificial Intelligence*, 104(1–2):1–69, 1998.

V. Krishna. *Auction Theory*. Academic Press, 2002.

V. Krishna and R. Rosenthal. Simultaneous Auctions with Synergies. *Games and Economic Behaviour*, 17:1–31, 1996.

K. Lai, M. Feldman, I. Stoica, and J. Chuang. Incentive for cooperation in Peer-to-Peer networks. In *Proc. First Workshop on Economics of Peer-to-Peer Systems*, Berkeley, CA, June 5-6 2003.

K. Lang and R. Rosenthal. The Contractor's Game. *RAND Journal Economics*, 22:329–338, 1991.

D. Lehmann, L. Ita O'Callaghan, and Y. Shoham. Truth Revelation in Approximately Efficient Combinatorial Auctions. *Journal of the ACM*, 49(5):577–602, September 2002.

V. Lesser, C. Ortiz, and M. Tambe, editors. *Distributed Sensor Networks: a Multiagent Perspective*. Kluwer Publishing, 2003.

Victor R. Lesser and Daniel D. Corkill. Functionally accurate, cooperative distributed systems. *IEEE Transactions on Systems, Man, and Cybernetics*, 11(1):81–96, 1981.

K. Leyton-Brown, M. Pearson, and Y. Shoham. Towards a Universal Test Suite for Combinatorial Auction Algorithms. In *ACM Conf. on Electronic Commerce (EC'00)*, pages 66–76, 2000.

R. Mailler and V. Lesser. Solving Distributed Constraint Optimization Problems using Cooperative Mediation. In *Proc. Third International Conference on Autonomous Agents and Multi-Agent Systems*, pages 438–445, New York, USA, 2004.

J. Manyika and H.F. Durrant-Whyte. *Data Fusion and Sensor Management: A Decentralized Information-Theoretic Approach*. Ellis Horwood, 1997.

S. Martello and P. Toth. *Knapsack Problems, Algorithms and Computer Implementations.* John Wiley and Sons Ltd, England, 1990.

A. MasColell, M. Whinston, and J.R. Green. *Microeconomic Theory*. Oxford University Press, 1995.

R.P. McAfee. A Dominant Strategy Double Auction. *Journal of Economic Theory*, 56:434–450, 1992.

K. A. McCabe, S. J. Rassenti, and V. L. Smith. Designing Call Auction Institution: Is Double Dutch The Best? *The Economic Journal*, 102(410):9–23, 1992.

C. Mezzetti. Mechanism Design with Interdependent Valuations: Efficiency and Full Surplus Extraction. Technical report, University of North Carolina, February 2003.

R. Mirrlees. An Exploration in the Theory of Optimum Income Taxation. *Review of Economic Studies*, 38:175–208, 1971.

Pragnesh Jay Modi, Wei-Min Shen, Milind Tambe, and Makoto Yokoo. An Asynchronous Complete Method for Distributed Constraint Optimization. In *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, 2003.

D. Monderer and M. Tennenholtz. Distributed games. *Games and Economic Behavior*, 28:5572, 1999.

R.B. Myerson and M. A. Satterthwaite. Efficient mechanisms for bilateral trading. *Journal of Economic Theory*, 29:265–281, 1983.

J. Nicolaisen, V. Petrov, and L. Tesfatsion. Market Power and Efficiency in a Computational Electricity Market with Discriminatory Double-Auction Pricing. *IEEE Transactions on Evolutionary Computation*, 5(5):504–523, 2001.

N. Nisan. Bidding and allocation in combinatorial auctions. In *ACM Conference on Electronic Commerce*, pages 1–12, 2000.

N. Nisan and A. Ronen. Algorithmic Mechanism Design. In *Proc. 31st ACM Symp. on Theory of Computing*, pages 129–140, 1999.

M.J. Osborne and A. Rubinstein. *A Course in Game Theory*. The MIT Press, 1994.

P. Padhy, R. K. Dash, K. Martinez, and N. R. Jennings. A Utility-Based Sensing and Communication Model for a Glacial Sensor Network. In *Proc. Fifth International Conference on Autonomous Agents and Multi-Agent Systems*, pages 1353–1360, Hakodate, Japan, 2006.

C. H. Papadimitriou and K. Steiglitz. *Combinatorial Optimization: Algorithms and Complexity*. Prentice-Hall, 1982.

C.H. Papadimitriou. Algorithms, Games, and the Internet. In *Proceedings on Thirty-Third Annual ACM Symposium on Theory of Computing,*, pages 749–753, 2001.

D. Parkes. *Iterative Combinatorial Auctions:Achieving Economic and Computational Efficiency*. PhD thesis, University of Pennsylvania, May 2001.

D. Parkes, J. Kalagnanam, and M. Eso. Achieving Budget-Balance with Vickrey-Based Payment Schemes in Exchanges. In *17th Int. Joint Conf. on Art. Intell. (IJCAI'01)*, pages 1161–1168, 2001.

D. C. Parkes and J. Schneidman. Approximately Efficient Online Mechanism Design. In *Proc. of the Eighteenth Annual Conference on Neural Information Processing Systems (NIPS'04)*, 2004.

D. C. Parkes and J. Shneidman. Distributed Implementations of Vickrey-Clarke-Groves Mechanism. In *Proc. Third International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS '04)*, pages 261–268, New York, USA, 2004.

R. Porter, A. Ronen, Y. Shoham, and M. Tennenholtz. Mechanism Design with Execution Uncertainty. In *Proceedings of Uncertainty in Artificial Intelligence (UAI '02)*, pages 414–421, 2002.

C. Preist, A. Byde, and C. Bartolini. Economic Dynamics of Agents in Multiple Auctions. In *Proc. of the Fifth International Conference on Autonomous Agents*, pages 545–551, Montreal, Canada, 2001.

D.V. Pynadath and M. Tambe. Automated teamwork among heterogeneous software agents and humans. *Journal of Autonomous Agents and Multi-Agent Systems*, 7:71–100, 2003.

B.R. Radrinath, A. Acharya, and T. Imelinski. Impact of Mobility on Distributed Computations. *Operation Systems Review*, April 1993.

K. Ramamritham, J. A. Stankovic, and W. Zhao. Distributed Scheduling of Tasks with Deadlines and Resource Requirements. *IEEE Trans. Computing*, 38(8):1110–1123, 1989.

S. D. Ramchurn, D. Huynh, and N. R. Jennings. Trust in Multi-Agent Systems. *The Knowledge Engineering Review*, 19:1–25, 2004.

B. Rao, H. Durrant-Whyte, and A. Sheen. A Fully Decentralised Multi-Sensor System for Tracking and Surveillance. *International Journal of Robotics Research*, 12(1):20–45, 1991.

A. Rogers, E. David, and N.R. Jennings. Self Organised Routing for Wireless Micro-sensor networks. *IEEE Trans. on Systems, Man and Cybernetics: Part A*, 35(3):349–359, 2005.

J.S. Rosenschein and G. Zlotkin. *Rules of Encounter*. MIT Press, 1994.

R. Rosenthal and R. Wang. Simultaneous Auctions with Synergies and Common Values. *Games and Economic Behaviour*, 17:32–55, 1996.

A.E. Roth and I. Erev. Learning in Extensive Form Games: Experimental Data and Simple Dynamic Models in the Intermediate Term. *Games and Economic Behaviour*, 8:164–212, 1995.

A.E. Roth and A. Ockenfels. Last-Minute Bidding and the Rules for Ending Second-Price Auctions: Evidence from eBay and Amazon Auctions on the Internet. *The American Economic Review*, 92(4):1093–1103, 2002.

M. H. Rothkopf, A. Pekec, and R. M. Harstad. Computationally Manageable Combinatorial Auctions. *Management Science*, 44:1131–1147, 1998.

T. Roughgarden. *Selfish Routing*. PhD thesis, Cornell University, 2002.

A. Rubinstein. *Modeling Bounded Rationality*. MIT Press, 2002.

N. Sadagopan and B. Krishnamachari. Decentralized utility based sensor network design. In *2nd Workshop on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, 2004.

T. Sandholm. Algorithm for optimal winner determination in combinatorial auctions. *Artificial Intelligence*, 135(1-2):1–54, 2002a.

T. Sandholm. eMediator: A next generation electronic commerce server. *Computational Intelligence*, 18(4):656–676, 2002b.

T. Sandholm. Making Markets and Democracy Work: A Story of Incentives and Computing. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1649–1671, 2003.

T. Sandholm, S. Suri, A. Gilpin, and D. Levine. Winner determination in combinatorial auction generalizations. In *Proceeding of the First International Joint Confernce on Autonomous Agents and Multiagent Systems*, pages 69–76, 2001.

D. Sarne and S. Kraus. Solving the Auction-Based Task Allocation Problem in and Open Environment. In *AAAI*, pages 164–169, 2005.

J. Shneidman and D. Parkes. Rationality and Self-Interest in Peer to Peer Networks. In *2nd International Workshop on Peer to Peer Systems*, 2003. To appear.

Y. Shoham and M. Tennenholtz. Emergent conventions in multi-agent systems. *Proceedings of Knowledge Representation and Reasoning*, pages 225–231, 1992.

S. S. Skiena. *The Algorithm Design Manual*. Springer-Verlag, New York, 1998.

V. L. Smith. An experimental study of competitive market behaviour. *Journal of Political Economy*, 70:111–137, 1962.

B. Szentes and R. Rosenthal. Three-object Two-bidder Simultaeous Auctions:chopsticks and tetrahedra. *Games and Economic Behaviour*, 44:114–133, 2003.

Gerald Tesauro and Jonathan L. Bredin. Strategic sequential bidding in auctions using dynamic programming. In *Proceedings of the First International Joint Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*, pages 591–598. ACM Press, 2002.

H. Varian. *Intermediate Microeconomics : A Modern Approach*. W.W.Norton and Co., fifth edition, 1999.

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

P. Vytelingum, R. K. Dash, E. David, and N. R. Jennings. A risk-based bidding strategy for continuous double auctions. In *Proc. 16th European Conference on Artificial Intelligence*, Valencia, Spain, 2004. to appear.

M.P. Wellman. A Market-Oriented Programming Environment and its Application to Distributed Multicommodity Flow Problems. *Journal of Artificial Intelligence Research*, 1:1–23, 1993.

M.P. Wellman. Online Marketplaces. In M.P. Singh, editor, *Practical Handbook of Internet Computing*. CRC Press, 2004.

M.P. Wellman, D.M. Reeves, K.M. Lochner, and Y. Vorobeychik. Price prediction in a trading agent competition. *Journal of Artificial Intelligence Research*, (21):1936, 2004.

R. Wolski, J. S. Plank, J. Brevik, and T. Bryan. Analyzing Market-Based Resource Allocation Strategies for the Computational Grid. *The International Journal of High Performance Computing Applications*, 15(3):258–281, Fall 2001.

H. Wu, J. Lee, M. Hunter, R. M. Fujimoto, R. L. Guensler, and J. Ko. Simulated Vehicle-to-Vehicle Message Propagation Efficiency on Atlantas I-75 Corridor. *Transportation Research Record, Journal of the Transportation Research Board*, 2005.

P. Wurman, M. Wellman, and W. Walsh. A Parametrization of the Auction Design Space. *Games and Economic Behavior*, 35:304–338, 2001.

Mu Xia, Jan Stallaert, and Andrew Whinston. Solving the combinatorial double auction problem. *European Journal of Operational Research*, 164(1):239–251, 2004.

I. Yarom, Z. Topol, and J. S. Rosenschein. Decentralized Marketplaces and Web Services. In *The Workshop on Agent Mediated Electronic Commerce (AMEC-VI), at the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 267–276, New York, July 2004.

M. Yokoo, E.H.Durfee, T. Ishida, and K. Kuwabara. Distributed Constraint Satisfaction Problems: Formalization and Algorithms. *IEEE Trans. on Knowledge and Data Engineering*, 10 (5):673–685, 1998.

F. Zhao, J. Shin, and J. Reich. Information-Driven Dynamic Sensor Collaboration for Tracking Applications. *IEEE Signal Processing Mag.*, pages 61–72, March 2002.

G. Zlotkin and J. S. Rosenschein. Mechanism Design for Automated Negotiation, and its Application to Task Oriented Domains. *Artificial Intelligence*, 86(2):195–244, 1996.