MARKET-BASED GRID RESOURCE ALLOCATION
USING A STABLE CONTINUOUS DOUBLE
AUCTION

A THESIS SUBMITTED TO THE UNIVERSITY OF MANCHESTER
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
IN THE FAULTY OF ENGINEERING AND PHYSICAL SCIENCES

2007

Zhu Tan
School of Computer Science
Contents

CONTENTS...................................................................................................................................2

LIST OF TABLES................................................................................................................................6

LIST OF FIGURES.........................................................................................................................8

ABSTRACT ................................................................................................................................... 11

DECLARATION .............................................................................................................................13

COPYRIGHT ....................................................................................................................................14

ACKNOWLEDGEMENTS........................................................................................................15

1 INTRODUCTION ............................................................................................................17

1.1 COMPUTATIONAL GRID RESOURCE ALLOCATION......................................................... 18

1.2 GRID SCHEDULING REQUIREMENTS ............................................................................20

1.3 DYNAMIC PRICING MECHANISMS........................................................................... 23

1.4 EFFICIENCY .................................................................................................................. 26

1.4.1 Pareto Efficiency and Aggregate Surplus........................................................................26

1.4.2 Scheduling Efficiency .............................................................................................31

1.4.3 Overall Efficiency .................................................................................................33

1.5 CONTRIBUTIONS OF THIS THESIS...............................................................................33

2 MARKET FORMULATIONS..............................................................................................37

2.1 THE SPECTRUM OF MARKET FORMULATIONS............................................................39
2.2 SINGLE-SIDED AUCTIONS ................................................................. 40

2.2.1 Four Major Single-Sided Auction Types ........................................ 40

2.2.2 Strategy-proofness ........................................................................ 42

2.2.3 Applying Single-sided Auctions to Grid Resource Allocation ............. 44

2.3 MECHANISM DESIGN ................................................................. 45

2.3.1 Introduction ................................................................................ 45

2.3.2 The Vickrey-Clarke-Groves Mechanism ........................................... 47

2.4 DOUBLE-SIDED AUCTIONS ....................................................... 48

2.4.1 Discrete-time Double Auctions ..................................................... 48

2.4.2 Continuous-time Double Auctions ................................................. 50

2.4.3 Hybrid Auctions .......................................................................... 54

2.5 COMBINATIONAL AUCTIONS .................................................... 55

2.6 COMMODITY MARKET MODEL .................................................. 58

2.7 SUMMARY .................................................................................... 59

3 ANALYSIS OF THE CONTINUOUS DOUBLE AUCTION ..................... 63

3.1 CONTEXT AND TERMINOLOGY .................................................. 64

3.2 ZERO INTELLIGENCE STRATEGY ............................................... 68

3.3 KAPLAN STRATEGY ...................................................................... 69

3.4 FIXED MARK-UP STRATEGY ....................................................... 71

3.5 ZERO INTELLIGENCE PLUS STRATEGY ....................................... 72
3.6 Gjerstad Dickhaunt Strategy ................................................................. 74
3.7 Fuzzy-logic Based Strategy ..................................................................... 75
3.8 Observations ......................................................................................... 76

4 SYNTHESIS OF THE STABLE CONTINUOUS DOUBLE AUCTION .......... 79

4.1 Architecture of the Stable Continuous Double Auction ...................... 81
4.2 Principles of the Price Adjustment Mechanism .................................... 84
4.3 Implementation of the Price Adjustment Mechanism ........................... 87
  4.3.1 Mandani Fuzzy Controller ............................................................... 88
  4.3.2 Fuzzy Sets and Membership Functions for the Price Adjustment Mechanism ... 90
  4.3.3 Fuzzy Reasoning Rules and Heuristics of the Price Adjustment mechanism .... 93

4.4 Summary ............................................................................................... 100

5 EXPERIMENTAL ENVIRONMENT .......................................................... 102

5.1 Architecture .......................................................................................... 103
5.2 Experiment Settings ............................................................................... 109
5.3 Construction of Experiments ................................................................. 114
  5.3.1 Creating the Continuous Double Auction and Agents with Different Bidding Strategies ........................................................................................................ 114
  5.3.2 Building the Stable Continuous Double Auction ............................. 117
  5.3.3 Extending to Evaluate Scheduling Efficiency .................................... 119
5.4 Summary ............................................................................................... 120

6 Evaluation .................................................................................................. 121
6.1 RESULTS FOR THE CONTINUOUS DOUBLE AUCTION ................................................................. 122
  6.1.1 Populated with Truth-Telling Agents ................................................................. 123
  6.1.2 Populated with Zero Intelligence Agents ....................................................... 129
  6.1.3 Efficiency Comparison of Different Strategies .............................................. 133

6.2 RESULTS FOR THE STABLE CONTINUOUS DOUBLE AUCTION................................. 137
  6.2.1 Populated with Truth-Telling Agents ............................................................. 138
  6.2.2 Populated with Zero Intelligence Agents ....................................................... 143
  6.2.3 Efficiency Comparison of Different Strategies .............................................. 147

6.3 ANALYSIS OF SCHEDULING EFFICIENCY ................................................................. 148
  6.3.1 Single Resource Requests .............................................................................. 149
  6.3.2 Multiple Resource Requests .......................................................................... 151

6.4 SUMMARY ............................................................................................................... 153

7 CONCLUSION .............................................................................................................. 158
  7.1 THE STABLE CONTINUOUS DOUBLE AUCTION MARKET DESIGN FOR GRID RESOURCE
       ALLOCATION ........................................................................................................... 158
    7.1.1 Core Services .................................................................................................. 159
    7.1.2 Integration with a Local Scheduler ............................................................... 160
  7.2 FINAL REMARKS ................................................................................................... 163

BIBLIOGRAPHY ............................................................................................................... 167
List of Tables

Table 1.1: Simple demand and supply functions ................................................................. 29

Table 6.1: Market efficiency of a CDA populated with TT agents in a stable market condition ........................................................................................................................................... 126

Table 6.2: Market efficiency of a CDA populated with TT agents in a changing market condition ........................................................................................................................................... 128

Table 6.3: Market economic efficiency of a CDA populated with ZI agents in the case of generally stable market conditions ........................................................................................................................................... 130

Table 6.4: Market efficiency of a CDA populated with ZI agents in a changing market condition ........................................................................................................................................... 131

Table 6.5: Efficiency comparison of a CDA under one GD user agent (u0) and another nine ZI agents ........................................................................................................................................... 134

Table 6.6: Market efficiency of a SCDA populated with TT agents in a stable market condition ........................................................................................................................................... 139

Table 6.7: Market efficiency of a SCDA populated with TT agents in a changing market condition ........................................................................................................................................... 140

Table 6.8: Market efficiency of a SCDA populated with TT agents in a significantly changing market condition ........................................................................................................................................... 142

Table 6.9: Market efficiency of a SCDA populated with ZI agents in a stable market condition ........................................................................................................................................... 144
Table 6.10: Market efficiency of a SCDA populated with ZI agents in a changing market condition. ........................................................145

Table 6.11: Market efficiency comparison of the CDA with the SCDA.........154
List of Figures

Figure 1.1: The aggregate surplus .................................................................28

Figure 1.2: Illustration of Competitive Equilibrium. .................................30

Figure 1.3: The user-centric performance value associated with achieved allocation time..................................................................................................32

Figure 2.1: The spectrum of market formulations.......................................39

Figure 3.1: The structure of the CDA. ..........................................................67

Figure 4.1: The architecture of the SCDA. ....................................................83

Figure 4.2: The architecture of fuzzy logic inference. ...............................89

Figure 4.3: The membership function of (a_{min} - b_{max}). .........................91

Figure 4.4: The membership function of a_{min}. ........................................92

Figure 4.5: The membership function of the output. ..................................92

Figure 4.6: The degree values of output price............................................97

Figure 5.1: Interactions of software agents in simulations. .........................104

Figure 5.2: The agent decision problem....................................................105

Figure 6.1: The trading process of a CDA populated with TT agents in a stable market condition.................................................................124
Figure 6.2: The trading process of a CDA populated with TT agents in a changing market condition..................................................................................................................................................127

Figure 6.3: The trading process of a CDA populated with ZI agents in a stable market condition.................................................................................................................................................................130

Figure 6.4: The trading process of a CDA populated with ZI agents in a changing market condition..................................................................................................................................................................................................131

Figure 6.5: Efficiency comparison of agents with different bidding strategies in a CDA.............................................................................................................................................................................................................136

Figure 6.6: The trading process of a SCDA populated with TT agents in a stable market condition........................................................................................................................................................................................................139

Figure 6.7: The trading process of a SCDA populated with TT agents in a changing market condition................................................................................................................................................................................................140

Figure 6.8: The trading process of a SCDA populated with TT agents in a significantly changing market................................................................................................................................................................................................142

Figure 6.9: The trading process of a CDA populated with ZI agents in a stable market condition................................................................................................................................................................................................144

Figure 6.10: The trading process of a SCDA populated with ZI agents in a changing market condition................................................................................................................................................................................................145

Figure 6.11: Efficiency comparison of agents with different bidding strategies in the SCDA.............................................................................................................................................................................................................147

Figure 6.12: User-centric performance comparison........................................................................................................................................................................................................150

Figure 6.13: Comparison of user-centric performance between the SCDA and
the CDA under parallel jobs. ................................................................. 152

Figure 7.1: Logical architecture of a typical local scheduler.................... 161

Figure 7.2: The integration of the SCDA market with local schedulers. ....... 162
Abstract

This thesis describes the design, implementation and evaluation of a market based Grid resource allocation mechanism which takes into account the architectural features and special requirements of computational Grids, while ensuring economic efficiency, even when the underlying resources are being used by participants who are self-interested and uncooperative.

The design of the market-based Grid resource allocation mechanism is considered from four salient angles. First, the Grid computing environment is regarded as a distributed two-sided market; competition occurs on both sides of the market, consumers and providers, simultaneously. Second, the Grid resource allocation mechanism needs to provide a flexible and effective scheduling environment; more specifically, it must have the ability to offer resources and resource bundles with minimal delay to high value jobs (immediate allocation and co-allocation). Third, the thesis focuses on a discriminated-price mechanism, as opposed to a market-clearing mechanism (or single price scheme), as the dynamic pricing mechanism. Finally, the resource allocation mechanism is evaluated in terms of both economic efficiency and scheduling efficiency; more specifically, Pareto efficiency and user-centric performance.

In considering the above aspects, and in surveying existing market formulations, the Continuous Double Auction (CDA) is regarded as the most appropriate existing market model for Grid resource allocation. The CDA is simple and yet is able to achieve high market efficiency, with only a small amount of information passing to participants and a low computational cost. Furthermore, it offers continuous matching and clearance, which makes it flexible and fulfils
the requirement for immediate allocation. However, the basic form of the CDA has unnecessarily high volatility, which causes dissatisfaction among participants and difficulty in co-allocation.

An innovative Stable Continuous Double Auction (SCDA) has been developed in order to overcome these problems while maintaining the other beneficial features of the CDA. A price adjustment mechanism is implemented for the SCDA in order to reduce unnecessary price volatility caused by impatient and/or insensitive behaviour of market participants. Experimental results show that the SCDA is superior to the CDA in terms of both economic efficiency and scheduling efficiency. The SCDA has features of continuous matching and low communication/computation cost, allied with low price volatility and low bidding complexity. In particular, the immediate allocation and stable price features offered by the SCDA ease the problem of co-allocation.

Market-based Grid resource allocation using the SCDA is flexible. It can work easily alongside other resource allocation mechanisms, allowing incremental evolution of the Grid resource market. It allows resources or resource bundles to be allocated at any time and at fair and stable prices. Effective market-based Grid resource allocation is thus shown to be feasible.
Declaration

that no portion of the work referred to in the thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institute of learning.
Copyright

Copyright in text of this thesis rests with the author. Copies (by any process) either in full, or of extracts, may be made only in accordance with instructions given by the author and lodged in the John Rylands University Library of Manchester. Details may be obtained from the Librarian. This page must form part of any such copies made. Further copies (by any process) of copies made in accordance with such instructions may not be made without the permission (in writing) of the author.

The ownership of any intellectual property rights which may be described in this thesis is vested in The University of Manchester, subject to any prior agreement to the contrary, and may not be made available for use by third parties without the written permission of the University, which will prescribe the terms and conditions of any such agreement.

Further information on the conditions under which disclosures and exploitation may take place is available from the Head of School of Computer Science.
Acknowledgements

I am deeply indebted to my supervisor, Professor John R Gurd. His encouragement, support and advice have been immensely valuable, both in personal and professional terms. I am particularly grateful for his emphasis on simplicity and elegance in research, and for his guidance on building confidence to pursue a novel endeavour. Without him this thesis would not have been possible.

Thanks also to all the members of the Centre of Novel Computing. In particular, Dr. Ken Mayes, Dr. Len Freeman, Dr. Mikel Lujan, and Dr. Anna Manning for their help during the years of my PhD. Thanks also go to Dr Stephen M Pickles, Dr John Brooke, Prof. John Darlington, Dr. Jon MacLaren, Dr. William Lee and John Ainsworth for their help and constructive comments.

Special thanks go to my PhD viva examiners, Professor David S. Brée and Professor Paul Watson. Each devoted significant time and effort to my thesis, and their suggestions and comments led to substantial improvement in the final product.

I have also benefited from time spent with friends at Manchester. I would especially like to thank Qiwei Zhu, Bin Cheng, Na Li, Keping Chen, Henan Zhao, James Mao, Hui Zhang, Ting Xu, Yibo Fan, Yuhuan Zhu, Min Yang, and Xiaolu Liu for their supports.

This work was supported by the OST-funded MCS project (THBB/C/008/00020 – ‘A Market for Computational Services’) and the EPSRC-funded RealityGrid project (GR/R67699). I would like to, therefore, extend my gratitude to OST and EPSRC for supporting this work.
To my wife, Shanshan, and our daughter, Ping.

They make every day a joy.
1 Introduction

This thesis addresses a problem at the nexus of computer science and microeconomics: how to efficiently allocate distributed computational Grid resources located in different management domains among competing parties. Evidently, constraints are imposed on the resource allocation mechanism by the inherent architecture of the computational Grid. But these constraints are counterbalanced by the need to allocate Grid resources in an economically efficient manner, even when the computational Grid is being both provided and used by participants who are self-interested and mostly uncooperative.

The aim of the work described in this thesis is to design, implement and evaluate a market based Grid resource allocation mechanism which takes into account the architectural features and special requirements of computational Grids, while ensuring economic efficiency. The design considers four salient aspects, namely, the computational Grid environment as a two-sided market, the requirement to provide a flexible and reliable scheduling environment, dynamic pricing mechanisms and evaluations of efficiency, as described in the next four sections.
1.1 Computational Grid Resource Allocation

The computational Grid is a rapidly emerging and expanding technology that enables users (consumers) to utilize distributed computing resources across multiple administrative boundaries in a transparent manner (Foster, Kesselman et al. 2001; Foster and Kesselman 2003). Grid computing is able to fulfill a range of application needs that could not be achieved on any one resource or in a single organization. Other benefits offered by Grid computing include efficient utilization of IT infrastructures and resources, cost reduction, and higher scalability (Foster and Kesselman 2003). The problem of efficiently allocating distributed resources located in different management domains among competing parties is an important open research topic (Buyya 2002; Berman, Fox et al. 2003; Nabrzyski, Schopf et al. 2003).

Grid resources are not subject to centralized control (Foster 2002). Grid resources are typically controlled within self-interested local administrative domains and are shared amongst competing users. Conflicts of interest between users and providers are inevitable. In this environment, both users and providers need to express their preferences, and conflicts of interest between competing parties need to be reconciled. Furthermore, more complex requirements may be imposed on Grid resource allocation. Multiple self-interested parties can simultaneously provide and consume sets of resources from different management domains. Users can demand large sets of disparately controlled resources. This creates a combinatorial distributed resource allocation challenge that cannot easily, if at all, be solved by users dealing with providers on a one-to-one basis. A carefully designed Grid resource allocation mechanism is required in order to solve these challenges.

Market-based resource allocation addresses the above challenges more naturally than the common alternatives, such as first-come-first-served
allocation, reservation mechanisms or priority queues. Market-based allocation mechanisms are attractive for Grid resource allocation for the following reasons:

- Markets eliminate the need for a central control point and suit the decentralized nature of a computational Grid; the decision-making process is distributed across all users and providers. Once an exchange protocol and associated trading rules have been established, self-interested participants can make effective decisions using their local information.

- Markets constitute a competitive environment that naturally balances the conflicts of interest between parties. High prices generate an incentive for providers to offer their resources but may cause users to retreat from using them. Low prices attract users but may bankrupt providers. A market encourages participants to use resources wisely and tries to encourage overall usage decisions that maximize overall value.

- Markets can provide an environment to facilitate complex combinatorial resource requests. Users can freely acquire resources in a market at the time they want, provided they pay sufficiently high prices for these resources. Recently, especially in the last decade, market formulations have been designed and applied to combinatorial resource allocation problems, such as energy markets, wireless spectrum auctions, and airport landing and takeoff scheduling (Rassenti, Smith et al. 1982; Pekec and Rothkopf 2003; Milgrom 2004). For example, combinational auctions allow bidders to place bids on combinations of items, called bundles, rather than just on individual items (Vries and Vohra 2003).

The Grid resource allocation environment is modelled as a two-sided market in this thesis. In practice, competition typically occurs simultaneously on both
sides of the market in larger scale systems. In computational Grids, users may be competing for resources while providers compete with each other to capture market share. Moreover, the workloads on computational Grids have demonstrated high fluctuation (Wolski, Plank et al. 2001; Young et al. Dec. 2004). Resource supply may vary due to computer failures or other reasons. This means that there is elasticity of both supply and demand in computational Grids, which further supports the idea of treating the Grid resource allocation environment as a two-sided market.

1.2 Grid Scheduling Requirements

Co-allocation, that is the simultaneous allocation of multiple resources (also referred to as co-scheduling), is necessary for effective use of computational Grids. The NSF Teragrid project has noted a growing demand amongst users for co-scheduling of resources and is currently working on its provision (Catlett, Bair et al. 2005). Similarly the European DEISA high performance Grid has "… decided to implement a co-scheduling service on the supercomputing grid, to enable grid applications that run concurrently on different platforms." (www.deisa.org).

The need for co-allocation can be traced to two, related, emerging major computational Grid paradigms, namely, computational steering for parameter search “experiments” and simulations consisting of loosely-coupled models.

The aim of parameter search is to find regions in the parameter space of a simulation which are associated with "interesting" scientific phenomena. The classical approach to locating such regions is by running large numbers of simulations concurrently. However, it has been noted that many task-farm
simulations waste CPU time and disk space because they execute in uninteresting regions of the parameter space. Further, the time required to analyse the large amount of data that is generated is excessive (Chin, Harting et al. 2003). The idea behind computational steering is to permit user-intervention in order to steer the execution of a small number of concurrent simulations so as to interactively guide the simulations to areas of interest (Brooke, Coveney et al. 2003). However, this steering (and accompanying visualization) requires that Grid resources are "... co-allocated so that the scientist knows when his/her simulations are running... " (Venturoli, Harvey et al. 2005).

Coupled model applications are an important new class of applications for Grids, for example sophisticated multi-scale, multi-physics applications. Previously independent software components are coupled so that they interact together by exchanging data at run-time (Armstrong, Ford et al. 2005). All software components have to run simultaneously so that they can be coupled. Such applications "... are at the forefront of modern software engineering ..." and require co-scheduling facilities (www.deisa.org).

It can be argued that such coupled-model applications belong to the more general paradigm of workflow applications (the dominant kind of Grid application), which typically comprise multiple components that run in different locations, either concurrently or with various sequential dependency relationships. Co-allocation is required when the components are to be executed concurrently.

While the necessity of co-allocation is well recognized, another equally important requirement, namely, immediate allocation, is often overlooked by designers. Immediate allocation requires resources to be allocated with minimal delay from the time of request. This capability is fundamental for the flexibility of the Grid. Users need access to distributed resources at a time determined by
them. Without immediate allocation, users must predict their resource needs in advance. Users may not be able accurately to predict what time a job will start and finish, especially in a highly dynamic Grid computing environment. Failure may occur and demand may change over time. Immediate allocation is therefore critical for failure recovery and on-demand provision of resource. As co-allocation entails providing the ability to obtain multiple resources with minimal delay, the flexibility offered by immediate allocation helps with co-allocation.

In summary, Grid resource allocation needs to support co-allocation of resources and to be able to allocate resources with minimal delay, i.e. immediate allocation, in order to provide a practical environment on which Grid applications can run.

Many Grids currently utilize the traditional batch queue approach. Batch queue systems (typically based on first-come-first-served allocation) are widely used by supercomputers and clusters. The batch queue is regarded as necessary because it prevents tasks from saturating a resource, while maintaining a high level of utilization. However, the time a task waits in the queue is uncertain, i.e. there is no guarantee when tasks will be executed. This causes problems where dependencies exist between components of a distributed application (Sulistio and Buyya 2004). Clearly, a batch queue system will cause problems for both co-allocation and immediate allocation. Queues with different priority have been used to ease the tension of uncertain waiting time. However, these priority-based schemes beg the question of how to set priorities so that desirable results follow.

One possible solution to the co-allocation problem is provided by Advance Reservation (AR) techniques. However, AR is static and is non-trivial to apply for both providers and users. Users may not be able to give accurate prediction
of both starting time and ending time, which is essential for the proper functioning of an AR system (Smith, Taylor et al. 1999; Maclaren, Sakellariou et al. 2004). In the case of scheduling multiple components, a failure to meet the reservation time slot for one component will cause rescheduling of all tasks having a subsequent dependency relationship with it. Applications with dependencies between components provided by different resource providers can present particular difficulties for AR; the required coordination between the participating local schedulers (AR systems) necessitates a delicate balancing of the conflicts of interest between providers and users.

In order to achieve this, dynamic pricing has been explored. Prices move up if demand exceeds supply, and down otherwise. Prices thus converge to a point where demand matches supply. Therefore, high priority jobs, i.e. those that value resource(s) the most, are able to acquire resource(s) with minimal delay. The resulting Grids are able to offer immediate-allocation and co-allocation (obtaining multiple resources with minimal delay) to high priority jobs. Mechanisms for achieving dynamic pricing are discussed next.

1.3 Dynamic Pricing Mechanisms

Having modelled Grid allocation environments as two-sided markets, and having indicated the need for co-allocation and immediate allocation, dynamic pricing mechanisms are now considered. This thesis focuses on discriminated price mechanisms, in which the prices of resources typically vary for every transaction. Continuous matching and clearance can therefore be offered. Market efficiency in these schemes is achieved through the process of price convergence.
The reasons for using discriminated price mechanisms, as opposed to *market-clearing* mechanisms, which set a single price for all transactions in a certain period, are as follows:

- Firstly, the Grid resource allocation mechanism needs to support both co-allocation and immediate-allocation. Market-clearing mechanisms choose a single price so that demand equals supply in order to “clear the market”. Clearing time frames are therefore required for all transactions in market-clearing mechanisms, which conflicts with the immediate-allocation requirement. Moreover, there is no easy answer to the question of what the length of the clearing time frames should be in order to generate desirable results. If the length is set too short, the generated clearing price does not reflect that the overall demand equals supply, leading to an inefficient solution. When a long length is set, immediate allocation is increasingly difficult to achieve. In contrast, there is no general time frame constraint on discriminated price mechanisms. Transactions can happen at any time. The continuous matching and clearance feature suits the requirement for immediate allocation.

- The second motivation for discriminated price mechanisms comes from the consideration of setting up and starting the market. The computational Grid is still in its infancy and Grid environments and Grid applications are evolving rapidly. Their economic features therefore change quickly. This leads to difficulty in defining and observing parameters of the market-clearing mechanism. For example, the length of the clearing time frame and the type of resources offered in the market need to be predefined before the market starts. In contrast, discriminated price mechanisms are easier to start. There is no constraint on the general clearing time frame and there are few constraints on introducing new types of resource into the market. A discriminated price mechanism also
has the advantage of facilitating a large volume of trade. The Continuous Double Auction (a discriminated price mechanism) has replaced the Clearing House Auction (a market-clearing mechanism) as the preferred trading mechanism in stock markets in order to cope with a dynamic trading environment and to facilitate a large volume of trade (Friedman and Rust 1993).

It is important to note that a greater degree of sophistication in the response of the market participants is expected using a discriminated price mechanism. Users are expected to implement sophisticated mechanisms in order to cope with discriminated and highly dynamic prices. Moreover, discriminated price mechanisms require the responses of participants on a fast time scale, which is also a requirement for immediate allocation. These factors cause high complexity of participation and potentially high dissatisfaction for users, especially for small users who do not have the resources to cope with this high complexity. Predominantly automatic trading is therefore required in order to achieve the required speed and to isolate users from the underlying complexity. The required automatic trading can be realized through software agents, i.e. autonomous, problem-solving entities capable of effective operation in a dynamic and open environment.

The complexity of participating in the resource allocation market has to be reduced in order to provide a stable allocation environment for agents. It does not seem feasible to achieve co-allocation and immediate-allocation effectively in an unstable environment. This is the prime reason for the design of a new auction for Grid resource allocation in this thesis. The designed auction is intended to produce a stable allocation environment, reduce the complexity of participation and promote market efficiency, while keeping the continuous matching and clearance feature. Chapter 2 and Chapter 3 will return to this topic in more detail.
1.4 Efficiency

This section defines what constitutes an efficient allocation. Considering the Grid resource allocation as a two-sided market, the goal of distributed resource allocation is no longer to maximize utilization. Instead, two types of efficiency are defined for Grid resource allocation in this thesis, namely *economic efficiency* and *scheduling efficiency*. These two definitions of efficiency guide the design of the resource allocation market and the selection of benchmarks for the simulations that are used later for evaluation.

1.4.1 Pareto Efficiency and Aggregate Surplus

The resource allocation problem considered in this thesis consists of two active types of market participants: *consumer* (user) and *provider*. Consumers demand allocations of resources and providers supply resources. Each consumer $i$ is defined by a *utility function* $U_i(R_i)$ which defines the monetary value to consumer $i$ of obtaining $R_i$ units of resource, where $R_i > 0$. Similarly, each provider $n$ is defined by a *cost function* $C_n(R_n)$, which defines the monetary value to provider $n$ of supplying $R_n$ units of resource, where $R_n > 0$.

A key assumption is that both utility and cost are measured in monetary units. This assumption implies that there are actually two types of goods in the resource allocation scenarios: the first is the computation resource under consideration, and the second is the money paid to secure the resource. Suppose then that a consumer with utility function $U_i$ receives a resource allocation $r$, for which they make a payment $p$. Then the net *payoff* to this consumer is:
On the other hand, suppose that a provider with cost function $C_n$ supplies a resource $r$, for which they receive revenue $p$. Then the net payoff to this provider is:

$$U_i(r) - p.$$

This thesis adopts the well known concept of *Pareto efficiency* as its notion of economic efficiency: an allocation is Pareto efficient if the benefit to one market participant cannot be strictly increased without simultaneously strictly decreasing the benefit to another player. In other words, if there is some way to make some market participant(s) better off without harming others, then it should be done. An allocation that cannot be improved in this way is Pareto efficient. The Pareto optimal solution is the justification for the intuitive notion of the desirability of competition (Vickrey 1961; Friedman and Rust 1993). A Pareto optimal solution encourages competition since participants make active moves (revealing their true preferences), rather than shade their true preferences in order to move closer to market consensus.

As long as arbitrary monetary transfers between market participants are allowed, the benefits to participants are their net payoffs. Therefore, any Pareto optimal allocation must maximize aggregate utility less aggregate cost. This quantity is know as the *aggregate surplus* (Mas-Colell, Whinsten et al. 1995). Hence, the notion of maximization of aggregate surplus is treated as a Pareto efficient allocation rule (Marshall 1920).
In order to illustrate the notion of maximized aggregate surplus, take the case of a single resource for simplicity. As shown in Figure 1.1, each consumer $i$ chooses a demand function, $D_i(p)$, which describes their demand for the resource as a function of the price $p$ of that resource. Analogously, each provider $n$ chooses a supply function, $S_n(p)$, which describes the quantity the provider is willing to supply as a function of the price of the resource. The demand functions from all consumers construct an aggregate demand curve $AD(p) = \sum_i D_i(p)$. Similarly, the aggregate supply curve is $AS(p) = \sum_n S_n(p)$. The price $p'$ is the point at which aggregate demand equals aggregate supply, i.e. $AD(p') = AS(p')$. The price $p'$ is also called the Competitive Equilibrium (CE).

In the case that consumers and providers truthfully derive their demand and supply functions from their utility and cost functions, a rough measurement of the consumers’ surplus from this resource allocation case is the area under the demand curve and above the price $p'$, as shown in Figure 1.1. Similarly, a
rough measurement of the providers’ surplus is the area above the supply curve and below the price \( p' \). Clearly, the aggregate surplus, the sum of consumers’ and providers’ surplus, is maximized when trading at the CE, \( p' \), i.e. the Competitive Equilibrium yields a Pareto efficient allocation. This result is actually the first fundamental theorem of welfare economics (Mas-Colell, Whinston et al. 1995). The second fundamental theorem is that, under sufficient assumptions, any Pareto efficient allocation can be achieved as a competitive equilibrium. These two fundamental theorems are the cornerstone results of general equilibrium theory (Arrow and Hahn 1971). Competitive equilibrium is also referred to as Walrasian equilibrium as the notion was firstly developed by Walras.

Taking a simple example, Table 1.1 shows some possible demand and supply functions for a single type of resource.

<table>
<thead>
<tr>
<th>Demand/supply Functions</th>
<th>Supplies</th>
<th>Demands</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>offers 1 unit at price 5</td>
<td>consumes 1 unit at price 30</td>
</tr>
<tr>
<td></td>
<td>offers 1 unit at price 15</td>
<td>consumes 1 unit at price 20</td>
</tr>
<tr>
<td></td>
<td>offers 1 unit at price 25</td>
<td>consumes 1 unit at price 10</td>
</tr>
</tbody>
</table>

Table 1.1: *Simple demand and supply functions.*
The corresponding supply and demand curves are shown in Figure 1.2. The CE in this example is the interval from 15 to 20 (when demand and/or supply functions have vertical or horizontal sections then there may be an interval of equilibrium prices or quantities, rather than a single competitive equilibrium price and quantity). If the CE is chosen to be 17.5, then the maximized aggregate surplus (full efficiency) is:

\[(30-17.5) \times 1 + (20-17.5) \times 1 + (17.5 - 15) \times 1 + (17.5 - 5) \times 1 = 30.\]

The theory of general equilibrium provides a first justification for the attractiveness of market mechanisms. Assuming that no market participants anticipate the effects of their actions on the final market price, i.e. they truthfully reveal their utility function, market mechanisms provide a simple and decentralized method to ensure efficient allocation of resources. But the assumption of limited price anticipation is quite a strong one, particularly if only a few market participants compete at any given time.

As a discriminated price mechanism has been chosen, the number of participants at any instant is likely to be small, i.e. only a few participants
compete at a given time. Therefore, the decision of market participants will affect the final price. Hence, the competition between market participants is more like a game: the payoff of a given player is directly expressed as a function of his own strategy, as well the strategy of all other players. This aspect has to be taken into consideration when designing the Grid resource market. This game theory aspect will be discussed further in Section 2.2.3.

It is important to note that the above definition of Pareto efficiency and Competitive Equilibrium is a rather static definition. In practice, the supply and demand curves are time-variant, as are the Competitive Equilibria.

### 1.4.2 Scheduling Efficiency

Pareto efficiency only considers the monetary value delivered; the scheduling quality is not taken into consideration. Users’ satisfaction levels are correlated with the speed with which their high priority jobs are executed, and this will also be represented in the measurement of efficiency in this thesis. Hence, the time taken to allocate resources needs to be included in the utility function when considering the overall efficiency of any Grid resource allocation mechanism.

The *user-centric performance metric* is used to measure scheduling efficiency in this thesis. The user-centric performance metric was developed for performance analysis of market-based batch schedulers for clusters of workstations (Chun and Culler 2002). The metric is modified slightly in this thesis in order to use it for evaluation of the market-based allocation mechanism.

In the user-centric performance metric, each user $i$ is modelled as having a utility function for each job. This utility function measures the value that the
job delivers to the user as a function of its allocation time as follows

$$V_i(R_i) = U_i(R_i) \times TV_{R_i},$$

where $V_i(R_i)$ defines the value to consumer $i$ of obtaining $R_i$ units of resource for the job, $U_i(R_i)$ is the valuation of the allocated resource to the consumer (i.e. the maximum valuation the consumer would pay for the resource) and $TV_{R_i}$ is the value to the consumer of the achieved allocation time, derived from the value function against allocation time that is shown in Figure 1.3. The latter is a function of the latest allocation time of all resources $R_i$ allocated for a job; the value decays linearly over time as a function of slowdown. The user-centric performance metric uses aggregate utility as a measure of the overall value delivered to users, as follows

$$\sum_i V_i(R_i).$$

![Figure 1.3: The user-centric performance value associated with achieved allocation time. The value $TV$ decreases linearly from 1, where resource is allocated immediately, to 0, where the maximum acceptable delay tolerance of the user is reached.](image-url)
1.4.3 Overall Efficiency

Economic efficiency is the primary efficiency in this thesis. The scheduling efficiency is secondary to the Pareto efficiency. When the market is close to Pareto efficiency and is stable, scheduling efficiency follows. Nevertheless, scheduling efficiency is an important performance metric for the allocation mechanism and could measure the overall value delivered to users. Through the remainder of the thesis, the term “efficient allocation” is used to refer to a Pareto efficient allocation.

1.5 Contributions of This Thesis

In considering the above aspects, and in surveying existing market formulations, the Continuous Double Auction (CDA) is regarded as the most appropriate existing market model for Grid resource allocation. The CDA is simple and yet is able to achieve high market efficiency, with only a small amount of information passing to participants and a low computational cost. Furthermore, it offers continuous matching and clearance, which makes it flexible and fulfils the requirement for immediate allocation. However, the basic form of the CDA has unnecessarily high volatility, which causes dissatisfaction among participants and difficulty in co-allocation.

An innovative Stable Continuous Double Auction (SCDA) has been developed in order to overcome these problems while maintaining the other beneficial features of the CDA. A price adjustment mechanism is implemented for the SCDA in order to reduce unnecessary price volatility caused by impatient
and/or insensitive behaviour of market participants. Experimental results, presented in Chapter 6, show that the SCDA is superior to the CDA in terms of both economic efficiency and scheduling efficiency. The SCDA has features of continuous matching and low communication/computation cost, allied with low price volatility and low bidding complexity. In particular, the immediate allocation and stable price features offered by the SCDA ease the problem of co-allocation.

Market-based Grid resource allocation using the SCDA is flexible. It can work easily alongside other resource allocation mechanisms, allowing incremental evolution of the Grid resource market. It allows resources or resource bundles to be allocated at any time, and at fair and stable prices. Effective market-based Grid resource allocation is thus shown to be feasible.

The remainder of the thesis is organized as follows. Chapter 2 reviews the entire spectrum of market formulations (the process of determining trade prices). The strengths and weaknesses of each formulation are discussed. Special attention is given to whether each market formulation is appropriate for delivering market-based Grid resource allocation. Two general approaches for achieving efficient allocation are identified in this chapter, namely, incentive compatible mechanism design and iterative price-convergence. The Continuous Double Auction (CDA) is regarded as the most appropriate existing market model for Grid resource allocation in this thesis. The CDA is simple and yet is able to achieve high market efficiency with only a small amount of information passing to participants and a low computational cost. Furthermore, it offers continuous matching and clearance, which makes it flexible and fulfils the requirement for immediate allocation. However, the CDA has unnecessarily high volatility, which causes dissatisfaction among participants and difficulty in co-allocation. This leads to the motivation for the design of the Stable Continuous Double Auction (SCDA), to overcome this undesirable high
volatility while maintaining the other beneficial features of the CDA.

In Chapter 3, the CDA is analyzed in order to provide a foundation for designing the SCDA. Since the analysis of an auction is essentially an analysis of the strategic interactions between participants, and in the current absence of robust analytical tools, the analysis of the CDA is carried out by means of reviews of the prominent bidding strategies developed for the CDA and associated experiments. This chapter concludes with observations about these reviews and leads to the design of the SCDA.

Chapter 4 presents the synthesis of the SCDA. The SCDA mainly aims to reduce the unnecessary price volatility caused by impatient and/or insensitive behaviour of some market participants. A novel fuzzy logic and heuristic based price adjustment mechanism is implemented to achieve the above goal. The architecture of the SCDA and its price adjustment mechanism is also described in detail in Chapter 4.

Chapter 5 presents the experimental environment for evaluating the SCDA. The decision problem for a participant in the CDA is presented as the main justification for the settings of the experimental environment. The Zero Intelligence trader is chosen as benchmark trader for the experiments as it has shown to closely mimic traders in a real stock market. In order to observe the response of the auction to dynamic changes in overall market supply and demand, each experiment is set to be a continuous process with changing Competitive Equilibrium. Resource requests and offers are generated gradually. This is different to the experiments described in Chapter 3, in which experiments are conducted by individual games with a static CE. The notion of the time interval is introduced in order to calculate market efficiency. A step-by-step description of how to construct each experiment is given at the end of the chapter.
Chapter 6 evaluates the SCDA against the CDA in terms of economic efficiency and scheduling efficiency. The experimental results show clearly that the SCDA is superior to the CDA in both economic efficiency and scheduling efficiency.

In Chapter 7, an outline of a SCDA market designed for Grid resource allocation is firstly given, in order to further demonstrate how the SCDA can be used for the Grid environment and, secondly, to indicate future research directions. The designed SCDA market allows seamless integration of current local schedulers and facilitates incremental evolution of the Grid resource market. The thesis concludes with some final remarks.
2 Market Formulations

This chapter reviews the possible market formulations for Grid resource allocation. Market formulations are processes for determining trade prices, i.e. how the offers and other messages can be exchanged in the market to determine a trade. Market formulations are at the heart of market-based allocation mechanisms. The choice of market formulation can significantly affect the market efficiency, the complexity (in terms of both participating and implementation) and the way in which resources can be scheduled in Grid computing environments.

The market formulations studied in this thesis all involve a third party, apart from the consumer and the provider. The third party gathers information (preferences of consumers and providers) and sets transaction prices for the trades between providers and consumers. For example, in the case of an auction market, the third party is the auctioneer. With a third party, the consumers can avoid the tedious process of going through all providers to find the best deal. More importantly, the transaction prices are jointly determined on the basis of the preferences of both consumers and providers. Hence, the price is set by the market, rather than any single participant. Only by introducing a third party can the overall supply and demand be reflected. Individual participants have only partial information about the marketplace and therefore a participant can not
make a decision reflecting overall supply and demand on their own.

The remainder of the chapter is organized as follows. The chapter starts with an illustration of the spectrum of market formulations (Section 2.1), with single-sided auctions at one end and commodity market models at the other. Different market formulations are then reviewed. The main design concerns for the markets are also introduced. In Section 2.2, four major types of single-sided auction are studied. Through the study of single-sided auctions, especially the design of the seminal Vickrey auction, two main auction design principles, namely, openness and strategy-proofness, are introduced. The notion of strategy-proofness leads to a research field, mechanism design, which is introduced in Section 2.3. The novel Vickery-Clarke-Groves mechanism, which offers a strategy-proof solution to the complex combinatorial resource allocation problem, is also introduced in Section 2.3. Section 2.4 presents double-sided auctions; these are divided into discrete-time, continuous-time and hybrid types. The Continuous Double Auction (CDA) stands out because it offers continuous clearance and efficient market outcome under a wide range of conditions. In Section 2.5, combinational auctions are presented; the advantages, complexity and computational cost of combinational auctions are discussed. Section 2.6 introduces the Commodity market model and its main disadvantage, namely that it is not incentive compatible. Section 2.7 then summarises up two general approaches of achieving efficient allocation: incentive compatible mechanism design and iterative price-convergence. The section also explains why the CDA is considered to be the most appropriate existing market formulation for Grid resource allocation among all the different market formulations. Finally, the main undesirable feature of applying the CDA to Grid resource allocation, namely high price volatility, is identified. This provides the motivation for designing a more stable form of CDA.
2.1 The Spectrum of Market Formulations

The most prominent market formulations can be viewed as covering a spectrum, with the two ends being defined by the commodity market model and the single-sided auction (as shown in Figure 2.1).

![Figure 2.1: The spectrum of market formulations. The distances between different formulations are indications of the trend, rather than implying scale.](image)

The commodity market model is based around centralized information and strives to generate a price that ensures the equilibrium of supply and demand. The same, or similar, resources are traded identically in a commodity market. A consumer does not purchase a “specific” resource, but rather takes one of many equivalents. At the other end of the spectrum, the single-sided auction is more dynamic and flexible, as negotiation can be done for every individual resource, but at the expense of efficiency. Resources are allocated on an individual basis and single-sided auctions are easy to implement and straightforward to participant in. Different market formulations in the spectrum will be discussed...
in the reminder of the chapter. Their suitability for being applied to Grid resource allocation will also be considered.

2.2 Single-sided Auctions

Single-sided auctions are the traditional and most cited forms of *auction*: an institution with an explicit set of rules determines resource allocation and prices on the basis of bids from the market participants (McAfee and McMillan 1987). As its name suggests, a single-sided auction is a mechanism for one-to-many price negotiation, i.e. the competition comes only from one side (either consumers or providers). William Vickrey established the basic taxonomy of single-sided auctions based upon the order in which prices are quoted and the manner in which bids are tendered. He established four major types of single-sided auction (Vickrey 1961), as described below.

2.2.1 Four Major Single-Sided Auction Types

The four major types of single-sided auction are:

- **English Auction** (also called Ascending Bid Auction). The auctioneer begins with the lowest acceptable price. All bidders are free to increase their bids so as to exceed other bids. When no one will increase the bid, the auction ends, and the highest bidder wins the item at the final price (Milgrom 1989). A computerized English auction normally specifies a hard deadline for implementation reasons. The auction is an open auction and can generate an efficient outcome based on iteration of the price signal and the bidder’s response. The main problem with the English auction is the “winner’s curse” – a bidding war among bidders results in
the bidder who wins the auction being the bidder who has most likely
over-valued the resource.

- **Dutch Auction** (Descending Bid Auction). The auctioneer starts with an
extremely high price and lowers the price progressively until a buyer
claims the resource at that price. Winners are normally allowed to
purchase as many resources as they can in the case of multiple-unit
resource auctions, and the process will continue until all the resources are
allocated. This makes the Dutch auction a good choice for clearance or
fast trading. The Netherlands flower market adopts this auction in order to
achieve fast trading because flowers fade quickly.

- **First Price Sealed Auction** (FPSA). A deadline is set up for the auction.
Each bidder independently submits one bid without knowing any other
bids (bids are sealed). The highest bidder wins the item at the price of his
bid at the end of the auction.

- **Vickrey Auction** (Second Price Sealed Auction). As in a first price sealed
auction, bids are submitted once and each bidder is ignorant of the other
bids. However, the item is awarded to the highest bidder at a price equal
to the second-highest bid (i.e. the highest unsuccessful bid).

The four auctions can be catalogued into two main auction types: open auctions
and sealed auctions. The English auction and the Dutch auction are open
auctions; the FPSA and the Vickrey auction are sealed auctions. Sealed auctions
generally require more trust toward the auctioneer than open auctions. In
particular, in the case of a Vickery auction, bidders must trust the auctioneer not
to submit a false bid, since an unscrupulous auctioneer can insert a bid just
under the highest bid in order to increase his profit without any risk. On the
other hand, an open auction is more transparent; although the auctioneer can
still use a shill to increase the bid price, there is a risk that the shill bids too
high and becomes the winner in the auction, i.e. the loser!

Furthermore, price has to be the only factor in determining the winner in open auctions. In contrast, sealed auctions are able to cope with multi-dimensional problems within the auction. For example, in the FPSA, the winner can be determined on multiple factors, rather than solely on price.

### 2.2.2 Strategy-proofness

Although single-sided auctions seem simple, they can actually provide deep insight about market formulations. For example, an interesting point emerges when the Vickrey auction is compared with the FPSA. It seems intuitively obvious that a provider would make more money by using a first-price auction. However, in practice, this has been shown to be untrue. Bidders fully understand the rules and modify their bidding strategy accordingly. Although it is difficult to specify a single optimal strategy in a FPSA, because a profit-maximizing bid depends upon the actions of others, most bidders generally attempt to shade their bids in order to move closer to the market consensus. On the other hand, bids are adjusted upward in the case of a Vickrey auction when participants fully understand the rules. Players’ bid determines the range of prices that they will accept, but not the actual price. The price that players pay is completely independent of their bid price and, even if they know the second-highest bid, they can still bid their true value because the player only pays the second-highest bid price. As a consequence, no one is deterred out of fear that they will pay too high a price in a Vickrey auction. Aggressive bidders receive sure and certain awards but pay a price closer to the market consensus. The logic of bidding in a Vickrey auction can be described as follows:
Consumer $i$ has valuation $V_i$.

Should $i$ bid $V_i$?

- If bid is $> V_i$: Payoff is possibly worse if $i$ wins, and zero otherwise
- If bid is $< V_i$: Payoff is the same if $i$ wins, but possibly worse otherwise

Therefore, a truth-telling strategy (a participant submits its bid truthfully) is the dominant-strategy, an important concept in non-cooperative game theory, in a Vickrey auction. A strategy is dominant if, regardless of what any other players do, the strategy earns a player a larger payoff than any other. Hence, a strategy is dominant if it is always better than any other strategy, regardless of what opponents may do (Milgrom 1989).

When an allocation mechanism’s dominant strategy is the truth-telling strategy, the allocation mechanism is regarded as strategy-proof, or as incentive-compatible. In a strategy-proof mechanism, every participant’s best strategy (i.e. that which maximises their expected return) in equilibrium with every other agent is to report its true preferences, i.e. participants have an incentive to reveal their preferences truthfully. Strategy-proofness is also a useful computational property; participants can compute their optimal strategy without modelling the preferences and strategies of other participants.

The design of the Vickery auction also demonstrates a crucial aspect about auction design: An auction design needs to anticipate how autonomous, self-interested bidders will react to its rules.

Furthermore, an important reservation emerges when the Vickery auction is compared with the English auction. These two auctions are found to be strategically equivalent, i.e. they produce the same amount of expected revenue
for the auctioneer and they also have the property that an identical bidder would follow the same strategy in both auctions, despite that they are two totally different forms of auction. The English auction relies on gradual revelation of information - via the iterative bidding. On the other hand, the Vickery auction is a sealed-bid auction and depends on strategy-proofness. As it is a sealed auction, the Vickery auction points to a possible allocation solution that is able to cope with multi-dimensional problems and achieve overall economic efficiency when participants consider only their own preferences. Together with advances in Game Theory, this potential leads to a new research field: mechanism design, which will be introduced in detail in Section 2.3.

It is important to note that the Vickery auction has problems in the context of repeated auctions. Whenever a bidder reveals his true value of a resource, the revealed true value can then be used against the bidder in the future. Participants in repeated auctions are therefore reluctant to employ a truth-telling strategy.

2.2.3 Applying Single-sided Auctions to Grid Resource Allocation

Single-sided auctions are straightforward and can be easily implemented. They offer flexibility and decentralization. However, they are not likely to be suitable for Grid resource allocation due to the following reasons:

- First, it is difficult to fulfil co-allocation requests in a single-sided auction market. A co-allocation resource request must place simultaneous bids in multiple auctions, and may only be successful in a few of these. To achieve their co-allocation goal, users have to spend on the acquired
resources while waiting to obtain the others. This spending is wasteful, and the uncertain nature of the auction process may lead to inefficiency for both users and providers.

- Secondly, a computerized single auction is typically set with a hard deadline of bidding due to implementation reasons. Potentially, users are not able to obtain a resource when it is needed as there may be no auctions finishing by the time they require the resources. This causes problems with immediate-allocation requests. The deadline issue will add more complexity to both the application scheduling decision-making and the offering of resources.

- Finally, as observed in (Wolski, Brevik et al. 2003), – “the auction market shows considerably more variance than prices set by the commodity market. The spikes in workload are not reflected in the price, and the variance seems to increase (the price becomes less stable) over time” (The auction market implies a single-sided auction market in the cited paper). Hence, the single-sided auction market fails to generate “fair” and “stable” prices that reflect overall supply and demand, thereby leading to inefficient resource allocation. As already discussed, this thesis treats the Grid resource environment as a two-sided market. Single-sided auction markets contradict this assumption and, not surprisingly, lead to inefficient allocation as they fail to generate enough competition.

2.3 Mechanism Design

2.3.1 Introduction

Mechanism design is a subfield of microeconomics and game theory. It seeks
the design of games to achieve prospective outcome; typically, the theory is applied to achieve a Pareto efficient outcome in environments with multiple competing participants. Trading rules are treated as variables to be optimized in the mechanism design (Mas-Colell, Whinston et al. 1995). The optimization problem is solved on the basis of predicting participants’ behaviour using game theory concepts (with the assumption that participants are likely to behave in ways that maximize their utility).

The standard concept to predict participants’ behaviour is *Nash equilibrium* (Nash 1950), that is a strategy vector from which no player has a unilateral incentive to deviate. Thus, if each player has chosen a strategy, and no player can benefit by changing his or her strategy while the other players keep their strategies unchanged, then the current set of strategy choices and the corresponding payoffs constitute a Nash equilibrium. Nash equilibrium implies that players have complete information about their environment, including the utility functions of all other players. Furthermore, it assumes that players will play their best-response strategies to each other’s best-response strategies. These assumptions are very strong and raise the question of whether the players have sufficient knowledge to implement Nash equilibrium, i.e. whether the Nash equilibrium can model the competitive market realistically. A more realistic concept is *Bayesian equilibrium* (or Bayes-Nash equilibrium), where players have knowledge about other players, but only through a probability distribution over the possible utility functions that other players have, i.e. a “*game of incomplete information*”. Finally, the strongest concept is *dominant strategy equilibrium* (strategy-proofness). Participants have no information about the utility functions of other players and must choose strategies which maximize their payoffs independently of the choices of others. Obviously, the dominant strategy equilibrium is the least likely to exist of the three concepts and demands most from the mechanism design.
2.3.2 The Vickrey-Clarke-Groves Mechanism

The above concepts have been extensively explored. The seminal result is the Vickrey-Clarke-Groves (VCG) mechanism (Vickrey 1961; Clarke 1971; Groves 1973) which is the implementation of a dominant strategy concept that aims to maximize aggregate utility. The VCG mechanism is strategy-proof and achieves efficient allocation given rational agent strategies. The VCG mechanism offers an economically efficient answer to the multi-dimensional allocation problems, such as the combinatorial resource allocation problem (Parkes 2001). This will be discussed further in Section 2.4.

The main feature of the VCG mechanism is its price mechanism, which is an extension of Vickrey auction pricing and is often referred to as the “pivotal” rule (Clarke 1971). In the VCG mechanism, a player reports a “declared” utility function. The VCG mechanism chooses an allocation which maximizes the aggregated declared utility of all players. The payment made by each player \( i \) is the difference between the maximum possible aggregate utility of all other users if player \( i \) were not present, and the aggregate utility of all other users given that player \( i \) is present. Hence, a player is charged for resource(s) that the player wins according to valuations evaluated at the minimum signal that the player could have reported and still won that resource(s). As in the Vickrey Auction, players in VCG mechanisms therefore have the incentive to declare their own utility function truthfully, independent of the strategy of other players. This is the reason why the VCG mechanism is also referred to as the Generalized Vickrey Auction (GVA) (Ausubel 1999).

The VCG mechanism has two undesirable features from the perspective of a large scale distributed system. First, the mechanism requires players to report their entire utility function (for every possible bundle), a potentially high dimensional object. Players may not have sufficient information or may have
computationally hard evaluation problems. Second, it is NP-hard to compute payments in the VCG mechanism; it requires $i + 1$ centralized computations of maximal aggregate utility (where $i$ is the number of players) (Parkers 2001). The application of the VCG mechanism in combinational auctions and commodity markets is discussed further in Section 2.5 and Section 2.6.

2.4 Double-sided Auctions

Double-sided auctions allow many-to-many price negotiation. In a double-sided auction, both buyers and sellers can submit bids (offers to buy) and asks (offers to sell) for standardized units of well-defined commodities and securities (Friedman 1993). There are various forms of double-sided auction. Whether the auction is continuous-time or discrete-time makes the most difference when distinguishing between them.

In a discrete-time auction, all traders move in a single step (a predefined time frame) from initial allocation to final allocation. All transactions in one step are traded at the same price at the end of the time frame. In contrast with discrete-time auctions, a continuous-time auction permits exchange at any moment (i.e. real-time trading) and the overall net trade is typically composed of many bilateral transactions at different prices. Hybrid auctions are formed by a combination of continuous-time and discrete-time auctions; they have developed for practical reasons.

2.4.1 Discrete-time Double Auctions

The Clearing House (CH) auction, or Call auction, is the key example of a discrete-time double-sided auction. Its key feature is that bids and asks are
collected over specified intervals of time and then “cleared” at the expiration of the bidding interval. Given the supply and demand revealed in the bid messages, a market-clearing price is determined to maximize market efficiency. All transactions will be traded at the market-clearing price. Another prominent discrete-time example is the Walrasian Auction (Friedman 1993). In a Walrasian Auction, a Walrasian auctioneer announces a price and each participant indicates whether they wish to buy or sell at that price. If demand is not equal to supply, then the Walrasian auctioneer changes the price. No trades take place until a price is found at which demand is equal to supply. This section focuses on the CH auction as it is more common in computerized auctions.

The CH auction is designed directly under the concept of general competitive equilibrium except that participants’ preferences are provided in the simple form of bids and asks, rather than utility functions. Consumers and providers in a CH auction submit their sealed bids and asks and these are integrated into revealed supply and demand curves, from which a market-clearing price is determined. In the case of an interval \([a, b]\) in which a market-clearing price can be selected, the auction selects \(ka + (1 - k)b\) (\(k \in [0,1]\)) as the market-clearing price – this is the reason why the CH auction is also often referred to as the k-Double Auction. In the simple case where \(k = 0.5\), there is only one bid (2.0) and one ask (1.0), the market-clearing price will be 1.5. The CH auction works well for one-time allocation in a large market since the dominant strategy is nearly truth-telling in such a situation of more than 6 participants on each side (Satterthwaite and Williams 1993), i.e. the assumption that offers are truthful signals about the valuation for a resource is sound in a large market. However, as in the Vickery auction, participants in a repeated CH auction may be reluctant to reveal their true value or cost for fear of being take advantage of in subsequent auctions.
As its name suggests, the CH auction is a typical example of a market-clearing mechanism. Therefore, as already noted in Chapter 1, it begs the question of how to decide a suitable clearing-time frame when considering repeated CH auctions. Nor will it be able to offer immediate-allocation. Moreover, there is insufficient information revealed between repeated CH auctions, i.e. no market information is revealed to participants between successive clearing-time frames. The uncertainty of bidding is therefore high, as participants are kept in the dark and have to wait until the end of the clearing-time frame to know whether their bids have been successful. As a consequence, participants cannot adjust to changes in supply and demand promptly, especially when the market conditions change significantly between clearing-time frames, which leads to low efficiency. These are the main reasons why the Continuous Double Auction (CDA) has replaced the CH auction as the preferred trading mechanism in the stock market (but a call market is still used to determine the opening price on the New York Stock Exchange) (Friedman and Rust 1993). The CDA produces higher throughput than the CH auction. The CDA is discussed next.

2.4.2 Continuous-time Double Auctions

The primary example of a continuous-time double-sided auction is the Continuous Double Auction (CDA). There is no clearing-time frame in a continuous-time double auction. Bids and asks are continuously received and matched. Trades can occur at any time, i.e. there is continuous matching and clearance. The trades consist of bilateral transactions triggered by an acceptance of the best bid or ask. The matching bid and ask will be removed from the auction to form a transaction. Many such individual transactions are carried out and trading does not stop as transactions are concluded. The CDA is one of most common market formulations and is, in fact, the primary instrument for trading of equities, commodities and derivatives in financial
markets such as the London Stock Exchange (LSE) and the New York Stock Exchange (NYSE).

There are different forms of CDA that are distinguished by various factors. For example, the identity of bidders may or may not be revealed. Bids lower than the current best bid may or may not be made public or even be admissible. The most prominent form of CDA, and that most used for real world trading, is the *continuous double auction with order queue*, or *persistent shout auction*. In this scheme, a participant may make a bid or ask at any time but, once made, it persists until it is accepted or the trader chooses to alter it. This form of CDA will also be utilized in this thesis. It can be formally defined as follows:

**Definition 1:** The descriptor of a CDA with order queue is:

\[
\text{CDA} = (r, P, U, \text{ASK}, \text{BID}, a_{\text{min}}, b_{\text{max}}), \text{where:}
\]

a) \( r \) is the type of resource auctioned by the CDA.

b) \( P = \{p_1, \ldots, p_m\} \) is a finite set of identifiers of providers, where \( m \) is the number of providers.

c) \( U = \{u_1, \ldots, u_n\} \) is a finite set of identifiers of users, where \( n \) is the number of users.

d) \( \text{ASK} = \{a_1, \ldots, a_k\} \) is a finite queue of *asks* that are the amount (price) submitted by providers in ascending order, where \( k \) is the number of asks.

e) \( \text{BID} = \{b_1, \ldots, b_i\} \) is a finite queue of *bids* that are the amount (price) submitted by users in descending order, where \( i \) is the number of bids.

f) \( a_{\text{min}} \) is the current lowest *ask* of \( \text{ASK} \).

g) \( b_{\text{max}} \) is the current highest *bid* of \( \text{BID} \).

**Definition 2:** In a CDA, providers submit *asks* that may decrease \( a_{\text{min}} \), while
users submit bids that are likely to increase $b_{\text{max}}$, until $b_{\text{max}}$ is not less than $a_{\text{min}}$. At the moment of $b_{\text{max}}$ becoming not less than $a_{\text{min}}$, a transaction happens between the provider that has submitted the triggering $a_{\text{min}}$ and the consumer who has submitted the triggering $b_{\text{max}}$. The transaction price is $ka_{\text{min}} + (1-k)b_{\text{max}}$ ($k = 0.5$, normally, i.e. the mean value of the pertinent $a_{\text{min}}$ and $b_{\text{max}}$). The matching $a_{\text{min}}$ and $b_{\text{max}}$ will then be removed from the auction; the $a_{\text{min}}$ and $b_{\text{max}}$ values are replaced by the appropriate remaining bid and ask. The time period between two successful transactions is called a round; this period is non-constant.

**Definition 3:** For a CDA that has lasted $r$ ($r$ is a positive integer) rounds, $p_i$ ($0 < i \leq r$) is the $i$th transaction price. Let history $H_s$ in a CDA be the set of transaction prices during the last $s$ rounds, then

$$H_s = \{p_r, \ldots, p_{r-s}, \ldots, p_{r-s+1}\}$$

where $p_i$ ($r-s+1 \leq i \leq r$) is the transaction price of round $i$, and $s$ ($s \leq r$) is called the history length.

**Definition 4:** A CDA consists of the following steps:

1. A CDA starts with $r = 1$, ASK = $\emptyset$, BID = $\emptyset$, $a_{\text{min}} = \infty$ and $b_{\text{max}} = 0$.
2. the following situation may arise during a round:
   a. When a provider submits an ask with value $a$,
      i. if $a \geq a_{\text{min}}$ then the ask is inserted into the appropriate place in ASK;
ii. if \( b_{\text{max}} < a < a_{\text{min}} \) then \( a_{\text{min}} = a \) and \( a \) is inserted into the appropriate place in \textit{ASK};

iii. if \( a \leq b_{\text{max}} \) then this provider makes a deal at price 
\[
(0.5 \times a + 0.5 \times b_{\text{max}})
\]
with the consumer that submitted \( b_{\text{max}} \).

b. When a user submits a \textit{bid} with value \( b \),

i. if \( b \leq b_{\text{max}} \) then \( b \) is inserted into the appropriate place in \textit{BID};

ii. if \( b_{\text{max}} < b < a_{\text{min}} \) then \( b_{\text{max}} = b \) and \( b \) is inserted into the appropriate place in \textit{BID};

iii. if \( b \geq a_{\text{min}} \) then this user makes a deal at price 
\[
(0.5 \times a_{\text{min}} + 0.5 \times b)
\]
with the provider that submitted \( a_{\text{min}} \).

3. Step 2 repeats

The CDA is simple and robust, yet can achieve high efficiency. The CDA is an open auction. Therefore it is transparent and relies less on the trustworthiness of the auctioneer than a sealed auction. Most importantly, it offers a continuous matching and clearance service. Hence, the immediate allocation requirement can be fulfilled. Since orders (\textit{asks} and \textit{bids}) are cleared continuously, both consumers and providers can make instant decisions with less computational complexity and thus less overhead. In contrast with single-sided auctions and the CH auction, the consequence of a bid is clear and intuitive. The near universal finding of several decades of experimental research on the CDA has been that prices and quantities typically converge to the CE and result in highly efficient allocations under both human traders and software agents. Furthermore, the CDA is robust and can cope with many different market
situations (Friedman and Rust 1993).

On the other hand, transaction prices in a CDA may vary considerably. Such price volatility can be unacceptably high even in the case where the overall supply and demand does not vary much. Worse, on occasion, the auction may not converge to CE, causing short-term inefficient allocation. Moreover, high price volatility increases bidding difficulty and causes dissatisfaction among bidders (McCable, Rassenti et al. 1993). More importantly, it is not viable to achieve effective co-allocation and immediate-allocation in such an unstable environment. Thus, the high price volatility has to be dealt with in order successfully to apply the CDA to Grid resource allocation.

2.4.3 Hybrid Auctions

The hybrid auction sits somewhere between the continuous-time double auction and the discrete-time double auction. The Uniform-Price Double Auction (UPDA) is the main example of a hybrid auction. The UPDA tries to exploit the advantage of having more bids and asks in determining a more “fair” and stable price (as in the CH auction) combined with the advantage of continuous matching which reduces uncertainty and increases throughput (as in the CDA). The UPDA allows continuous matching but executes contracts at previously defined points in real time (time frames in the case of repeated auctions) at a unified price. Thus, when a new submitted bid(ask) is greater(less) than the current \(a_{min}(b_{max})\), a deal is guaranteed between the owner of the bid(ask) and the owner of the corresponding \(a_{min}(b_{max})\). However, the trading price will be generated by a call auction on the basis of all the successful asks and bids up to previously defined point in time. The trades are executed at a single point in time at one market-clearing price (McCable, Rassenti et al. 1993).

According to the results of the experiments presented in (McCable, Rassenti et
al. 1993), the UPDA is more favourable than the CDA in terms of price volatility and market efficiency. The UPDA design shows the possibility of combining the advantages of the CDA and the CH auction - a valuable new direction in auction design. However, the UPDA is more conceptually equivalent to a CH auction and does not provide the same degree of immediacy as the CDA. Hence it still poses the problem of deciding the length of the clearing-time frame.

### 2.5 Combinational Auctions

A combinational auction allows consumers to bid for multiple combinations of resources (a resource bundle) and only one price is submitted for the entire combination. After clearing, each consumer receives either the whole bundle or nothing. Combinational auctions are attractive for co-allocation for the obvious reason that consumers do not need to worry about the case of only obtaining part of the required resources when dealing with multiple auctions for individual resources. Moreover, certain recently developed combinational auctions allow participants to express *complementarity* and/or *substitutability* of their choices within the bids (Milgrom 2004). *Complementarity* indicates that acquiring one resource makes the bidder willing to pay more for the second. A combined resource request is an example of *complementarity*. In contrast, when more than one type of resource can fulfil a bidder’s request, *substitutability* between resources is implied, i.e. acquiring one resource makes the bidder less willing to pay for the second. There are broadly two types of combinational auctions, based on either the one-shot (or single-round) mechanism or the iterative mechanism.

In a *single-round mechanism*, all bids (for a single resource or for a bundle of
resources) are collected before the deadline and then an allocation, which normally maximizes the aggregate utility, is determined so as to identify the winning bids, immediately after the predefined deadline. There are two types of pricing scheme. One is the first-price auction, i.e. winners pay their bidding price for their resources. In this case, participants have an incentive to shade their bids to maximize their payoffs in this auction. On the other hand, as introduced in Section 2.3, the Generalized Vickrey Auction (GVA) delivers a strategy-proof solution to this problem. Participants in a GVA have an incentive to report their value truthfully.

A key issue for the single-round mechanism is the associated computational expense incurred in determining the winners. When there are no constraints on the combinatorial resource bids, i.e. the bid can consist of any arbitrary combination of resources (both complementarity and substitutability in resource choice are allowed), it is NP-complete to compute a revenue-maximizing set of bids to find winners in the form of a first-price auction (Sandholm, Suri et al. 2002), and NP-hard in the case of the GVA auction (Rothkopf, Pekec et al. 1998; Parkers 2001; Pekec and Rothkopf 2003). Approximate algorithms for determining winners in a combinational auction have been developed to solve this problem (Sandholm, Suri et al. 2002). However, the task remains complex and computationally expensive.

Furthermore, participants in a single-round mechanism often have a hard evaluation problem. That is, they need to determine values for all possible resource bundles that satisfy their requirements. There is no information to help them to generate their bids

Iterative combinational auctions are the alternative approach to single-round combinational auctions. Iterative auctions allow bidders to compute incremental values for resources or bundles of resources in response to bids
from other bidders. A prime example is the Simultaneous Ascending Auction (SAA), which was first used for Federal Communications Commission (FCC) spectrum allocation (Milgrom 2004). The key features of the SAA are that multiple rounds of English auctions are held simultaneously for each single item. The bidding process finishes when no bidder wants to change the resource allocation any further, i.e. no bidder is willing to raise their bid on any of the items. Where some items are complements, the SAA suffers from the exposure problem, since package bids are not allowed. To limit this problem, the auction rules typically allow the withdrawal of previous high bids. This allows a bidder to back out of a failed aggregation. Interestingly, an examination of withdrawals in the FCC spectrum auctions reveals that these withdrawals typically were not used for this purpose, but rather were mostly used for unanticipated bid signalling. As a result, recent SAAs have limited a bidder’s withdrawals to occur in just a few rounds of the auction. The bidding process of the SAA is long and complex; for example, each SAA normally lasted about 30 rounds in the FCC auctions. Participants usually use software to assist in their decision making.

The combinational auction is attractive as it offers the possibility directly to acquire bundles of resources. There have been many successful combinational auction applications, such as the FCC Spectrum allocation (Milgrom 2004), airport landing and takeoff scheduling (Rassenti, Smith et al. 1982) and job shop scheduling in a flexible manufacturing environment (Wellman, Walsh et al. 2001). However, a combinational auction has some undesirable features from the perspective of Grid resource allocation. First, it requires a central auctioneer and can be computationally expensive; hence, there are potential scalability problems and the auctioneer may become a bottleneck. The bidding processes in iterative combinational auctions are still too complex to be
implemented for the kind of automatic trading environment required by Grid resource allocation. Second, the transactions in combinational auctions are conducted at intervals of time, which conflicts with the need for immediate allocation. And, once again, there is the question of deciding a suitable clearing-time frame. Last, but not least, bundles of resources are treated as a whole, and it is not clear how to determine which resources contribute more (and, hence, which resources should be rewarded more) in a bundle allocation.

### 2.6 Commodity Market Model

In a commodity market model, prices of resources are decided from the overall demand and supply. A central planner gathers all the supply and demand information (the entire utility functions) about all commodities from all providers and consumers in order to generate a set of equilibrium prices. Algorithms, such as Smale’s method (Smale 1975), are used to generate the equilibrium prices. Once the equilibrium prices have been generated, all transactions are made at these generated prices in a certain period until a further set of prices is generated (Wolski, Brevik et al. 2003).

The commodity market model is able to provide a desirable scheduling environment for participants. Participants can make their local decision based on a fixed price and resources can be consumed (provided) at any time as long as there are corresponding offers (requests) standing. The market efficiency and the balance of supply and demand rely on the generated prices being equilibrium prices. However, the commodity market model is not incentive compatible. The utility functions submitted by participants are merely used for generating equilibrium. Participants are typically not forced to commit their
resource allocation as their submitted utility functions. Thus, participants will generally have an incentive to misrepresent their preference (in the form of the utility function) to the central planner in order to distort the computation of price to their advantage. Therefore, it is no surprise that the generated prices’ which are derived from participants’ utility functions, fail to reflect the overall supply and demand correctly.

Furthermore, the commodity market model lacks flexibility, since all types of resource have to be predefined. In addition, users may not be able to state or compute their utility functions reliably, especially in a highly dynamic environment.

### 2.7 Summary

The most prominent current market formulations have been introduced in this chapter. Following the discussion, two general approaches to achieving efficient allocation have emerged: incentive compatible mechanism design and iterative price-convergence. Although these two general approaches are distinct from one another, studies have shown that both of them can yield a highly efficient allocation.

The difference between the two general approaches can be illustrated by a comparison between the English auction and the Vickrey auction. The English auction represents the iterative price-convergence approach. The final price is decided by the iteration of increasing bids. It is an open auction and relies on the gradual revelation of information via iterative bidding and signalling. On the other hand, the Vickrey auction is in the category of an incentive compatible mechanism design. It is a sealed auction and strives to guarantee
efficient outcome through being strategy-proof. The beauty of the Vickrey auction is that bidders can compute their optimal strategy without modelling their counterparts’ preferences. Both auctions can yield a highly efficient outcome; in fact, they have been proven to be strategically equivalent (Vickrey 1961), as described in Section 2.2.2.

In more complex forms, considering competition from both users and providers’ sides and also combinatorial resource requests, the seminal GVA mechanism is a prime example of an incentive compatible mechanism. The CH auction and commodity market model also belong to this approach. Incentive compatible design is best applied to sealed and market-clearing mechanisms. A significant feature of incentive compatible mechanisms is that they are able to cope with the combinatorial resource allocation problem, i.e. the co-allocation problem in a Grid computing context. The CDA is a key example of the price-convergence approach. It is an open and discriminated-price mechanism. It allows continuous matching and clearance, which makes it flexible and fulfils the requirement for immediate allocation. Furthermore, the immediacy also offers the possibility to obtain multiple resources without waiting a long time (i.e. a possible instant decision on a combinatorial resource request), which facilitates co-allocation. Some mechanisms have been developed in-between the two approaches, such as the UPDA and the iterative combinational auction, which attempt to combine the strengths of both approaches.

As stated in Chapter 1, the attention of this thesis is focused on the design of a discriminated-price Grid resource allocation mechanism that can cope with a highly dynamic two-sided market and provide co-allocation and immediate allocation functions. The CDA is considered as the most suitable market formulation for Grid resource allocation for the following reasons:
The CDA offers continuous matching and clearance, which is flexible and provides a solution to immediate allocation and the possibility of co-allocation since immediate allocation of multiple resources can also be regarded as a form of co-allocation.

The CDA is simple and cheap to implement compared to other formulations, especially those based on incentive compatible mechanism designs. Its low computational cost shows both in the market formulation part and in the participants’ parts. For example, the GVA mechanism imposes a NP-complete problem on the auctioneer and a NP-hard evaluation problem on the users. The CDA, on the other hand, imposes low computational costs; there are less complex algorithms in either auctioneer or participants (participants can make instant decisions with less computational complexity and thus less overhead). Furthermore, it is an open auction; therefore it is transparent and relies less on the trustworthiness of the auctioneer than a sealed auction. All of the above features make the CDA highly scalable, which is important to Grid resource allocation as huge numbers of jobs are expected to run on a Grid computing environment daily.

The CDA offers great flexibility. There is no fixed time window to worry about. Participants in a CDA can offer/acquire resources at the time they want. A market based on the CDA can easily co-exist with other resource allocation methods, thereby allowing a gradual process of introducing market-based Grid resource allocation into existing Grid systems. There is no need to predefine the types of resource that are to be allocated. A CDA instance can be created for a new type of resource, on-the-fly and on-demand.

The CDA is robust and is able to achieve highly efficient competitive outcomes under a wide range of conditions. More significantly, the high
efficiency can be achieved under human traders, software agents or a mixture of both (Friedman 1993; Rust, Miller et al. 1993; Gjerstad and Dickhaut 1998; Das, Hanson et al. 2001). This makes feasible automatic Grid resource allocation through software agents, which is expected to be necessary for a large scale Grid environment.

However, as mentioned in Section 2.4, in its basic form, the CDA can generate high price volatility which causes dissatisfaction among bidders, increases the complexity of bidding and transaction costs, and may even fail to converge to market equilibrium (McCable, Rassenti et al. 1993; Wolski, Plank et al. 2001). The high price volatility also fails to deliver the stable environment that is required to make co-allocation and immediate-allocation decisions effectively. The high price volatility has to be dealt with in order to apply the CDA to Grid resource allocation successfully. A novel Stable Continuous Double Auction (SCDA) is designed in this thesis to tackle the problem of high price volatility while maintaining the desirable features offered by the CDA. The design of the SCDA is described in detail in Chapter 4. The next chapter analyzes the CDA in detail in preparation for guiding the design of the SCDA.
3 Analysis of the Continuous Double Auction

In the previous chapter, the CDA was identified as the most appropriate market formulation for Grid resource allocation. However, the potential high price volatility exhibited by the CDA is undesirable for this application. In order to search for a way to make the CDA even more appropriate for Grid allocation, i.e. a way to reduce the price volatility without losing the beneficial features offered by the CDA, firstly there is a need to look in more detail at the CDA and gain more understanding of it.

The analysis of an auction is essentially an analysis of the strategic interactions between participants. Non-cooperative game theory is an important tool for analyzing strategic interaction between self-interested parties. Many existing market mechanisms consist of game-theoretic concepts – these mechanisms perform well in game-theoretic equilibria, i.e. they can satisfy both economic efficiency and strategy-proofness when participants follow rationally prescribed bidding strategies. However, there is no clear game-theoretic solution for the CDA as game theory is only suitable for highly stylized, simple settings (Myerson and Satterthwaite 1983; Jennings, Faratin et al. 2001). Moreover, the game-theoretic approach also suffers from its strong common-knowledge assumption and being a static analysis. In practice, irrational behaviour of
market participants is often observed, even when the situation is straightforward. The equilibrium concepts merely guarantee that, if all market participants behave as prescribed by the equilibrium, then no participant will have an incentive to deviate in a static sense. It does not consider the dynamic process of reaching equilibrium. Hence, in the absence of robust analytical tools, much prior analysis of the CDA has used an ad hoc mixture of computer simulation and laboratory experiments (Friedman 1993; Jennings, Faratin et al. 2001). The prominent experimental results and bidding strategies are discussed in this chapter in order to gain further understanding of the CDA.

The remainder of the chapter is organized as follows: Section 3.1 discusses the bidding strategies and experiments developed for CDA, and gives the context of these bidding strategies. Then, from Section 3.2 to Section 3.7, the prominent bidding strategies are described, namely, Zero Intelligence strategy, Kaplan strategy, Fixed Markup strategy, Zero Intelligence Plus strategy, Gjerstad Dickhaunt strategy and Fuzzy-logic based strategy. Zero intelligence strategy, Kaplan strategy and Fuzzy-logic based strategy (and the corresponding experiments) make most impact on the design of the SCDA. Finally, Section 3.8 analyzes some observations taken from the study of these bidding strategies. These observations provide valuable insight into the CDA and guide the design of the SCDA, which aims to reduce high price volatility without losing the beneficial features offered by the CDA.

### 3.1 Context and Terminology

Bidding with one’s true valuation (a *truth-telling* strategy) is clearly a bad strategy in a CDA as the truth-telling strategy can be easily exploited by even slightly smarter strategies. In addition, the efficiency of a CDA is poor (it can
only achieve approximately 70-80% of potential aggregate surplus in most cases (Code and Sunder 1993)) when the CDA is populated with truth-telling participants. The reason for this poor efficiency is easy to spot: The continuous clearing rule is set to maintain a high throughput of transactions and, as a result, only a partial view of the aggregate supply and demand information is reflected. The transaction prices therefore may be far from the CE price and it may not be possible to extract some potential surplus. Moreover, the price volatility is often intolerable in this case.

However, rational participants acting locally to maximize their own profits are able to compensate for the above loss of efficiency by placing non-truthful bids, which collectively result in highly efficient outcomes. Such highly efficient outcomes have been observed universally in experimental research under human traders (Friedman and Rust 1993). Less expectedly, a CDA populated by Zero Intelligence (ZI) strategies (submitting random bids and asks with only a budget constraint) exhibits quite high efficiency and the corresponding transaction price trajectories frequently converge to CE (Code and Sunder 1993; Rust, Miller et al. 1993). This suggests that the efficient outcome of a CDA may have more to do with the properties of the CDA itself than the rationality of its participants. A strategy with minimal intelligence is able to extract high surplus from the auction. Encouraged by this observation, many heuristic bidding strategies have been developed in order to further explore the space. The above studies have the limitation that they all depend on implicit coordination via an assumption that all participants use identical bidding strategies.

In order to see what would happen if different strategies are chosen and used privately, a tournament of over 30 different software programs (implementing different bidding strategies submitted by participating institutions) was held (Rust, Miller et al. 1993). It turns out that the CDA under different software
programs behaves very like the CDA under human traders, with price trajectories converging to CE and allocations that were nearly 100% market efficient. Surprisingly, a simple “wait in the background” bidding strategy was the clear winner of the tournament, beating many more complex algorithms that use statistically based predictions of future transaction prices and sophisticated learning principles. Another interesting preliminary experiment (Das, Hanson et al. 2001) showed that software programs (agents), employing the simple bidding strategies which are discussed below (and are used later to benchmark the SCDA auction), outperformed their human counterparts in CDAs. These results strongly suggest that a simple heuristic bidding strategy may be sufficient for participating in CDAs. These prominent strategies, used in the above-mentioned experiments, will be discussed individually in the remainder of the chapter. The study of these strategies is intended to provide crucial analytical insight into the CDA.

In order to provide a platform for the discussion of bidding strategies, the structure of the CDA is presented in Figure 3.1. More specifically, this depicts the form of the CDA, open cry with order queue (i.e. unsuccessful bids and asks will stay open until they are altered by their owner or are accepted in a transaction, see Section 2.4.2), which is used in this thesis.
Figure 3.1: The structure of the CDA. The information held in the CDA is the current minimum ask, $a_{\min}$, the current maximum bid, $b_{\max}$ and the transaction price history of length $l$, $H_l$. Participants can query these values. Providers offer resources by submitting asks, and users acquire resources by submitting bids. When there is a match (a submitted bid is greater than the current minimal ask $a_{\min}$, or a submitted ask is smaller than the current maximum bid $b_{\max}$), a deal is done. The matching bid and ask will be deleted from the auction to form the transaction and the corresponding provider and user will be informed about the transaction.

In the experiments discussed in this chapter, random valuations (costs) will be assigned to the users (providers). The aggregated surplus under the CE price, which is generated from these valuations (costs), will be used as the benchmark
for allocation efficiency.

3.2 Zero Intelligence Strategy

A Zero Intelligence (ZI) buyer submits a random bid distributed independently and uniformly between 0 and the current valuation of its resource. Similarly, a ZI seller submits a random ask distributed independently and uniformly between the cost of its resource and some high value (Gode and Sunder 1993). ZI traders are “minimally rational” in the sense that they do not attempt to optimize or learn from past observations, although they do avoid trading at a loss by always bidding below their true valuation (or asking above their cost). Thus a budget constraint is applied. The interesting point is that ZI traders in a CDA are collectively rational and achieve high efficiency even though they are not individually rational. Experimental results show that a CDA populated with ZI traders can achieve high efficiency and the corresponding transaction price trajectories frequently converge to CE (Code and Sunder 1993; Rust, Miller et al. 1993). This significant finding relaxes the assumption of individual rationality and demonstrates the power of the “invisible hand” implicit in the CDA structure and trading rules. The ZI strategy is also a good benchmark strategy to analyze the effect of the market formulation itself.

The ZI strategy, like the truth-telling strategy, is easily exploited by even slightly sophisticated bidding strategies. In addition, even though price trajectories frequently converge to equilibrium, overall price volatility in a
CDA populated by ZI traders may be unreasonably high (Friedman and Rust 1993).

### 3.3 Kaplan Strategy

The Kaplan strategy is a simple, non-adaptive, non-predictive and non-optimizing strategy that makes a decision based on simple heuristics. It can be described in one sentence as “wait in the background and let others do the negotiation, but when the maximal bid $b_{\text{max}}$ and the minimal ask $a_{\text{min}}$ get sufficiently close, jump in and steal the deal”. The Kaplan strategy was developed for a tournament held at the Santa Fe Institute in 1990, in which over 30 computer programs that have been developed by institutions around the world, competed. The programs played the roles of traders in a synchronized double auction with short trading intervals. This synchronized double auction is essentially a CDA; although the transaction is set to be discrete, in order to offer equal opportunity to all programs, the trading can be treated as a continuous-time trading environment since this can be well approximated by a discrete-time environment with many short trading intervals. This short trading interval setting was necessary because different programs were running around the world and the delays of communicating with the auction program therefore varied significantly. Participating programs with long communication delays are disadvantaged in a standard CDA, so short trading intervals were imposed to create a fair environment for the participants (Rust, Miller et al. 1993). The orders are not persistent in the synchronized double auction, i.e. unsuccessful orders will be removed at the end of the short trading interval. Therefore, the
results from these tournaments may not fully apply to the CDA with the order queue setting that is used in this thesis.

The Kaplan strategy was the clear winner in these tournaments. It was able to outperform more complex algorithms that used statistically based predictions of future transaction prices, explicit optimizing principles, and sophisticated “learning algorithms”. It is simple but highly efficient and robust.

It appears that the success of the Kaplan strategy is due to the fact that, in an efficient market, if the maximum bid and minimum ask are close, then it is likely to be the case that either (1) the current bid and ask are close to the equilibrium price, or (2) the current bid and ask are close as a result of a mistake in which a trader failed to place his/her bid or ask at a sufficiently favourable price. The Kaplan strategy attempts to “steal the deal” by placing a bid equal to the previous minimum ask price, but only if it can make a profit at that price (budget constraint). By staying out of the bidding game, the Kaplan strategy is able to avoid making bidding mistakes on its own account while capitalizing on the bidding mistakes of other participants.

The reason for the relatively poor performance of complex, adaptive, optimizing and predictive strategies is due to the trading environment being dynamic and noisy. The traders taking part in the auction change all the time. The strategy used by a trader is subject to change. However, a large number of observations on the trading outcomes and a long period of time in interacting with the same group of traders are required for success in these sophisticated strategies.
The market efficiency of a synchronized double auction populated entirely by Kaplan strategy traders is bad because all the traders are waiting for their opponents to make the first move. If all traders do this, little information will be generated and the market will be unable to function efficiently. Hence, a certain portion of active bidders is needed to make the market move so that the Kaplan strategy stays in the “stealing the deal” mode. In order to avoid such a deadlock, the Kaplan strategy defaults to a “truth-telling” bidding strategy if a sufficiently long time has elapsed since the trader has made a trade. Experimental results have shown that the auction can achieve high efficiency when the number of Kaplan traders is less than 80% of the total traders. The experiments also show that the market populated with mixed strategy software agents, like the market under human traders, can achieve high efficiency (Rust, Miller et al. 1993).

3.4 Fixed Mark-up Strategy

A Fixed Mark-up provider submits the minimal ask, $a_{\text{min}}$, minus some predefined mark-up. Similarly, a Fixed Mark-up user submits the maximal bid, $b_{\text{max}}$, plus some predefined mark-up (Preist and Tol 1998). This is a very simple strategy because the trader does not model other agents or consider market conditions. Fixed Mark-up traders try to increase (decrease) their bids (asks) gradually until the resource valuation (cost) is met, i.e. budget constraint is applied.
The CDA populated entirely by Fixed Mark-up traders achieve slight lower market efficiency and lower price volatility than with the ZI traders. When we compare a Fixed Mark-up trader with a ZI trader in a same simulation environment, the Fixed Mark-up trader gains slight less surplus than the ZI trader.

### 3.5 Zero Intelligence Plus Strategy

The first version Zero Intelligence Plus (ZIP) strategy was proposed by Cliff to explore the minimum degree of agent intelligence required to reach market equilibrium in a simple version of the CDA (Cliff and Bruten 1997). The market dynamics studied in (Cliff and Bruten 1997) were unrealistic as there was no explicit notion of time and no persistent order queue in the auction being set up i.e. apart from the orders that make current $a_{\min}$ and $b_{\max}$, unsuccessful orders are withdrawn from the auction immediately. A refined version of ZIP was proposed in (Das, Hanson et al. 2001) to handle the more realistic CDA with persistent order queue (a similar approach was also proposed in (Preist and Tol 1998)). In the refined ZIP strategy, each trader maintains a vector of the internal price $P$; the $i$-th component of $P$, $p_i$, is used to set the order (bidding) price when trading the $i$-th unit. At the start of trading, the $p_i$ are initialized to random positive-surplus values (as in the ZI strategy), and are adjusted according to the observed trading actions.
When a trade occurs at trade price $P_T$, each $p_i$ is adjusted by a small random increment in the direction of $P_T$. If the adjustment is in the direction of increasing profit margin (i.e. raising $p_i$ for providers and lowering $p_i$ for buyers), the change is always made, regardless of whether or not the $i$-th unit has already been traded. However, for adjustments in the direction of decreasing profit margin, the change is made only for units that are “active”, i.e. have not yet been traded. The size of the adjustment is proportional to a learning rate parameter, $\delta$, similar to that used in Widrow-Hoff with momentum (Rumelhart and McClelland 1986), which is also used for back propagation learning in neural networks. The learning rule has two parameters: The learning rate $\beta$ determines the speed with which the adjustment takes place and the momentum $\gamma$ acts to damp oscillation. Given $\delta(t)$ and $p(t)$, the learning rate parameter and the trading price at time $t$, the learning rule determines the new learning rate parameter, $\delta(t+1)$, as follows:

$$
\delta(t+1) = \gamma \delta(t) + (1 - \gamma) \beta (p(t) - p_i) - \delta(t).
$$

Preliminary experimental results presented in (Das, Hanson et al. 2001) demonstrate that a CDA populated by ZIP traders achieves higher efficiency than a CDA populated by ZI traders. ZIP traders are not easily exploited by others.
3.6 Gjerstad Dickhaunt Strategy

The Gjerstad Dickhaunt (GD) strategy is a memory based algorithm (Gjerstad and Dickhaut 1998). GD traders record all asks and bids that they made in the history $H_m$ of their last $m$ transactions. From $H_m$, a GD trader computes a “brief” function $\hat{q}(p)$ to estimate the probability of a bid or ask at price $p$ being accepted. For example, for a GD buyer,

$$\hat{q}(p) = \frac{TBL(p) + AL(p)}{TBL(p) + AL(p) + RBG(p)}$$

where $TBL(p)$ is the number of successful bids not greater than $p$ in $H_m$, $AL(p)$ is the number of asks not greater than $p$ in $H_m$, and $RBG(p)$ is the number of unsuccessful bids not less than $p$ in $H_m$. The cubic Spline interpolation is used for prices at which no asks (bids) or transactions have been registered in $H_m$. Finally, a GD trader chooses a price that maximizes its expected surplus, defined as the product of $\hat{q}(p)$ and the profit gain from trade at that price (equal to $v - p$ for buyers and $p - c$ for sellers, where $v$ is the valuation of the resource and $c$ is the cost of the resource).

Like the ZIP strategy, a CDA populated entirely by GD traders shows convergence to equilibrium within a few rounds. The GD strategy is also much less vulnerable than the ZI strategy (Tesauro and Das 2001). However, it is important to note that the interpolation and other calculations of this strategy can be complex and computationally expensive.
More interestingly, the preliminary experiments in (Das, Hanson et al. 2001) show that ZIP traders and GD traders outperform their human counterparts in CDAs. This suggests the possibility that a simple algorithm, based on only a few parameters, may be sufficient for participating in a CDA. More complex algorithms may be unnecessary.

### 3.7 Fuzzy-logic Based Strategy

A Fuzzy-logic (FL) trader uses heuristic fuzzy rules and a fuzzy reasoning mechanism (a Sugeno controller (Sugeno 1985)) to decide the bidding price (He, Leung et al. 2003). The reference price $p_{ref}$, the median of the ordered transaction price history $H_t$, is treated as an important factor in deciding the final bidding price. Two fuzzy sets are defined for the heuristic fuzzy rules: the distance of $a_{min}$ to $p_{ref}$ and the distance of $b_{max}$ to $p_{ref}$. Given the values of two fuzzy sets, the FL trader can decide whether the current market condition is favourable and set the next bidding price accordingly. The main feature of the FL strategy is that it can dynamically vary the rate of adjustment in the bidding price according to the prevailing context. Sometimes, for example, a FL buyer can jump from a very low price to a transaction price in order to adapt to new information in the market (especially when a mistake bid has been made).

The experiments in (He, Leung et al. 2003) show the superior performance of the FL strategy over the ZI strategy, the Fixed Mark-up strategy, and the GD
strategy. The market efficiency is also not significantly affected when the CDA is populated entirely with FL traders.

3.8 Observations

From the review of the most prominent strategies developed for the CDA and from the associated experiments, several interesting observations emerge, as follows:

- The CDA itself plays an important role in generating efficient allocation outcomes – this is the power of the “invisible hand” implicit in the CDA structure and trading rules. The CDA can achieve high efficiency under human traders, multiple software programs and a mixture of both. The experiments conducted under the ZI and ZIP strategies clearly demonstrate the power of the CDA itself. Furthermore, it also suggests that the characteristics of the CDA depend less on the strategic behaviour of traders and more on the structure and constraints of the trading system itself.

- Although ZI traders can achieve high market efficiency collectively in a CDA, individual behaviour is irrational and can be easily exploited by more sophisticated strategies. Many strategies have been developed in order to investigate the strategy space. The most prominent ones, listed in this chapter, all apply a budget constraint, in order to avoid trading at a loss, and are based on only a small number of simple parameters taken from the CDA. Interestingly, these simple algorithms are quite effective and perform well. The Kaplan strategy (a simple “waiting in the
background” heuristic strategy) was the clear winner in synchronized double auctions held in a series of tournaments with over 30 different strategies (Rust, Miller et al. 1993). Both ZIP traders and GD traders outperformed their human counterparts in a series of experiments (Das, Hanson et al. 2001). These results suggest that a heuristic bidding strategy based on simple parameters ($a_{\text{min}}, b_{\text{max}}, H_l$) may be sufficient for participating in CDAs.

- Looking at this from another angle, most of the strategies are actually designed to deal with price volatility, especially unnecessary price volatility. A certain degree of price volatility is necessary and indeed important to the auction as the prices should reflect any new information or changes in overall demand and supply. However, impatient traders or those who want to make transactions quickly for reasons unrelated to the market condition, and so are insensitive to price, submit bidding prices far from the equilibrium prices, especially prices that are unfavourable to them; these cause most of the price volatility. Price volatility caused by impatient or insensitive traders is unnecessary and increases the difficulty of bidding. The bidding strategies introduced above thus strive to cope with (or take advantage of) the uncertain price environment, that is, to bid at a price that enables them to be competitive while not giving too much away and, where possible, making profit from others’ mistakes.

Considering Grid resource allocation, it is socially desirable to lower price volatility so as to provide a stable environment for co-allocation requests. It is too hard for users to deal simultaneously with multiple auctions in the presence of high price volatility. The Stable Continuous Double Auction (SCDA) is thus designed in this thesis in order to generate a more appropriate auction market for Grid resource allocation. The methodology is to reduce the unnecessary price volatility in a CDA, which is caused by impatient or insensitive
behaviours of bidders. As a result, the overall transaction prices are more stable and still reflect changes in overall demand and supply, yet the auction is still efficient and offers immediacy as in the basic form of CDA.

The above bidding strategies developed for the CDA provide a valuable reference point for the design of the SCDA. Although these strategies are designed to maximize participants' profits in the uncertain environment of a CDA, which is not exactly the same as the goal of the SCDA, they do demonstrate how to avoid and detect impatient or insensitive behaviours. The design of the SCDA will be described in the next chapter.
4 Synthesis of the Stable Continuous Double Auction

As stated in the previous chapter, the motive for the SCDA is mainly to reduce unnecessary price volatility caused by impatient and/or insensitive behaviours of bidders. It is a common observation among economists that there is more volatility than there should be in the CDA market. For example, once on one day the prices of US stock dropped 20% on no apparent news. High volatility makes participants jittery. It also creates a high ask-bid spread (the price difference between the minimum ask and maximum bid), which can make it more expensive to trade (Davis 2005).

There are two known methods used in financial markets to reduce the volatility of the CDA. One method, often used in a newly established market, is simply to limit the maximum changes in transaction prices of a CDA; for example, a constraint that does not allow price to move more than 10% of the previous day’s price in a single day. This method is not suitable for the Grid resource allocation environment as resources are highly dynamic. Constraints on transaction prices may fail to reflect the change in supply and demand. Another
method is the charging structure used by the London Stock Exchange. The orders (asks and bids) are broken down into two basic types: limit orders and market orders. If an ask is lower than the current maximum bid in the auction, the transaction will take place and the order will be removed - a market order. If the ask is higher than the current maximum bid, it will stay in the system until a matching bid is found - a limit order. The charging structure of the London Stock Exchange encourages limit orders by charging higher commission for market orders (Davis 2005). It also increases the cost of manipulating market price. However, a limit order at the time of submission may become a market order by the time the order reaches the market if the market conditions change in the time between making the limit order decision and the order reaching the auctioneer. More importantly, this charge structure may fail to tackle one of the main sources of high price volatility, namely, insensitive behaviour by bidders. When a market participant is not sensitive to the current market condition and simply wishes to sell or buy immediately, the small commission charge is unlikely to make the participant sensible.

It is considered socially desirable to reduce volatility, especially unnecessary volatility. It is more socially desirable for Grid resource allocation to give low volatility so as to provide a desirable scheduling environment for Grid applications, especially co-allocation requests. Therefore, it is acceptable to adjust “unreasonable” order prices (which are probably unfavourable to their bidders) submitted by impatient or insensitive bidders in the CDA in order to lower volatility for the mutual interests of all market participants. This is exactly the motif of the SCDA. In order to achieve the design goal of the SCDA, the following two questions need to be answered:
1) How to decide when an order is “unreasonable” for the current market condition?

2) What degree of adjustment should be made to the order price?

This chapter will describe the design of the SCDA and answer the above two questions.

The reminder of this chapter is organized as follows. Section 4.1 illustrates the architecture of the SCDA and how the information flows between components. The SCDA is to add a Compulsory Bidding Adjustment Layer (CBAL) around a standard CDA. The CBAL is the place where price adjustments happen. In Section 4.2, the principles of the price adjustment in the CBAL are given. Then, Section 4.3 describes the fuzzy logic and heuristic based implement of the price adjustment mechanism in detail. The fuzzy reasoning employs the conventional Mandani fuzzy controller (Mandani 1977; Sugeno 1985). Finally this Chapter is summarized in Section 4.4.

### 4.1 Architecture of the Stable Continuous Double Auction

As stated in the previous chapter, simple bidding algorithms that are based on a small number of simple parameters from the auction perform well and effectively in the CDA. An example is the clear winning of the Kaplan strategy in a series of tournaments with over 30 different strategies (Rust, Miller et al. 1993). Both ZIP traders and GD traders outperformed their human counterparts...
in a series of experiments (Das, Hanson et al. 2001). This strongly suggests that a heuristic bidding strategy based on simple parameters \((a_{\text{min}}, b_{\text{max}}, \text{ and } H_i)\) may be sufficient for participating in a CDA. If so, it would be possible to find a “reasonable” price for any current market condition based on the values of the simple parameters. With a “reasonable” price to hand, it is easy to answer the first question – an order price far from the “reasonable” price is an “unreasonable” price for current market condition. All unfavourable prices should be adjusted toward the “reasonable” price.

Based on the above observation, the design of the SCDA is to add a Compulsory Bidding Adjustment Layer (CBAL) around a standard CDA. The CBAL is the place where adjustments are made to unfavourable order prices submitted by impatient or insensitive market participants.

The architecture of the SCDA is shown in Figure 4.1. In a SCDA, all orders have to go through the CBAL before reaching the standard CDA. New orders will then be generated by a price adjustment mechanism in the CBAL that is based on both the current market situation \((a_{\text{min}}, b_{\text{max}}, \text{ and } H_i)\) and the original order price. The newly generated orders are then submitted to a standard CDA for trading. Participants can query information about the SCDA just as they normally do in a CDA.
Figure 4.1: *The architecture of the SCDA.*

Consider offering a resource to the SCDA as an example. The following four steps are involved:

1. Providers participate in a SCDA pretty much in the same way that they participate in a CDA. A provider submits a message, `offer(minPrice)`, to the SCDA to indicate a willingness to sell, where `minPrice` is the minimal price that the provider would accept. If the offer can be matched at the time of submitting, a contract (a deal for resource allocation) will be passed back to the provider. Otherwise, the offer will stay open in the SCDA. An identifier, which represents the offer in the
auction, will be generated by the SCDA and passed back to the provider for future reference. The provider can delete the offer as long as it has not been matched with a resource request.

2. At the time of receiving an offer message from a provider, the CBAL will query the current status of the standard CDA, i.e. $a_{\text{min}}$, $b_{\text{max}}$ and $H_l$.

3. Upon receiving the current market status information, a new ask is generated using the price adjustment mechanism in the CBAL based on both the market information and the minPrice. The new ask will then be sent to the standard CDA.

4. Once a deal is made for the new ask (the ask matches with a bid) in the standard CDA, the matching ask and bid will be removed and a contract will be generated. The corresponding provider and user will then be informed about the contract.

Resource requests follow a similar process. The price adjustment mechanism in the CBAL is at the heart of the SCDA, and this will be described in the reminder of this chapter.

### 4.2 Principles of the Price Adjustment Mechanism

The price adjustment mechanism can be divided into two tasks: (1) Find the reasonable price for the current market condition based on parameters ($a_{\text{min}}$, $b_{\text{max}}$, $H_l$).
CHAPTER 4. SYNTHESIS OF THE STABLE CONTINUOUS DOUBLE AUCTION

$b_{max}$, and $H_i$); (2) adjust any unfavorable order prices toward the reasonable price. It is important to remember that the reasonable price generated in task (1) needs to be favorable to its bidder, otherwise it will be difficult for participants to accept the price adjustment of the SCDA.

Valuable references can be obtained from the bidding strategies and experiments described in the previous chapter to help with task (1). Firstly, a budget constraint should be applied for all order price adjustments. That is, newly generated bids have to be less than the original bid and newly generated asks must be greater than or equal to the original ask submitted by the market participants. Two main intuitions are also identified in this thesis:

(a) The first intuition is learned from the success of the Kaplan strategy. That is: when the ask-bid spread is small, it is likely to be a good time to trade. A small ask-bid spread is likely to occur in the cases of when (I) $a_{min}$ and $b_{max}$ are close to the equilibrium price, or (II) the close proximity of $a_{min}$ and $b_{max}$ is the result of a mistake in which one market participant has failed to place its order at a sufficiently favorable price. Trading when the ask-bid spread is low, i.e bidding with value $a_{min}$ or asking with value $b_{max}$, can earn normal profit if case (I) holds. In the case of (II), depending on the nature of the mistake which caused the closeness of $a_{min}$ and $b_{max}$, trading when the ask-bid spread is low may earn a supernormal profit or cause a big loss. The most important feature of this intuition is that new changes in market conditions can be promptly reflected.

(b) The second intuition is that the reference price $r$ (the median value of the ordered history prices $H_j$) provides an important indication about the market
conditions, especially when considering \( r \) with \( a_{\text{min}} \) and \( b_{\text{max}} \) collectively. For example, when both \( a_{\text{min}} \) and \( b_{\text{max}} \) are less than \( r \), \( a_{\text{min}} \) is considered to be a favorable price for new bids to make supernormal profit in this intuition; when both \( a_{\text{min}} \) and \( b_{\text{max}} \) are greater than \( r \), \( a_{\text{min}} \) and \( b_{\text{max}} \) will be considered to be unfavorably high for new bids. This intuition is more about the history condition as it relies on \( r \). Hence, it will have a certain degree of delay in reflecting changes in market conditions.

Task (1) is done based on a combination of the above two intuitions. Intuition (a) reflects new changes in market conditions more efficiently and intuition (b) reflects the history of the market better. These two intuitions are complementary to one another. For example, in a big loss profit condition in case (II) of intuition (a), e.g. the closeness of \( a_{\text{min}} \) and \( b_{\text{max}} \) is caused by another resource user who bids too high when considering a new resource request, the intuition (a) would suggest a bid with value of \( a_{\text{min}} \), an incorrect action. This condition would be identified as an unfavourable case for new bids by intuition (b). The reason is because both \( a_{\text{min}} \) and \( b_{\text{max}} \) are likely to be greater than \( r \) under this market condition. Intuition (b) would suggest to bid with a value \( r \), which can counterbalance the incorrect suggestion made by intuition (a). The counterbalance is done by a fuzzy logic controller and the controller is trained by experiments.

The price adjustment mechanism employed by the CBAL needs to be heuristic as the CBAL has limited information, time and computational resources. Furthermore, task (1) of the price adjustment mechanism involves multiple factors. Fuzzy logic reasoning, on the other hand, has proven to be a heuristic
method that can cope with uncertainty and multiple factors in a timely manner. There have been many successful examples of applying fuzzy logic techniques to a wide range of domains which have inherent uncertainties and multiple factors, such as fuzzy control of traffic lights, emergency electric power distribution, and multi-site scheduling (Kosko 1992). Therefore, a heuristic and fuzzy logic reasoning price adjustment mechanism has been developed in this thesis. The design mechanism is proven to be successful according to the experimental results presented in Chapter 6. The heuristic and fuzzy logic reasoning price adjustment mechanism is explained next.

4.3 Implementation of the Price Adjustment Mechanism

This chapter will explain fuzzy logic reasoning briefly and focus on how the price adjustment mechanism is implemented. Readers are referred to (Zadeh 1965; Mandani 1977; Sugeno 1985; Zimmermann 1996) for more detail about fuzzy logic reasoning.

Fuzzy sets and fuzzy logic allow what is referred to as approximate reasoning. With fuzzy sets, an element belongs to a set with a degree of certainty. Fuzzy logic allows reasoning with these uncertain facts to infer new facts, with a degree of certainty associated with each fact. In some ways, fuzzy sets and logic allow the modelling of common sense. For the price adjustment mechanism case, the two intuitions presented in Section 4.2 are modelled as
fuzzy sets and fuzzy rules in order to infer a reasonable price for the current market condition. The conventional Mandani fuzzy controller (Mandani 1977; Sugeno 1985) is used for fuzzy logic reasoning in this thesis and will be explained below.

### 4.3.1 Mandani Fuzzy Controller.

A Mandani fuzzy controller attempts to capture intuition in the form of IF-THEN rules, and conclusions are drawn from these rules (Zadeh 1965). Based on both intuitive and expert knowledge, parameters can be modelled as linguistic variables and corresponding membership functions. Thus, nonlinear systems with great complexity and uncertainty can be effectively controlled based on fuzzy rules (through fuzzification and defuzzification processes) without dealing with complex, uncertain and error-prone mathematical models (Kosko 1992). In the case of price adjustment mechanism, the two intuitions used by the price adjustment mechanism are broken into fuzzy rules that are used to derive the reasonable price.

The architecture of a Mandani fuzzy controller is shown in Figure 4.2; it includes the following five components (the elliptical components in Figure 4.2):

- **The fuzzification process** is the input interface which maps a numerical input to a fuzzy set so that it can be matched with the premises of the defined fuzzy rules.
- **Fuzzy rules** are a set of fuzzy IF-THEN rules that define the actions of fuzzy inference in terms of linguistic variables.
• *Fuzzy sets* define membership functions of linguistic variables.

• The *inference Engine* applies the inference mechanism to the set of fuzzy rules to produce a fuzzy set output. This involves matching the input fuzzy sets with the premises of the fuzzy rules, activation of the fuzzy rules to deduce the conclusion of each rule that is fired, and combination of all activated conclusions using fuzzy set union to generate the fuzzy set output.

• The *defuzzification process* is an output mapping which converts a fuzzy set output to a non-fuzzy output.

![Diagram](image)

*Figure 4.2: The architecture of fuzzy logic inference.*

The implementation of the Mandani fuzzy controller includes the following three steps: 1) Fuzzification - defining the fuzzy set and corresponding membership functions for inputs and outputs; 2) Defining fuzzy rules from
intuitions or expert knowledge, or optimal training process; 3) Defuzzification - obtaining the non-fuzzy results form the inference process of fuzzy rules. These steps are repeated in trials in order to optimize a fuzzy controller.

The fuzzy sets and corresponding membership functions for the price adjustment mechanism are given below.

### 4.3.2 Fuzzy Sets and Membership Functions for the Price Adjustment Mechanism

The price adjustment mechanism is based on several simple inputs ($a_{\text{min}}$, $b_{\text{max}}$, $H_l$, and the original bids/asks) and generates a single output (the reasonable price). The fuzzy sets and corresponding membership functions for these inputs and output are described in this section.

The non-fuzzy inputs are defined as:

- The current minimum ask of the CDA, $a_{\text{min}}$.
- The current maximum bid of the CDA, $b_{\text{max}}$.
- The reference price of the CDA, $r$, which is the median value of the ordered transaction history $H_l$, where $l$ is the length of the history records. The median value of $H_l$ is chosen rather than the mean value because the mean value may be overly influenced by one too high (or low) price offered by an irrational trader.
- The bidding price $p$ submitted by each trader to CBAL. In the case of a
resource request (a bid), $p$ is regarded as the valuation of the resource, i.e. the maximum price that a user will accept. The final price generated by the price adjustment mechanism should be not greater than $p$. Similarly, $p$ will be treated as the cost of the resource in the case of offering a resource, i.e. the minimal price that a provider can accept.

The first fuzzy set is $(a_{\text{min}} - b_{\text{max}})$; its membership function is shown in Figure 4.3:

![Figure 4.3: The membership function of $(a_{\text{min}} - b_{\text{max}})$, where $r$ is the reference price and $\tau = 3\% \times r$. At the start of an auction, $r$ may be zero, and $(a_{\text{min}} + b_{\text{max}})/2$ is used instead in that case.](image)

The second fuzzy set is $a_{\text{min}}$; its membership function is shown in Figure 4.4.
Figure 4.4: The membership function of \( a_{\min} \), where \( \theta \) is a real number (0.1) and \( r \) is the reference price. At the start of an auction, the \( r \) may be zero, and \( (a_{\min} + b_{\max})/2 \) is used instead.

The third fuzzy set is \( b_{\max} \); its membership function is the same as the membership function of \( a_{\min} \).

The membership function of the output is shown in Figure 4.5.

Figure 4.5: The membership function of the output, where \( \tau = 3\% \times r \). The positions of \( a_{\min} \), \( b_{\max} \) and \( r \) are for the purpose of illustration; they may occur in a different order in a real situation.


4.3.3 Fuzzy Reasoning Rules and Heuristics of the Price Adjustment mechanism

Given the fuzzy sets and their membership functions, a set of fuzzy reasoning rules are defined in order to generate a favourable order price according to the prevailing market conditions. Hence, a favourable order price is a price that maximizes the bidder’s profit and is at the same time competitive.

As stated in Section 4.2, generation of the favourable order price is mainly based on two intuitions. These two intuitions are broken into fuzzy rules. For example, the first intuition, “when ask-bid spread is small, it is likely to be a good time to trade”, leads to the fuzzy rule, “If \((a_{\min} - b_{\max})\) is very-close then ask is \((b_{\max} - \tau)\)” in the case of offering a resource.

The fuzzy rules and heuristics described below have been determined as a series of experiments, which are discussed in Section 5.3. The fuzzy rule and heuristics for offering resources are described below.

4.3.3.1 Dealing with the Case of Offering Resources

The inference fuzzy rules for offering resources are as follows.

\[
\begin{align*}
& a) \quad \text{If } (a_{\min} - b_{\max}) \text{ is very-close then ask is } (b_{\max} - \tau) \\
& b) \quad \text{If } (a_{\min} - b_{\max}) \text{ is close then ask is } b_{\max} \\
& c) \quad \text{If } (a_{\min} - b_{\max}) \text{ is near then ask is } (b_{\max} + \tau) \\
& d) \quad \text{If } b_{\max} \text{ is very-high then ask is } (b_{\max} - \tau) \\
& e) \quad \text{If } b_{\max} \text{ is high then ask is } b_{\max}
\end{align*}
\]
Rules a) – c) are generated from the first intuition in Section 4.2, and the rest of the rules are derived from the second intuition. The rules work collectively to generate the favourable ask price. Take the following case, which is also used in Section 4.2, as an example. A mistake is made by a provider which causes \( a_{\text{min}} \) and \( b_{\text{max}} \) to close. Rules a) and b) will fire in this case and cause the generated ask to move towards \( b_{\text{max}} \). At the same time, the market conditions are likely to fall into one of rules l) – s): both \( a_{\text{min}} \) and \( b_{\text{max}} \) are smaller than \( r \). The firing of rules l) – s) will cause the generated ask to move towards \( r \). At the end, the final generated favourable price will be a combination of all the fired
rules, somewhere in between $b_{\text{max}}$ and $r$. For clarity, the rules d) – s) can be also described as the following matrix:

<table>
<thead>
<tr>
<th>output</th>
<th>vlow</th>
<th>low</th>
<th>mlow</th>
<th>close</th>
<th>mhigh</th>
<th>high</th>
<th>vhigh</th>
</tr>
</thead>
<tbody>
<tr>
<td>vhich</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
|        |      |     |      |       | $r + \tau$ & $(r + \tau)$ & $(r + \tau)$ & $(r + \tau)$ & $(a_{\text{min}} - \tau)$ & $(a_{\text{min}} - \tau)$ & $b_{\text{max}}$ & $b_{\text{max}}$ & $b_{\text{max}}$ & $b_{\text{max}}$
| high   |      |     |      |       | $r + \tau$ & $r + \tau$ & $r + \tau$ & $r + \tau$
| mhigh  |      |     |      |       | $r$ & $r + \tau$ & $r + \tau$ & $r$
| close  |      |     |      |       | $r$ & $r$ & $r + \tau$ & $b_{\text{max}}$
| mlow   |      |     |      |       | $r - \tau$ & $r - \tau$ & $r + \tau$ & $b_{\text{max}}$
| low    |      |     |      |       | $r - \tau$ & $r - \tau$ & $r + \tau$
| vlow   |      |     |      |       | $r - \tau$ & $r - \tau$

The creation process of the favourable price reasoning process is carried out as follows: First, the input values of $a_{\text{min}}$, $b_{\text{max}}$ and $r$ are mapped to their respective membership degree values on their membership graph (Figure 4.3 and Figure 4.4). These degree values are put into inference rules and cause some rules to fire. A set of degree values of output price (Figure 4.5) is generated as a result of the firing rules. The set of degree values of output will be used to generate a final non-fuzzy numeric output (defuzzification); the centroid method (Mandani 1977; Sowell 2002) is used in this thesis for defuzzification. In the centroid method, the crisp value of the output variable is computed by finding the variable value of the center of gravity of the membership function for the fuzzy value.

The centroid method can be formally represented as:

$$output = \frac{\sum_{i=1}^{N} C(x_i)V(x_i)}{\sum_{i=1}^{N} C(x_i)}$$
where: \( V(x) \) is the value of the membership function.

\[ C(x) \] is the value of the centre of gravity of each membership function.

\( N \) is the number of membership functions for the output.

The resulting numerical value, \( output \), is the favourable price for the current market condition. The preference of the provider should also be reflected. Two more simple heuristic rules are further applied in generating the final ask price:

- If \( output > p \), \( ask = output - 0.15 \times (output - p) \)

  where \( p \) is the price submitted by a provider, i.e. the minimal price.

- \( ask = \max(ask, p, b_{\text{max}}) \)

Basically, the above two rules are designed to increase unnecessarily low asks towards \( output \) (the favourable price). The first rule takes the users’ preference into consideration. Firstly, it checks whether \( output \) is greater than \( p \). (a) When \( p \) is greater than \( output \), \( p \) will be chosen as the final ask since \( p \) is the most favourable price following the budget constraint. There is no need to change \( p \) at all. (b) When \( output \) is greater than \( p \), \( p \) is regarded as an unnecessarily low price. The size of \( p \) should be taken into consideration. The second rule enforces two constraints: (1) a budget constraint and (2) the generated ask is not smaller than \( b_{\text{max}} \). Since the price adjustment mechanism only considers a single undivided resource offer, and no other \( ask \) can be submitted to the
standard CDA before this *ask*, there is no reason that the final ask should be smaller than $b_{\text{max}}$.

An example is given to further explain how the fuzzy logic adjustment process works. In the example, let $a_{\text{min}} = 3.79, b_{\text{max}} = 3.56, \tau = 4.30$ and a provider agent submitted its ask with value $p = 3.50$.

According to membership function of $(a_{\text{min}} - b_{\text{max}})$ shown Figure 4.3, two membership degree values are generated: 0.217 for *very-close* and 0.783 for *close* respectively. Similarly, according membership function for $a_{\text{min}}$ and $b_{\text{max}}$ (Figure 4.4), the corresponding membership degree values of fuzzy set $a_{\text{min}}$ are: 0.186 for *low* and 0.814 for *medium low*; and the corresponding membership degree values of fuzzy set $b_{\text{max}}$ are: 0.721 for *low* and 0.279 for *medium low*; Therefore the fuzzy rules a), b), k), l), and s) are fired. A set of degree values of output price is generated as a result of the firing rules. They are 0.217 for $(b_{\text{max}} - \tau)$, 0.783 for $b_{\text{max}}$, 0.279 for $(r + \tau)$, 0.721 for $(r - \tau)$ and 0.186 for $(r - \tau)$. They are illustrated in Figure 4.6.

![Figure 4.6: The degree values of output price.](image)
Therefore, the output price generated by centroid method is

\[
output = \frac{\sum_{i=1}^{N} C(x_i)V(x_i)}{\sum_{i=1}^{N} C(x_i)} = \frac{((b_{max} - \tau) \times 0.39 + b_{max} \times 0.95 + (r - \tau) \times 0.92 + (r - \tau) \times 0.34 + (r + \tau) \times 0.48)}{(0.39 + 0.95 + 0.92 + 0.34 + 0.48)} = \frac{(3.43 \times 0.39 + 3.56 \times 0.95 + 4.17 \times 0.92 + 4.17 \times 0.34 + 4.43 \times 0.48)}{3.08} = 3.93
\]

Where the centre of gravity of each membership function are calculated as follows:

\[
C(x_1) = (2 - U(x_1)) \times U(x_1) = (2 - 0.217) \times 0.217 = 0.39
\]

\[
C(x_2) = (2 - U(x_2)) \times U(x_2) = (2 - 0.783) \times 0.783 = 0.95
\]

\[
C(x_3) = (2 - U(x_3)) \times U(x_3) = (2 - 0.721) \times 0.721 = 0.92
\]

\[
C(x_4) = (2 - U(x_4)) \times U(x_4) = 0.34
\]

\[
C(x_5) = (2 - U(x_5)) \times U(x_5) = 0.48
\]

Then the final ask generated by the CBAL is

\[
ask = output - 0.15 \times (output - p) = 3.93 - 0.15(3.93 - 3.56) = 3.88
\]

### 4.3.3.2 Dealing with the Case of Requesting Resources

Similarly, the fuzzy reasoning rules for a bid are as follows.
A crisp value, output, is generated by the Centroid method:
\[
\text{output} = \frac{\sum_{i=1}^{N} C(x_i)V(x_i)}{\sum_{i=1}^{N} C(x_i)}
\]

where: \( V(x) \) is the value of the membership function.

\( C(x) \) is the value of the centre of gravity of each membership function.

\( N \) is the number of membership functions for the output.

Again, two further heuristic rules will be applied to generate the final bid:

- If \( \text{output} < p \), \( \text{bid} = \text{output} + 0.15 \times (p - \text{output}) \)

where \( p \) is the price submitted to the SCDA by a user, i.e. the maximal price,

- \( \text{bid} = \min(\text{ask}, p, a_{\text{min}}) \)

## 4.4 Summary

The SCDA is designed to reduce the impact of any impatient and/or insensitive behaviour of bidders. Such insensitive behaviours cause unnecessary price volatility. This chapter has described the design of the SCDA. A novel fuzzy logic based price adjustment mechanism has been designed to identify the unnecessary and unfavourable orders and adjust them accordingly.

One side effect of reducing volatility is that the auction may fail to reflect new information. In respect of this, special attention is given to examining whether
the SCDA is able to reflect changes in overall supply and demand promptly when the SCDA is evaluated. Furthermore, the reactions of participants towards the SCDA needs to be evaluated.

Experimental research has been conducted to illustrate the features of the SCDA. The experimental environment is discussed in the next chapter.
5 Experimental Environment

As mentioned in Chapter 3, in the current absence of robust theoretical analytical tools for the CDA, a robust theoretical analysis of the SCDA is not available because the SCDA essentially has the same or similar characteristics as the CDA. This thesis instead relies on simulation experiments to analyze the SCDA. This chapter discusses the simulation environment.

The reminder of the chapter is organized as follows:

Section 5.1 describes the simulation architecture and how the information flows among the different parts. Software agents - the auctioneer/monitor agents, the user agents and the provider agents - are developed to populate the simulation environment. The auctioneer/monitor agent is at the center of simulations. It coordinates and manages the bidding process by communicating with all the participating agents. It also monitors and evaluates the system efficiency. Moreover, the decision problem of a participating agent in a CDA is also discussed in this section as it is important for justifying the design and settings of the simulation environment.

In Section 5.2, the setting of the simulation environment is given in detail. The main difference between the experiments in this thesis and the experiments described in Chapter 3 is that the dynamic feature of the auction market is
modeled in this thesis. The experiment is a continuous process with changing CE, rather than individual games each with a static CE. This new feature facilitates observation of the response to changes in market conditions.

Section 5.3 gives details about how to construct a single experiment step-by-step. Firstly, the CDA and bidding strategies in the CDA are generated. The ZI strategy plays an important role in the experiments; it is the benchmark strategy and provides an environment to train and define other strategies. Secondly, the SCDA is created. Again, the ZI strategy is used to provide an environment in which to define the SCDA. Finally, Section 5.3.3 shows how to evaluate the scheduling efficiency.

The chapter ends with a summary in Section 5.4.

5.1 Architecture

The simulations in this thesis are mainly constructed through the interactions among three types of software agents, namely, provider agents, user agents and auctioneer/monitor agents, as illustrated in Figure 5.1.
A single auctioneer agent coordinates and manages the bidding process in one auction by communicating with all its participating user agents and provider agents (these are collectively called participating agents in the remainder of the thesis). Both the CDA and the SCDA are implemented so as to compare them. In addition, the auctioneer agent also acts as a monitor of the system efficiency, i.e. it collects the overall system data (including the true evaluation and cost for all resource offers and requests) and calculates the theoretical market efficiency for evaluation purposes. The user and provider agents communicate solely with the auctioneer (with no direct communication with each other). In the real world, it is possible that the participating agents are influenced by each other, e.g. one agent could copy another agent’s bidding strategy because the second agent’s strategy has produced better results. To explain the reason why this interaction between participating agents can be not modelled in this thesis, it is
first worth looking into each agent’s decision problem when participating in an auction.

As described in (Parkes 2001), it is helpful to assume that an agent's decision problem can be separated into a valuation problem, to compute the value of different resources, and a bidding problem, to compute an optimal bid. Figure 5.2 illustrates the agent decision problem. The arrows show the flow of information.

Figure 5.2: The agent decision problem.

The valuation and bidding problems are treated separately as shown in Figure 5.2. For example, the valuation problem can be solved with decision analysis tools and optimization methods that are independent of the particular auction, while the bidding problem can be solved with heuristic or game-theoretic methods that are based solely on the market conditions and the strategies of
other agents. The two problems are connected through information exchanges concerning valuation and meta-deliberation, i.e. the information relates to or affects the agent’s deliberation on its local valuation problem.

Although the valuation and bidding problems are considered separately, it is important to note that auction design can influence not only the bidding problem but also the effectiveness of an agent's deliberation on its local valuation problem. This is because an agent can decide how to determine its value dynamically during the course of an auction, for example, based on the current price information. Indeed, this aspect is one of the advantages that open iteration price-convergence type auctions have over incentive compatible auctions; that is, open iteration auctions can simplify an agent's valuation problem. In an open iteration auction, such as the CDA or the SCDA, the most up-to-date price information is fed to participants and this enables participants to avoid unnecessary computation on their valuation problem and bid instead with approximate valuations. In contrast, incentive compatible approaches, such as the GVC, require their participating agents to provide a full utility function in their bids. This can be a significant burden to participants when multiple resources are involved in the valuation problem – to compute all the possible utility functions for multiple resource bundles is NP-hard. Furthermore, the more stable the price offered by the SCDA, the simpler the valuation problem with the result that the agents make more effective scheduling decisions, as will be illustrated by experiments on bundle resource allocation, conducted in the next chapter.
This thesis is mainly focused on the design of the SCDA, which directly links with auction design and the bidding problem in Figure 5.2. Apart from the interactions between them, two other inputs, namely, valuation information from the internal valuation part and optional information from other agents, are also related to the bidding problem and therefore need to be considered. The valuation information is simply the input to the bidding problem since the valuation problem is separated from the bidding problem.

The problem of dealing with the optional information from other agents turns out to be tricky, in particular when an agent with information about the bidding strategies of other agents can manipulate the outcome of the auction. It is difficult to model this aspect. This also demonstrates why the strategy-proof concept plays a critical role in a game-theoretic analysis. In a strategy-proof mechanism, participating agents can make optimal decisions regardless of the other agents’ strategies. Hence, there is no longer any need to model the optional information from other agents. However, a strategy-proof solution is hard to achieve due to its strong assumptions and practical implementation issues. As this thesis focuses on iterative price-convergence mechanisms, the strategy-proof property is not present. Fortunately, zero intelligence agents have been proved to mimic the CDA market closely based on real trading data in a period of eighteen months that involved six million buy and sell orders from the London Stock Exchange (Davis 2005; Farmer, Patelli et al. 2005). Since the real stock market already includes the effect of influence between traders, zero intelligence agents offer a convenient benchmark strategy for the experiment which may be treated as if the optional information of other agents has been included in the benchmark environment. In particular, this thesis is
about the auction design, rather than designing the bidding strategies for users. With zero intelligence agents as the benchmark agents, the power of the auction market can be demonstrated and better understood.

The simulation environment is built based on a J2EE platform, namely, Sun Java System Application Server Platform. The three components are implemented as Web Services that communicate with each other through message-passing. The J2EE platform handles message-passing housekeeping and other quality of service (QoS) requirements, such as security, logging, and monitoring. This provides a suitable, open and scalable environment for both the experiment and any future real world deployment. The current development (.ear files) can be deployed on any J2EE compatible platform, which makes the experiment easy to distribute and therefore reduces the duration of the experimental work. This distributed feature is especially useful when complex bidding strategies are employed by participating agents in simulations. The algorithms for these bidding strategies can be computationally expensive, and the simulation will be very slow if all participating agents are running on a single machine. The information flow between an auctioneer and its participating agents has been illustrated in Figure 3.1 and Figure 4.1, for the CDA and the SCDA, respectively. The CDA and the SCDA are internally different, but there is no difference in terms of their interfaces and information exchange, i.e. participants communicate with both auction instances in exactly the same manner.
5.2 Experiment Settings

With the simulation architecture as defined in the previous section, this section gives details of the setting for simulations as follows.

a. Multiple user and provider agents are created for the simulations. The number of software agents is set to be five each for the user and provider agents, respectively. The reason for choosing a small number of agents is because a continuous double auction (both CDA and SCDA) is cleared frequently. Hence, the number of participating agents, especially those that will affect the outcome of the auction, is expected to be a small number at any given moment. The thesis is mainly concerned with a dynamically changing environment, since Grids are dynamic, which further justifies the small number of participating agents. Experiments reported in Chapter 3 commonly use five or six agents each (Cason and Friedman 1993; Friedman and Rust 1993; McCable, Rassenti et al. 1993; Gjerstad and Dickhaut 1998; Preist and Tol 1998; He, Leung et al. 2003). Hence, five each is chosen here in order that the new results will be comparable with these prior results.

b. As shown in Figure 5.2, a central monitor is generated to produce all the resource requests and offers, which are sent to user and provider agents, respectively. The cost of each offer is independently and randomly drawn from a uniform distribution in the interval [1.0-9.0]. Similarly, the valuation of each resource request is independently and randomly drawn from a uniform distribution in the interval [1.5-9.5]. These intervals are chosen
because the cost values are generally smaller than the valuation values (Cason and Friedman 1993). Hence, it is consistent with reality.

c. The time of resource offers and requests generated in the experiments conducted in this thesis is different from the experiments that are discussed in Chapter 3. The focus of all experiments discussed in Chapter 3 is on individual games. In an individual game, all resource offers and requests are generated once at the beginning of a game, and the game will be finished after a certain period or at the time when there are no more transactions. Many individual games are conducted to produce statistically significant results, but every single game is independent of every other single game. This setting does not represent the dynamic nature of an auction market well since the equilibrium is static in every game. Nonetheless, it is sufficient to compare different bidding strategies and test the process of price convergence in a static context. In contrast, in the experiments conducted in this thesis, resource requests and offers are generated gradually. That is, resource requests and offers are not created and passed to participating agents at the beginning of a game or a round. Instead, there is no beginning or ending of individual games. Resource requests and offers are generated continuously at random times by the central monitor. However, the number of resource requests and offers generated in a time interval, $t$, is controlled so that dynamic changes in overall supply and demand can be generated and measured. For example, in a time interval of 1 minute, the experiment is set to create a market condition in which supply roughly equals demand. That is, 50 resource requests and 50 offers will be
produced by the central monitor with costs and valuations drawn randomly from a uniform distribution as described above. Furthermore, the time that these requests and offers are sent to participating agents are independently and randomly drawn from a normal distribution in the time period of 1 minute in order to model bursts in requests and offers. By modifying the ratio of generated requests and offers, different market conditions can be generated. In addition, every request and offer is set to be valid for a fixed length of time (period, \( t \)). If a request or offer is not traded within the period \( t \), it will be removed from the auction market.

d. The above setting enables the experiment to manipulate the market conditions and assess the market’s responses to dynamic changes in overall supply and demand. However, unlike the independent game settings, where there is a static CE and potential profit value for every individual game, there is no clear separation of game rounds. In order to calculate the theoretical CE, all requests and offers in any time interval can be recorded and used to derive the CE for that time interval. Using this scheme, dynamic changes in the CE can be observed. For the potential market profit, although there is no clear separation of game rounds, the experiments can be treated as consecutive intervals of \( t \). That is, the potential market profit of a period of \((n \times t)\) will be the sum of the potential profits calculated for every time interval \( t \), based on all the requests and offers generated in that time interval. As the number of rounds \( n \) is going to be a big number in the experiments, in order to produce statistically significant results, the overall
potential profit calculation and, more importantly, the market efficiency measurement are valid, as discussed next.

e. The profits of participating agents will be calculated in the experiments in order to examine the market efficiency. For a provider agent, the gain on its $i$th unit offered is the difference between the actual transaction price, $p_i$, and the cost of the unit, $c_i$, i.e. $p_i - c_i$. If a provider offers $m$ units at prices $p_1, \ldots, p_m$ in a certain period, then its profit is $\sum_{i=1}^{m} (p_i - c_i)$. Similarly, for a user, if this user gains $n$ units of resource, its profit is $\sum_{i=n}^{m} (v_i - p_i)$, where $v_i$ is the valuation value for the $i$th unit and $p_i$ is the actual transaction price for the $i$th unit of resource. The market efficiency will be the sum of all participating agents’ profit divided by the theoretical potential market profit. When considering just a single time interval $t$, the market efficiency measurement may not be correct as some resource requests and offers (submitted and calculated in a time interval) may be traded in a subsequent time interval. But, if a large number of time intervals are used, the average market efficient measurement is valid. Indeed, in order to produce statistically significant results (so as to reduce the impact of random valuation and cost), 500 time intervals are chosen for the experiments. This number is based on a $t$-test (Stevens 2002), in which a $p$ value of 0.014 is reported when comparing results of both a sample of 450 and a sample of 500. Hence, the market efficiency variances for the two samples are virtually the same and the results are therefore statistically significant at the 98.6% level of confidence. With this large number of time intervals, the market efficient measurement is valid.
f. The CDA and the SCDA are not designed for strategy-proofness, thus participating agents probably employ some bidding strategies which aim to maximise their profit. Therefore, in order to compare the SCDA with the CDA, a benchmark strategy, which takes the strategy issue into consideration, is required. The ZI strategy is chosen as the benchmark strategy for the following reasons: (1) As already discussed, the ZI strategy closely mimics the real stock market (Davis 2005; Farmer, Patelli et al. 2005), which makes it an ideal candidate for the benchmark. (2) The market conditions are expected to be dynamic, i.e. the number of participants and their strategies may change over time. Participants with ZI strategies are more appropriate to model the dynamic situation than participants with other strategies. (3) There is no known optimal strategy for a CDA (Friedman and Rust 1993), and a user or provider’s choice of strategy may be affected by factors that are the concern only of the individual participant. In particular, users may not choose the most optimal or rational strategy for them; there are many cases in which simple strategies are chosen by the users, rather than more complex or optimal strategies (Friedman and Rust 1993).

g. Using the ZI strategy as the benchmark strategy seems more appropriate for modelling the diversity of the strategies possibly employed by participating agents. In addition, one of the main design goals of the SCDA is to reduce the complexity of participating in the auction. This can be more clearly evaluated using participating agents with ZI strategy and, even more so,
with the Truth-Telling strategy. The hypothesis is that, if the Truth-Telling strategy becomes less vulnerable to other strategies in the SCDA, the SCDA has achieved a reduction in the complexity of participating, thereby providing better information for the user to act on.

Based on the above settings, experiments can be constructed step-by-step. The detailed processes of constructing experiments and defining their parameters are described next.

5.3 Construction of Experiments

The construction of experiments involves several steps and multiple parameter definitions, which need to be managed carefully in order to produce valid and meaningful results. This section describes these steps and how each of the parameters is defined.

5.3.1 Creating the Continuous Double Auction and Agents with Different Bidding Strategies

The most widely used form of CDA, open cry with order queue, is employed in this thesis, as described in Section 3.1. The implementation is straightforward. Figure 3.1 shows the interaction between a CDA instance and its participating agents. The only thing that remains to be mentioned here is that the history length of transactions is set equal to 10 in the experiments for effective communication.
Some of the bidding algorithms introduced in Chapter 3 are implemented for evaluation purposes. These algorithms may not have been developed originally for the form of the chosen CDA in this thesis. For example, as the ZI strategy, firstly proposed by Cliff (Cliff and Bruten 1997), was designed for a CDA without a persistent order queue, and the Kaplan strategy was developed for the synchronized double auction with short trading intervals. Moreover, these algorithms in their original form were designed for a static setting with a fixed CE. Therefore, adjustments to these algorithms are to be expected in order to fit into the experimental environment of this thesis.

The first implemented strategy is the ZI strategy, which is also used as a benchmark strategy with which to train other strategies. That is, when developing a new strategy, such as the GD strategy, a single agent with the new strategy will participate in a CDA instance with other agents that all employ the ZI strategy. Different parameter settings for the new strategy are tried out and the profits of the new agent in different settings are measured. The parameter settings which generate most profit will be chosen for later use in the SCDA. This process of determining parameter settings is referred to as the training process in the reminder of this section.

With the CDA settings in this thesis, the ZI strategy is implemented as follows. Given a market condition with the maximum bid price (current best selling price) $b_{\text{max}}$, and the minimum ask price (current best buying price) $a_{\text{min}}$, a resource offer with resource cost $C$ will be submitted as an ask $(C + \tau)$, where $\tau$ is greater than 0 and is drawn randomly from a uniform distribution with interval $[0, (a_{\text{min}} - C)]$. Similarly, a resource request with resource valuation $V$ will be submitted as a bid $(V - \tau)$, where $\tau$ is greater than 0 and is drawn randomly from a uniform distribution with interval $[0, (V - b_{\text{max}})]$. 
The second implemented bidding strategy is the Truth-Telling (TT) strategy. The reason for choosing this strategy is to examine the degree of bidding complexity. The less the possibility to do better by not truthfully reporting one’s valuation, the less the degree of bidding complexity. If a mechanism is strategy proof, it should not be likely to do better than when reporting one’s true valuation. By simulating both the CDA and the SCDA with agents using the TT strategy, the bidding complexity of the two auctions can be compared. The Fixed Mark-up strategy is also developed to represent agents that simply shield their truth valuation.

Another two strategies are implemented because (1) they have demonstrated their superiority in other experiments and (2) they represent two main approaches, namely, the heuristics approach and the memory-based approach. The first of these is the Kaplan strategy and the second is the GD strategy. These are described in Section 3.3 and Section 3.6, respectively. The Kaplan strategy is based on one simple heuristic and is easy to implement. A Kaplan agent constantly monitors the auction condition and submits a bid/ask equal to the current $a_{\min}/b_{\max}$, when $a_{\min}$ and $b_{\max}$ are “close enough”. The budget constraint is enforced at the same time. A threshold of 4% of $a_{\min}$ is used to indicated “close enough” in this thesis, based on the strategy training process that was mentioned earlier. In addition, it may be the case that an offer or request does not get a chance to submit to the market because no suitable condition is detected. In that case, the offer or request will be submitted to the auction at true cost or valuation value, regardless of the market condition, before the offer/request becomes invalid. Implementation of the GD strategy is more sophisticated; for full details of the implementation see (Gjerstad and
Dickhaut 1998). The only free parameter in this strategy is the memory length, \( L_m \). In order to determine the memory length, a strategy training process is conducted. For short memory lengths \( (L_m \leq 3) \) the outcome of the GD strategy is unstable, i.e. the bidding prices and the profit of the GD agent fluctuate too much. For long memory lengths \( (L_m \geq 8) \), the outcomes are similar to those with intermediate memory lengths \( (4 \leq L_m \leq 7) \), but computational cost and time increase significantly. Furthermore, a GD agent with intermediate lengths will adapt to changes in market conditions more quickly. A memory length of \( L_m = 6 \) produces the most profit for the GD agent and this value will be used for the later experiments.

It is important to note that all the bidding strategies are created primarily for evaluation purposes, more specifically, for comparing the CDA with the SCDA, rather than to assess bidding strategies or find a superior strategy for the CDA. The methodology is to compare the CDA and the SCDA under the same experimental environment and the same bidding strategies in order to gain more insight into the similarities and differences between the two auctions.

### 5.3.2 Building the Stable Continuous Double Auction

As discussed in Chapter 4, the SCDA has the same interface and provides the same information to participating agents as the CDA. Therefore, the basis of the ZI strategy as the benchmark strategy still holds for the SCDA. Therefore, it is possible to use ZI agents to train the SCDA and determine the free parameters of the SCDA, specifically, the length of history transactions, \( l \), and the fuzzy reasoning settings of the Compulsory Bidding Adjustment Layer (CBAL). Assessment of training processes is based on three outcomes: market
efficiency, ask-bid spread and the response to changes in market conditions. The calculation of market efficiency was discussed in Section 5.2. The ask-bid spread is the distance between minimum ask and maximum bid and indicates the transaction cost. For measuring the market response to the changes in market conditions, Smith’s alpha measurement (Smith 1962) is used. This is defined as the standard deviation of the actual trade prices from the equilibrium trade price, expressed as a percentage of the equilibrium price:

\[
\alpha = \sqrt{\frac{\sum_{i=1}^{n} (p_i - p_0)^2}{np_0}}
\]

where \( p_0 \) is the equilibrium price and \( p_i \) is the actual trading price.

Because the market conditions may change significantly during the time interval \( t \), and the CE price can be calculated at any time in this thesis (see point d in Section 5.2), the value of the Smith’s alpha measurement is calculated two times in a time interval \( t \) in order to closely reflect any dynamic change in market conditions.

In considering the market efficiency, response to changes and ask-bid spread jointly, the free parameters in the SCDA are determined. The length of history transactions \( l \) is set to 8 and the fuzzy reasoning settings are chosen as described in Chapter 4. With above parameters and settings, the SCDA achieves nearly 100% market efficiency under ZI traders and closely reflects the changes in overall supply and demand, as will be shown in Section 6.2.2.

With the SCDA created, the comparison between the CDA and the SCDA can be conducted. The detailed comparisons are discussed in the next chapter. One further explanation is needed here, which is that the SCDA will be evaluated using the bidding strategies that are trained in the CDA environment. Although
these bidding strategies are not designed for participating in the SCDA, they are nonetheless still employed for the comparison for the following reasons:

1. The SCDA has the same interface and provides the same information to participating agents as the CDA. More importantly, the logic of the bidding strategies does not change; the bidding strategies cannot act differently. For example, the TT strategy is designed to submit one’s true valuation regardless of the feedback provided by the auction. The GD strategy is based on its own bidding history.

2. The focus of this thesis is not to find an optimal bidding strategy for the CDA or the SCDA. The evaluation is mainly to gain insight into the SCDA and demonstrate certain desirable features of the SCDA. The existing bidding strategies developed for the CDA are sufficient to fulfil this purpose.

5.3.3 Extending to Evaluate Scheduling Efficiency

Apart from the analysis of economic efficiency, the scheduling efficiency of the SCDA will also be evaluated. The scheduling efficiency is assessed using the user-centric performance metric that was introduced in Section 1.4.2. The user-centric performance metric is designed to measure the overall value delivered to users – whether the user’s high value jobs can be allocated more promptly than low value jobs, especially when demand exceeds supply. The SCDA will be compared with the CDA and the traditional round-robin queue mechanism in terms of user-centric performance. The same experimental setting as for the analysis of economic efficiency is used for analyzing scheduling efficiency, except that the user-centric performance metric is calculated for user agents, instead of their profits.

Furthermore, the ability of the SCDA and the CDA to handle co-allocation
requests is compared. Six auctions are set up to represent six different types of resource. Jobs that require multiple resources from different auctions are generated. The user agents receiving such multiple component jobs need to bid in multiple auctions for multiple resources in order to complete the job. The experiment is conducted using simple TT agents. When bidding for a multiple component job, the valuation of the job is divided in proportion to the median value of history prices in the targeted auctions; then the different proportions of the overall valuation are used as the bid prices for the different target auctions. For example, suppose a job involving two components has valuation 6. The median values of the history transaction prices of the two target auctions are 3.0 and 2.0, respectively. Then $6 \times \frac{3}{2+3} = 3.6$ and $6 \times \frac{2}{2+3} = 2.4$ will be submitted as values to the two target auctions, respectively. Although this bidding algorithm is very simple, a user-centric performance comparison of the SCDA and the CDA can still reveal which auction is more appropriate for co-allocation.

### 5.4 Summary

This chapter has given details of the experimental settings. The decision problem of participating in a CDA has also been described in order to explain the assumptions about the settings. The ZI strategy is regarded as the benchmark strategy for the CDA. There is a major difference between the experimental settings presented in this chapter and the experimental settings of the experiments described in Chapter 3. That difference is that, here, resource offers and requests are generated gradually to form a continuous game environment in order to observe the dynamic behaviour.

The experimental results are presented in the next chapter.
6 Evaluation

This chapter presents and evaluates experimental results obtained from the experimental environment described in the previous chapter. The focus of this chapter is to evaluate whether the SCDA is a more appropriate market formulation for market-based Grid resource allocation than the CDA, in terms of both economic efficiency and scheduling efficiency.

Based on the experimental settings described in Section 5.2, five provider agents and five user agents take part in a CDA in all the experiments conducted in this chapter. The number of resource offers and requests generated in each time interval can be manipulated to generate different overall supply and demand situations. Resource offers and requests are equally divided among participating agents. Hence, all user agents receive equal numbers of resource requests and all provider agents receive equal numbers of resource offers in each time interval. This makes the comparison of different agents, especially those that employ different strategies, easier.

Two main supply and demand conditions are used to compare the CDA with the SCDA. In the first case, the generated numbers of resource offers and requests are the same in each time interval and remain constant. In other words, the ratio of generated resource offers to requests in each time interval is 1 and
remains constant. This creates a generally stable market, although the valuation of these resource requests and offers are generated randomly. The second case represents a changing market condition. The ratio of generated resource offers to requests per time interval changes from 1 to 2. These two market conditions are referred to as the stable market condition and the changing market condition in the remainder of this chapter.

The remainder of the chapter is organized as follows. The experimental results for the CDA are presented as the benchmark results in Section 6.1. Firstly, the CDA entirely populated with Truth-Telling (TT) agents is illustrated, under both stable and changing market conditions. Then, results of the CDA with agents employing the benchmark Zero Intelligence (ZI) strategy are presented. Finally, the payoffs of different strategies in a benchmark environment are compared in order to further explore the bidding complexity of the CDA. In Section 6.2, the experimental results of the SCDA, with the same settings as the experiments for the CDA in Section 6.1, are presented. The experimental results of the first two sections compare the SCDA with the CDA in terms of price volatility, market efficiency and bidding complexity. Scheduling efficiency is analyzed in Section 6.3. Apart from the SCDA and the CDA, the traditional round-robin queue mechanism is also compared in terms of user-centric performance. The SCDA is also compared with the CDA in terms of supporting co-allocation. Section 6.4 summarizes and evaluates all the experimental results.

6.1 Results for the Continuous Double Auction

The analysis of an auction is essentially an analysis of the strategic interactions between participants. This section starts with the simplest scenario, a CDA
which is entirely populated by TT agents. Although this scenario is highly unlikely in the real world, it still provides useful insight into the CDA. More importantly, since the SCDA is able to reduce the bidding complexity, it would be interesting to compare the CDA and the SCDA in this scenario. After that, the experimental results of the CDA with agents employing the benchmark ZI strategy are presented. Both sets of results are based on two market conditions, the stable and changing market conditions, described above.

Finally, one of the user agents employs a set of different strategies while the other agents still employ the benchmark ZI strategy. The profit earned by the agent with the different strategies is then examined. By comparing the payoffs under these different strategies, the strategic space of the CDA is further explored.

### 6.1.1 Populated with Truth-Telling Agents

Figure 6.1 is a screenshot which is captured during an experiment of the CDA under TT agents in a stable market condition. Its corresponding market efficiency results are shown in Table 6.1. Similarly, the results of a changing market condition are illustrated in Figure 6.2 so as to evaluate the auction’s response to changes in overall supply and demand, and the corresponding market efficiency calculation is examined in Table 6.2.
Figure 6.1: The trading process of a CDA populated with TT agents in a stable market condition.

The table on the left hand side of Figure 6.1 illustrates the current lowest 5 *asks* at the time this screen shot is captured. The table on the right hand side shows the current highest 5 *bids* at the same time. The current minimum *ask* is 6.77 and the current maximum *bid* is 3.97. Therefore, the ask-bid spread of this CDA, which is the distance between minimum ask and maximum bid, equals 2.8. The middle subfigure of Figure 6.1 records the actual transaction prices and the equilibrium prices at each time interval. The horizontal axis of the subfigure shows the time line and the vertical axis represents price. The red curve (---) illustrates the actual trading transactions and the blue curve (--; -) shows the theoretical equilibrium price for each time interval based on the participating agents’ true resource valuations (cost) for their resource
requests (offers) in that time interval. A time interval is the gap between two marks on the blue curve. The end mark of a time interval is the equilibrium price for the time interval. It is important to note that all resource offers and requests are validated for the time period of a time interval. Unsuccessful resource offers and requests will be withdrawn from the experiment when they reach the end of their time interval. Different market conditions can be manipulated easily with the above setting. Clearly, there is a high level of volatility in transaction prices, with a high average ask-bid spread value of 2.6.

In this case, equal numbers of resource offers and requests are generated in each time interval to create a generally stable market condition. As illustrated in Figure 6.1, the generally stable market condition is reflected by the equilibrium prices; although there are some fluctuations in equilibrium prices, they are generally stable, especially during the time intervals in the middle of the subfigure of Figure 6.1. Therefore, the high volatility in Figure 6.1 is unnecessary, and causes unnecessarily high transaction costs since the market condition is generally stable.

When evaluating the economic efficiency of the auction, the theoretical potential profits in each time interval are used as the benchmark. The theoretical potential profit in a time interval is calculated as all transactions trade at the calculated equilibrium price in that time interval. The market efficiencies of 8 randomly selected time intervals in Figure 6.1 are shown in Table 6.1.
### Table 6.1: Market efficiency of a CDA populated with TT agents in a stable market condition.

<table>
<thead>
<tr>
<th>Time intervals</th>
<th>Overall Market Economic Efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Equilibrium</td>
</tr>
<tr>
<td>1</td>
<td>5.7</td>
</tr>
<tr>
<td>2</td>
<td>5.6</td>
</tr>
<tr>
<td>3</td>
<td>5.3</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>4.9</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>5</td>
</tr>
<tr>
<td>8</td>
<td>4.7</td>
</tr>
</tbody>
</table>

The market efficiency is poor in this case, only achieving levels in the range of 59% to 78%. As mentioned in Section 3.1, the poor efficiency is easy to spot: the continuous clearing rule means that only a partial view of the aggregate supply and demand information is reflected. The transaction prices may be far from the equilibrium price and, as a result, some potential profit may not be extracted.

It is worth remembering how the time interval is defined for calculating the potential profits in this thesis. There is no hard definition of time interval or individual game; a time interval is not an individual game (which is set in order to observe the responses of the auction to dynamic changes in supply and demand). Resource offers and requests are generated at continuous and random times. The overall supply and demand is therefore changed gradually. And a resource offer generated in one time interval may make a deal in the next time interval, which contributes to the variation of the market efficiency of time
intervals. Average efficiencies of time intervals are therefore used when comparing the CDA with the SCDA in order to reduce the effect of this variation.

The case of changing market condition is illustrated in Figure 6.2. The changes in overall supply and demand are reflected by the equilibrium prices in Figure 6.2. The average equilibrium price drops from around 5.5 to about 3.5 after the market condition changes.

Figure 6.2: The trading process of a CDA populated with TT agents in a changing market condition (see Figure 6.1 for legend).

Figure 6.2 shows clearly that the actual transaction prices do not reflect the changes in overall supply and demand, although the changes are clearly reflected by the theoretical equilibrium prices. The volatility of the auction is
high and the ask-bid spread of the CDA in this case is as high as in the stable market condition. The average ask-bid spread in this case is 2.5.

The corresponding market efficiency is illustrated in Table 6.2.

<table>
<thead>
<tr>
<th>Time intervals</th>
<th>Equilibrium</th>
<th>Potential Profit</th>
<th>Actual Profit</th>
<th>Efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5.6</td>
<td>109.9</td>
<td>79.1</td>
<td>72.0%</td>
</tr>
<tr>
<td>2</td>
<td>4.9</td>
<td>112.6</td>
<td>64.5</td>
<td>57.3%</td>
</tr>
<tr>
<td>3</td>
<td>6.2</td>
<td>109.9</td>
<td>81.1</td>
<td>73.8%</td>
</tr>
<tr>
<td>4</td>
<td>3.6</td>
<td>115.3</td>
<td>88.4</td>
<td>76.7%</td>
</tr>
<tr>
<td>5</td>
<td>3.2</td>
<td>145.8</td>
<td>105.8</td>
<td>72.6%</td>
</tr>
<tr>
<td>6</td>
<td>3.6</td>
<td>120.1</td>
<td>75.6</td>
<td>62.9%</td>
</tr>
<tr>
<td>7</td>
<td>4.1</td>
<td>127.5</td>
<td>101.3</td>
<td>79.5%</td>
</tr>
<tr>
<td>8</td>
<td>3.8</td>
<td>143.4</td>
<td>110.1</td>
<td>76.8%</td>
</tr>
</tbody>
</table>

Table 6.2: Market efficiency of a CDA populated with TT agents in a changing market condition.

The market efficiency in this case is similar to the case of the stable market, a poor result.

A CDA with TT traders in both the stable or changing market conditions can only achieve around 70% of potential market profit. The auction is highly volatile and the average ask-bid spread is about 2.5. Furthermore, changes in overall supply and demand are not reflected properly. It is unrealistic to model the CDA with TT strategy. However, it still helps to understand the CDA and, more importantly, is more useful when comparing with the SCDA under the
same circumstances. In the next section, the experimental results of a CDA under ZI agents, the benchmark agent for the CDA in this thesis, are revealed.

### 6.1.2 Populated with Zero Intelligence Agents

Similarly to the previous section, the experimental results are illustrated in two cases. The case of the stable market condition is shown in Figure 6.3 and Table 6.3. And the case of the changing market condition is illustrated in Figure 6.4 and Table 6.4. Instead of TT agents, ZI agents take part in the experiments in this section.

In the case of the stable market condition, the volatility of the CDA with ZI agents is reduced compared to the CDA with TT agents, but it is still substantial. The ask-bid spread of the CDA is also reduced (the average ask-bid spread has changed from 2.6 to 1.2), as shown in Figure 6.3.

Table 6.3 shows clearly that the overall market efficiency in the case of the stable market condition has been improved significantly compared to the similar case of a CDA with TT agents in Section 6.1.1 (the average market efficiency has improved from 69% to 94%). As discussed in Chapter 2, the high market efficiency of the CDA under ZI traders suggests that the characters of the CDA depend less on the strategic behaviour of traders and more on the structure and constraints of the trading system itself.
Figure 6.3: The trading process of a CDA populated with ZI agents in a stable market condition (see Figure 6.1 for legend).

<table>
<thead>
<tr>
<th>Time intervals</th>
<th>Overall Market Economic Efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Equilibrium</td>
</tr>
<tr>
<td>1</td>
<td>5.7</td>
</tr>
<tr>
<td>2</td>
<td>5.6</td>
</tr>
<tr>
<td>3</td>
<td>5.3</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>4.9</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>5</td>
</tr>
<tr>
<td>8</td>
<td>4.7</td>
</tr>
</tbody>
</table>

Table 6.3: Market economic efficiency of a CDA populated with ZI agents in the case of generally stable market conditions.
Figure 6.4: The trading process of a CDA populated with ZI agents in a changing market condition (see Figure 6.1 for legend).

<table>
<thead>
<tr>
<th>Time intervals</th>
<th>Overall Market Economic Efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Equilibrium</td>
</tr>
<tr>
<td>1</td>
<td>5.5</td>
</tr>
<tr>
<td>2</td>
<td>5.5</td>
</tr>
<tr>
<td>3</td>
<td>4.5</td>
</tr>
<tr>
<td>4</td>
<td>3.8</td>
</tr>
<tr>
<td>5</td>
<td>3.6</td>
</tr>
<tr>
<td>6</td>
<td>3.7</td>
</tr>
<tr>
<td>7</td>
<td>3.7</td>
</tr>
<tr>
<td>8</td>
<td>4.1</td>
</tr>
</tbody>
</table>

Table 6.4: Market efficiency of a CDA populated with ZI agents in a changing market condition.
As shown in Figure 6.4, the market volatility in a changing market condition is similar to the case of a stable market condition, with average market ask-bid spread of 1.2. Furthermore, the actual transaction prices reflect the changes in overall supply and demand, although the level of reflection is low and the auction is still highly volatile. High market efficiencies are also achieved in this case (average market efficiency is 94%), as shown in Table 6.4.

Although ZI agents possess “zero intelligence” in the sense that they do not attempt to optimize or learn from past observations, the collective efforts of these ZI agents on the CDA are quite significant. The market efficiency has been improved to around 94%, compared with approximately 69% in Section 6.1.1. The transaction prices also reflect the changes in overall supply and demand to a certain degree, especially in the case of significant changes in overall supply and demand. However, the price volatility is still high and the reflection to changes in overall supply and demand is not clear.

The experimental results, especially the achieved high market efficiency in this section, demonstrate the power of the CDA itself as ZI agents are “minimally rational”. The ZI agents provide a benchmark environment to compare between the SCDA with the CDA. An experiment with ZI agents focuses on the structure and constraints of the auction itself. Moreover, as mentioned in Section 5.1 and Section 5.2, the ZI strategy is also an excellent benchmark strategy for evaluating other strategies. In the next section, ZI agents will be used to create a benchmark environment for evaluating different strategies in a CDA.
6.1.3 Efficiency Comparison of Different Strategies

The experiments conducted in this section use almost the same settings as the last section, except that one user agent (u0) employs four different strategies in turn, namely the TT, Fix Mark-up, Kaplan and GD strategies. The efficiency of this user agent, u0, is compared with the efficiency of the other four user agents.

In order to evaluate strategies in different market conditions, the number of resource offers and requests generated in a time interval are changed regularly so as to create different supply and demand situations. Since the theoretical equilibrium price of a time interval reflects the overall supply and demand conditions in that time interval, different equilibrium price ranges are used to represent different market supply and demand conditions. For example, when the equilibrium price in a time interval is in the range 5 to 6, it means that the number of generated resource offers is about the same as the number of generated resource requests in this time interval. When the equilibrium price is in the range 3 to 4, the number of generated resource offers should be about two times the number of resource requests.

The efficiency analysis is conducted in time intervals. Table 6.5 shows the experimental results in randomly selected time intervals when evaluating the GD strategy.
<table>
<thead>
<tr>
<th></th>
<th>TS1</th>
<th>TS2</th>
<th>TS3</th>
<th>TS4</th>
<th>TS5</th>
<th>TS6</th>
<th>TS7</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Overall</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Eq</td>
<td>5.2</td>
<td>4.4</td>
<td>5.3</td>
<td>5.5</td>
<td>5.5</td>
<td>5.6</td>
<td>4.9</td>
<td>5.2</td>
</tr>
<tr>
<td>PP</td>
<td>119.1</td>
<td>132.8</td>
<td>118.2</td>
<td>112.3</td>
<td>116.1</td>
<td>109.9</td>
<td>112.6</td>
<td>117.3</td>
</tr>
<tr>
<td>AP</td>
<td>107.1</td>
<td>119.1</td>
<td>117.7</td>
<td>101.2</td>
<td>110.4</td>
<td>105.5</td>
<td>106.3</td>
<td>109.6</td>
</tr>
<tr>
<td>%</td>
<td>89.9%</td>
<td>89.7%</td>
<td>99.6%</td>
<td>90.1%</td>
<td>95.1%</td>
<td>96.0%</td>
<td>94.4%</td>
<td>93.5%</td>
</tr>
<tr>
<td><strong>User Agent u0</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PP</td>
<td>6.13</td>
<td>15.33</td>
<td>12.84</td>
<td>15.64</td>
<td>8.7</td>
<td>11.11</td>
<td>14.84</td>
<td>12.1</td>
</tr>
<tr>
<td>AP</td>
<td>9.08</td>
<td>11.58</td>
<td>15.23</td>
<td>16.71</td>
<td>14.86</td>
<td>15.83</td>
<td>22.92</td>
<td>15.2</td>
</tr>
<tr>
<td>%</td>
<td>148.1%</td>
<td>75.5%</td>
<td>118.6%</td>
<td>106.8%</td>
<td>170.8%</td>
<td>142.5%</td>
<td>154.4%</td>
<td>125.6%</td>
</tr>
<tr>
<td><strong>User Agent u1</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PP</td>
<td>12.96</td>
<td>10.18</td>
<td>10.45</td>
<td>10.3</td>
<td>12.01</td>
<td>17.39</td>
<td>11.51</td>
<td>12.1</td>
</tr>
<tr>
<td>AP</td>
<td>8.51</td>
<td>5.31</td>
<td>8.3</td>
<td>7.53</td>
<td>11.18</td>
<td>14.43</td>
<td>8.53</td>
<td>9.1</td>
</tr>
<tr>
<td>%</td>
<td>65.7%</td>
<td>52.2%</td>
<td>79.4%</td>
<td>73.1%</td>
<td>93.1%</td>
<td>83.0%</td>
<td>74.1%</td>
<td>75.2%</td>
</tr>
<tr>
<td><strong>User Agent u2</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PP</td>
<td>14.53</td>
<td>26.81</td>
<td>6.61</td>
<td>10.87</td>
<td>5.88</td>
<td>6.28</td>
<td>2.15</td>
<td>10.4</td>
</tr>
<tr>
<td>AP</td>
<td>8.41</td>
<td>17.45</td>
<td>11.77</td>
<td>9.97</td>
<td>2.67</td>
<td>10.06</td>
<td>1.72</td>
<td>8.9</td>
</tr>
<tr>
<td>%</td>
<td>57.9%</td>
<td>65.1%</td>
<td>178.1%</td>
<td>91.7%</td>
<td>45.4%</td>
<td>160.2%</td>
<td>80.0%</td>
<td>84.8%</td>
</tr>
<tr>
<td><strong>User Agent u3</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PP</td>
<td>9</td>
<td>14.76</td>
<td>15.01</td>
<td>9.56</td>
<td>17.69</td>
<td>7.81</td>
<td>16.54</td>
<td>12.9</td>
</tr>
<tr>
<td>AP</td>
<td>9.18</td>
<td>15.37</td>
<td>10.18</td>
<td>13.81</td>
<td>23.85</td>
<td>7.79</td>
<td>9.58</td>
<td>12.8</td>
</tr>
<tr>
<td>%</td>
<td>102.0%</td>
<td>104.1%</td>
<td>67.8%</td>
<td>144.5%</td>
<td>134.8%</td>
<td>99.7%</td>
<td>57.9%</td>
<td>99.3%</td>
</tr>
<tr>
<td><strong>Provider Agent p5</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PP</td>
<td>12.29</td>
<td>6.83</td>
<td>12.53</td>
<td>5.52</td>
<td>3.58</td>
<td>13.31</td>
<td>10.27</td>
<td>9.2</td>
</tr>
<tr>
<td>AP</td>
<td>10.5</td>
<td>8.29</td>
<td>15.76</td>
<td>5.73</td>
<td>4.83</td>
<td>10.1</td>
<td>10.04</td>
<td>9.3</td>
</tr>
<tr>
<td>%</td>
<td>85.4%</td>
<td>121.4%</td>
<td>125.8%</td>
<td>103.8%</td>
<td>134.9%</td>
<td>75.9%</td>
<td>97.8%</td>
<td>101.4%</td>
</tr>
<tr>
<td><strong>Provider Agent p6</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PP</td>
<td>17</td>
<td>16.01</td>
<td>3.9</td>
<td>10.43</td>
<td>10.62</td>
<td>8.4</td>
<td>7.1</td>
<td>10.5</td>
</tr>
<tr>
<td>AP</td>
<td>15.72</td>
<td>25.47</td>
<td>2.86</td>
<td>8.43</td>
<td>6.62</td>
<td>7.71</td>
<td>7.48</td>
<td>10.6</td>
</tr>
<tr>
<td>%</td>
<td>92.5%</td>
<td>159.1%</td>
<td>73.3%</td>
<td>80.8%</td>
<td>62.3%</td>
<td>91.8%</td>
<td>105.4%</td>
<td>101.1%</td>
</tr>
<tr>
<td><strong>Provider Agent p7</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PP</td>
<td>15.99</td>
<td>16.9</td>
<td>20.99</td>
<td>16.91</td>
<td>18.27</td>
<td>12.92</td>
<td>3.49</td>
<td>15.1</td>
</tr>
<tr>
<td>AP</td>
<td>14.9</td>
<td>12.76</td>
<td>17.78</td>
<td>9.62</td>
<td>19.14</td>
<td>8.97</td>
<td>1.81</td>
<td>12.1</td>
</tr>
<tr>
<td>%</td>
<td>93.2%</td>
<td>75.5%</td>
<td>84.7%</td>
<td>56.9%</td>
<td>104.8%</td>
<td>69.4%</td>
<td>51.9%</td>
<td>80.6%</td>
</tr>
<tr>
<td><strong>Provider Agent p8</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PP</td>
<td>8.47</td>
<td>6.29</td>
<td>12.94</td>
<td>11.73</td>
<td>13.23</td>
<td>17.4</td>
<td>14.71</td>
<td>12.1</td>
</tr>
<tr>
<td>AP</td>
<td>9.95</td>
<td>6.79</td>
<td>17.04</td>
<td>7.59</td>
<td>7.3</td>
<td>13.05</td>
<td>18.51</td>
<td>11.5</td>
</tr>
<tr>
<td>%</td>
<td>117.5%</td>
<td>107.9%</td>
<td>131.7%</td>
<td>64.7%</td>
<td>55.2%</td>
<td>75.0%</td>
<td>125.8%</td>
<td>94.6%</td>
</tr>
<tr>
<td><strong>Provider Agent p9</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PP</td>
<td>8.26</td>
<td>5.91</td>
<td>11.15</td>
<td>16.35</td>
<td>14.95</td>
<td>11.36</td>
<td>14.66</td>
<td>11.8</td>
</tr>
<tr>
<td>AP</td>
<td>6.33</td>
<td>7.63</td>
<td>8.6</td>
<td>14.82</td>
<td>10.35</td>
<td>9.55</td>
<td>11.92</td>
<td>9.9</td>
</tr>
<tr>
<td>%</td>
<td>76.6%</td>
<td>129.1%</td>
<td>77.1%</td>
<td>90.6%</td>
<td>69.2%</td>
<td>84.1%</td>
<td>81.3%</td>
<td>83.7%</td>
</tr>
</tbody>
</table>

Table 6.5: Efficiency comparison of a CDA under one GD user agent (u0) and another nine ZI agents.
In Table 6.5, columns are the time intervals (TS) and corresponding average values. Rows are the efficiency calculations for the overall market and for individual agents; Eq represents equilibrium; PP denotes potential profit in the case of all transactions trading at equilibrium price; AP is actual profit and % is efficiency (percentage of actual profit versus potential profit in a time interval).

As shown in Table 6.5, the efficiencies of an agent vary significantly from one time interval to another. One reason for this variation is that resource offers/requests are generated at random times with random valuations. A resource offer generated in one time interval may make a deal in the next time interval. This causes a mismatch between an actual profit and its corresponding potential profit in the same time interval. This mismatch is reflected more significantly in the efficiency calculations for individual agents than in those for the overall market. In order to reduce the effects of randomness on the final results and produce statistically significant results, the final results are based on data from 1000 time intervals. With this amount of data, the average efficiencies of the four Z1 agents are almost identical; the maximum difference between them is less than 2%. The average of percentages of accumulated actual profits to accumulated potential profits in time intervals is used for calculating the efficiency. It is possible to simply compare different agents’ accumulated actual profits at the end to evaluate different strategies. However, it would require a much larger amount of time interval data to produce a similar level of confidence in results.

When evaluating a strategy, the efficiency of each user agent will be accumulated in five equilibrium price ranges: 3-4, 4-5, 5-6, 6-7 and 7-8. The ratios of the evaluated user agent’s efficiency to the average value of
efficiencies for the other four ZI user agents in the same equilibrium ranges construct the final results, which are shown in Figure 6.5.

Figure 6.5: Efficiency comparison of agents with different bidding strategies in a CDA.

The horizontal axis in Figure 6.5 is the range of the calculated equilibrium price. Since the valuations of resources are uniformly distributed, when the equilibrium price is in the range 5 to 6, the overall supply and demand are about the same. If the equilibrium price is in the range 3 to 4, there is more
supply than demand. The vertical axis is the ratio of the evaluated user agent’s efficiency to the average value of efficiencies of the other four ZI user agents.

The TT agent performs poorly since it only achieves about 60% of a ZI agent’s efficiency. The Fixed Mark-up agent achieves a market efficiency similar to that of the ZI agents. Both Kaplan and GD agents are superior to the ZI agents, achieving about 20% to 40% more profit in different market conditions. In the case of equilibrium in the range 7 to 8 (supply < demand), they can make nearly 40% more profit. Evidently, the ZI strategy can be easily exploited and the payoff for a clever bidding strategy in a CDA with mostly ZI agents is significant.

6.2 Results for the Stable Continuous Double Auction

The experiments in this section are conducted, in the same manner as those in Section 6.1, in the following three steps: (1) the SCDA is populated entirely with TT agents; (2) the SCDA contains only ZI agents; (3) The efficiency of different bidding strategies is compared. Again, for the first two steps, the experimental results are illustrated in two main market conditions: the stable market and the changing market.

It is important to bear in mind that the goal of the SCDA is to reduce unnecessary volatility. A certain amount of volatility is required to move the market. Hence, the SCDA should be evaluated on both price volatility and the speed of response to changes in overall supply and demand. The ability of actual transaction prices to follow the changes in theoretical equilibrium prices
for each time interval is used to judge whether the auction market follows changes in supply and demand. The theoretical equilibrium prices are an artificial view of the dynamic trading process. They reveal the trend of the movement of market condition, but they should not be treated as absolute target values for actual transaction prices. Rather, they are indicative of the changes in overall supply and demand. The response to changes should be considered jointly with market efficiency and ask-bid spread in evaluating the SCDA.

### 6.2.1 Populated with Truth-Telling Agents

A segment of the trading process in the stable market condition is illustrated in Figure 6.6 and the corresponding efficiency calculation is shown in Table 6.6. The results for the changing market condition are presented in Figure 6.7 and Table 6.7. In addition, in view of the importance attached to having the SCDA transaction prices reflect changes in overall supply and demand (the need to evaluate whether the price adjustment mechanism of the SCDA causes slow response to changes), a more significant change scenario is also examined (the ratio of generated resource offers to requests per time interval changes from 2 to 0.5). The results for the more significant change scenario are illustrated in Figure 6.8 and Table 6.8.
Figure 6.6: The trading process of a SCDA populated with TT agents in a stable market condition (see Figure 6.1 for legend).

<table>
<thead>
<tr>
<th>Time intervals</th>
<th>Overall Market Economic Efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Equilibrium</td>
</tr>
<tr>
<td>1</td>
<td>5.6</td>
</tr>
<tr>
<td>2</td>
<td>5.3</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>4.9</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>4.7</td>
</tr>
<tr>
<td>8</td>
<td>5.2</td>
</tr>
</tbody>
</table>

Table 6.6: Market efficiency of a SCDA populated with TT agents in a stable market condition.
Figure 6.7: The trading process of a SCDA populated with TT agents in a changing market condition (see Figure 6.1 for legend).

<table>
<thead>
<tr>
<th>Time intervals</th>
<th>Overall Market Economic Efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Equilibrium</td>
</tr>
<tr>
<td>1</td>
<td>5.3</td>
</tr>
<tr>
<td>2</td>
<td>5.5</td>
</tr>
<tr>
<td>3</td>
<td>2.9</td>
</tr>
<tr>
<td>4</td>
<td>3.6</td>
</tr>
<tr>
<td>5</td>
<td>2.7</td>
</tr>
<tr>
<td>6</td>
<td>2.8</td>
</tr>
<tr>
<td>7</td>
<td>2.8</td>
</tr>
<tr>
<td>8</td>
<td>3.3</td>
</tr>
</tbody>
</table>

Table 6.7: Market efficiency of a SCDA populated with TT agents in a changing market condition
As shown in Figure 6.6, in the case of the SCDA under TT agents in the stable market condition, the price volatility is much reduced compared to the same case with the CDA in Figure 6.1. The average ask-bid spread in this case is improved to 0.41. There are more fluctuations (volatility) in the actual transaction prices than in the equilibrium prices. One reason for this is that there are bursts in the resource offers and requests, which cause fluctuations in the actual transaction prices. More importantly, the SCDA is designed to follow changes in overall market supply and demand promptly. A sufficient prompt response to changes is achieved at the expense of these acceptable fluctuations. Table 6.6 shows that the SCDA achieves nearly 100% market efficiency, with average value of 98%, a huge improvement from the 70% of the CDA.

As demonstrated in Figure 6.7 and Table 6.7, the volatility is low in a changing market condition. The changes in overall supply and demand are also clearly reflected by transaction prices. There are strong correlations between actual transaction prices and theoretical equilibrium prices. The average ask-bid spread is just 0.39. Again, the SCDA achieves nearly 100% market efficiency in this case, a very satisfactory result.

To further evaluate whether the auction market is able to follow changes in supply and demand, a case with significant changes in resource offers and requests is evaluated with the SCDA. In the significant changing market condition, the ratio of generated resource offers to requests per time interval changes from 2 to 0.5. A case in a significant changing market condition is demonstrated in Figure 6.8 and Table 6.8.
Figure 6.8: The trading process of a SCDA populated with TT agents in a significantly changing market condition (see Figure 6.1 for legend).

<table>
<thead>
<tr>
<th>Time intervals</th>
<th>Overall Market Economic Efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Equilibrium</td>
</tr>
<tr>
<td>1</td>
<td>2.6</td>
</tr>
<tr>
<td>2</td>
<td>3.4</td>
</tr>
<tr>
<td>3</td>
<td>3.4</td>
</tr>
<tr>
<td>4</td>
<td>2.7</td>
</tr>
<tr>
<td>5</td>
<td>6.9</td>
</tr>
<tr>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>7</td>
<td>6.8</td>
</tr>
<tr>
<td>8</td>
<td>7.1</td>
</tr>
</tbody>
</table>

Table 6.8: Market efficiency of a SCDA populated with TT agents in a significantly changing market condition.
Again, the SCDA demonstrates a good correlation between the actual transaction prices and theoretical equilibrium prices, even in the significant changing market condition. The changes in overall supply and demand are reflected in the transaction prices. There is a slight delay in the response to changes of the transaction prices compared to the equilibrium prices (indicated with a dotted circle in Figure 6.8). This is a compromise between stability and fast response. Recall that the SCDA is not a strategy proof mechanism. Participants are expected to respond to changes in market conditions. TT agents have no reflections to the changes in market conditions and thus these experiments are unrealistic. The case of the SCDA with ZI agents is more realistic and meaningful, as will be shown below. To further evaluate this situation, there is a need to look at the efficiency loss in such a situation. The market efficiencies in the time intervals where the change happens (time interval 4 in Table 6.8) are still high and exhibit only a small efficiency loss when significant change happens.

To sum up, the SCDA populated with TT agents demonstrates much better results than the corresponding CDA. Volatility is reduced and the changes in supply and demand are reflected promptly. The transaction prices are strongly correlated with the theoretical equilibrium prices. The auction market achieves nearly 100% market efficiency.

6.2.2 Populated with Zero Intelligence Agents

The new experiments are conducted in two market situations: results are presented for the stable market condition in Figure 6.9 and Table 6.9 and for the changing market condition in Figure 6.10 and Table 6.10.
Figure 6.9: The trading process of a CDA populated with ZI agents in a stable market condition (see Figure 6.1 for legend).

<table>
<thead>
<tr>
<th>Time intervals</th>
<th>Overall Market Economic Efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Equilibrium</td>
</tr>
<tr>
<td>1</td>
<td>5.3</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>4.9</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>4.7</td>
</tr>
<tr>
<td>7</td>
<td>5.2</td>
</tr>
<tr>
<td>8</td>
<td>4.4</td>
</tr>
</tbody>
</table>

Table 6.9: Market efficiency of a SCDA populated with ZI agents in a stable market condition.
Figure 6.10: The trading process of a SCDA populated with ZI agents in a changing market condition (see Figure 6.1 for legend).

<table>
<thead>
<tr>
<th>Time intervals</th>
<th>Overall Market Economic Efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Equilibrium</td>
</tr>
<tr>
<td>1</td>
<td>5.6</td>
</tr>
<tr>
<td>2</td>
<td>3.9</td>
</tr>
<tr>
<td>3</td>
<td>3.9</td>
</tr>
<tr>
<td>4</td>
<td>3.3</td>
</tr>
<tr>
<td>5</td>
<td>3.4</td>
</tr>
<tr>
<td>6</td>
<td>3.8</td>
</tr>
<tr>
<td>7</td>
<td>3.5</td>
</tr>
<tr>
<td>8</td>
<td>3.1</td>
</tr>
</tbody>
</table>

Table 6.10: Market efficiency of a SCDA populated with ZI agents in a changing market condition.
Both cases of the stable and changing market condition show low price volatility (see Figure 6.9 and Figure 6.10). The average ask-bid spread of the SCDA is 0.38 and 0.39 respectively. The SCDA achieves high market efficiency, with average market efficiency of around 97-98% in both cases. Unlike the CDA, there is a significant improvement in market efficiency when changing from TT agents to ZI agents. The market efficiency of the SCDA under TT agents, in fact, is similar efficiency with ZI agents (see Table 6.6 – Table 6.10).

Figure 6.10 actually demonstrates two changing market conditions. The number of generated resource offers changes from about the same as the number of resource requests to about two times more than the number of resource requests, and then to around half of the number of resource requests. Therefore, the process is a combination of a changing market condition and a significant changing market condition, which are demonstrated in Figure 6.7 and Figure 6.8 for the CDA, respectively. Clearly, the changes in overall market supply and demand are reflected by transaction prices in the SCDA. Compared to the same case with TT agents (in Figure 6.7 and Figure 6.8), a stronger correlation between theoretical equilibrium prices and actual transaction prices is seen, especially in the significantly changing market condition.

In summary, the SCDA under ZI agents achieves high market efficiency. This situation is unlike that of the CDA, where there is a significant improvement in efficiency when ZI agents replace TT agents. The SCDA, under both TT agents and ZI agents, achieves high market efficiency. In fact, the SCDA with ZI agents actually achieves slightly worse efficiency than the SCDA with TT
agents. The volatility of the SCDA is low in both cases. A strong correlation between transaction prices and equilibrium prices is also demonstrated. Hence, the SCDA auction responds to changes in overall supply and demand clearly and promptly.

6.2.3 Efficiency Comparison of Different Strategies

In the same way as the comparison conducted for the CDA, this section compares the efficiencies of different strategies under the SCDA. The experiment settings are almost identical to the settings in Section 6.1.3, except that the CDA is replaced by the SCDA. The results are shown in Figure 6.11.

![Figure 6.11: Efficiency comparison of agents with different bidding strategies in the SCDA.](image-url)
The horizontal axis is the range of the calculated equilibrium prices. Since the valuations of resources are uniformly distributed, when the equilibrium price is in the range 5 to 6, the overall supply and demand are about the same. If the equilibrium price is in the range 3 to 4, there is more supply than demand. The vertical axis is the ratio of the evaluated user agent’s efficiency to the average value of efficiencies of the other four ZI user agents.

Under the SCDA, the TT agent is no longer the worst player in the four evaluated strategies. In fact, it performs better than the Fixed Mark-up and Kaplan strategies. More importantly, there is no significant difference in the four strategies and all four evaluated agents gain profit similar to that of the ZI agent. Although the Kaplan and GD strategies are designed for the CDA, they still provide useful reference values (see Section 5.3.1). Hence, the result strongly indicates that the ZI and TT agents can no longer be exploited easily under the SCDA. The complexity of bidding in a SCDA is much less than in a CDA. This is also implied by the stable transaction prices in the previous two sections.

### 6.3 Analysis of Scheduling Efficiency

So far, in this chapter, evaluation has focused on economic efficiency. As the auction market for Grid resource allocation needs to provide a scheduling environment suitable for applications, the scheduling efficiency of the SCDA should also be evaluated.

The user-centric performance metric is used next to measure scheduling
efficiency. Detail of the user-centric performance metric has been presented in Section 1.4.2. The metric evaluates the overall value delivered to users; the more promptly high value jobs are allocated, the higher the performance of the market, especially under market conditions in which resources become scarce and users compete to obtain them.

The same experimental environment as the previous experiments in this chapter is used for evaluating scheduling efficiency. The experiments focus on market conditions in which demand exceeds supply. The number of generated resource requests in each time interval is fixed. Different numbers of resource offers in each time interval are generated in order to create different market conditions. Two experiments are performed. The first compares the user-centric performance of various market conditions under varying demand/supply ratios when users submit only single resource requests. The second investigates what happens when users submit multiple parallel requests.

### 6.3.1 Single Resource Requests

Apart from the CDA, the traditional round-robin queue mechanism is also included in this evaluation. The round-robin queue can be treated as a CDA with an equal order price for asks and bids. The participating agents simply bid or ask with the same value for resources in this case, although random valuations or costs are generated for resources and are used to calculate the corresponding user-centric performance. The SCDA and the CDA are evaluated under both TT agents and ZI agents.

For each deal that user agents make, the corresponding resource valuation and the allocation time (the period from submitting the bid to receiving confirmation of the deal) are collected in order to determine the associated user-centric performance. Detailed calculation of the user-centric performance
metric is described in Section 1.4.2. In order to produce statistically significant results, the experiments are run for 500 time intervals for each case. The final results are shown in Figure 6.12.

Figure 6.12: User-centric performance comparison. The horizontal axis is the ratio of demand to supply. The number of generated resource requests in a time interval is constant. Different demand and supply conditions are generated by modifying the number of generated resource offers.

The performance of the round-robin queue mechanism declines sharply once demand exceeds supply. That is, once supply cannot fulfil demand, the queue builds up quickly and resource requests are stuck in the queue. The speed of obtaining resource for every user agent is slowed down, and the user-centric
performance declines correspondingly. Furthermore, high value requests are traded in the same way as low value requests and, as a result, the user-centric performance is further decreased because low value jobs deliver less value to users. In contrast, both the SCDA and the CDA maintain their user-centric performance even when resources become scarce. High value resource requests are able to get a deal quickly. The SCDA is superior to the CDA in this case. High value resources are more likely to be allocated quickly in the SCDA, due to the stable transaction prices. It is important to note that a decrease of resource supply, whilst maintaining the same resource offers, naturally causes a decline in the user-centric performance in an auction market environment. In fact, the decline rate caused by the decrease of supply is about the same as the decline rate in the SCDA shown in Figure 6.12. Hence, the decline of the user-centric performance of the SCDA following the decrease of resource supplies is mainly caused by the decrease of resource requests, not by the SCDA.

### 6.3.2 Multiple Resource Requests

Co-allocation is an important requirement of a Grid computing environment, because Grid applications often involve multiple components running simultaneously. The following experiment is conducted in order to evaluate the impact of such parallel execution on user-centric performance. The experiment compares user-centric performance under the SCDA and the CDA with parallel jobs. Instead of having one auction in the experiment, six auctions are built to represent six different types of resource. Parallel jobs that require multiple resources from different auctions are generated. The number of components needed for a job is randomly drawn from a uniform distribution. Different maximum values (1-6) of components in the above random generation are evaluated in order to demonstrate the impact of different levels of parallelism.
on user-centric performance.

The evaluation is made in the case of demand/supply ratio 1.5, i.e. in a time interval, the number of generated resource requests is 1.5 times more than the number of generated resource offers for every auction. A parallel job involving \( n \) components is counted as \( n \) resource requests. Simple TT agents are employed in this experiment. For a multiple component job, the valuation of the job (the aggregated valuation of multiple resource requests) is made up of the appropriate proportions of the median values of history prices for each of the target auctions. The allocation time for the job will be the allocation time of the last allocated component. The value of delay tolerance still equals the value of the time interval. The final result is shown in Figure 6.13.

![Graph showing the comparison of user-centric performance between SCDA and CDA](image)

**Figure 6.13:** *Comparison of user-centric performance between the SCDA and the CDA under parallel jobs. The vertical axis is the ratio of the SCDA's user-centric performance to the CDA's user-centric performance*
Clearly, the higher the degree of parallelism, the better the user-centric performance of the SCDA compared to the CDA. It is possible to conclude that the SCDA is a more appropriate mechanism for the co-allocation requirement.

6.4 Summary

This chapter has evaluated the SCDA in terms of economic efficiency and scheduling efficiency. The efficiency analysis results for the CDA and the SCDA are summarized in Table 6.11.

In the scheduling analysis, the traditional round-robin queue mechanism’s user-centric performance declines sharply once demand exceeds supply. Hence, a waiting queue builds quickly and the overall value delivered to users drops significantly once resources become scarce. Resource requests have to wait in the queue for allocation and no job can be scheduled without delay. In contrast, the user-centric performance deteriorates slowly following an increase in demand relative to supply in the case of the CDA and the SCDA. The SCDA achieves better user-centric performance than the CDA in all cases. In fact, the user-centric performance of the SCDA declines at the same rate as the decline in potential user-centric performance which is caused by the decrease in resource offers. This indicates that high value resources do get allocated without much delay in a SCDA, even when resource becomes scarce.

Furthermore, the SCDA performs better than the CDA when there are resource requests that need multiple resources simultaneously. The more the degree of parallelism in resource requests, the better the user-centric performance of the SCDA relative to the CDA is. Hence, the SCDA is a more appropriate candidate to fulfil the co-allocation requirement than the CDA.
The CDA | The SCDA
--- | ---
**Populated with TT agents in a stable market condition** | (a) Unnecessarily high price volatility. The average ask-bid spread is 2.6. (b) Poor market efficiency, with average value of 69%. (a) Low price volatility. The average ask-bid spread is 0.41. (b) High market efficiency, with average value of 98%.

**Populated with TT agents in a changing market condition** | (a) High price volatility. The average ask-bid spread is 2.5. (b) No reflection of the changes in overall supply and demand. (c) Poor market efficiency, with average value of 71%. (a) Low price volatility. The average ask-bid spread is 0.39. (b) Transaction prices follow the changes in overall supply and demand. (c) High market efficiency, with average value of 98%.

**Populated with ZI agents in a stable market condition** | (a) High price volatility. The average ask-bid spread is 1.2. (b) High market efficiency, with average value of 94%. (a) Low price volatility. The average ask-bid spread is 0.38. (b) High market efficiency, with average value of 97%.

**Populated with ZI agents in a changing market condition** | (a) High price volatility. The average ask-bid spread is 1.2. (b) Slight reflection to changes in overall supply and demand. (c) High market efficiency, with average value of 94%. (a) Low price volatility. The average ask-bid spread is 0.39 (b) Transaction prices follow the changes in overall supply and demand. (c) High market efficiency, with average value of 97%.

**Efficiency comparison of different strategies** | The difference in the efficiency of strategies is significant. The TT agent is the poorest strategy and makes only approximately 60% of the profit that a ZI agent gets. In contrast, a GD agent or a Kaplan agent makes about 130% of the profit of a ZI agent. The payoff for a clever bidding strategy is high. The difference in the efficiency of strategies is small. The maximum efficiency difference is less than 10%. The TT agent is no longer the worst player. It is actually superior to Kaplan and ZI strategies. The payoff for a clever bidding strategy is not significant.

Table 6.11: *Market efficiency comparison of the CDA with the SCDA*
In terms of price volatility and market efficiency, the experimental results in this chapter show clearly that the SCDA is superior to the CDA. Moreover, the complexity of participating in the SCDA is much reduced from that required by the CDA since even the TT strategy is not easily exploited by other strategies in a SCDA. The changes in overall supply and demand are also reflected well in the SCDA. The actual transaction prices in the SCDA closely follow the changes in theoretical equilibrium prices. The stable transaction prices also make it easier for users to bid in multiple SCDAs. This further emphasizes that the SCDA is more appropriate to support co-allocation requests in Grid resource allocation. This is also confirmed by the scheduling efficiency evaluation conducted in Section 6.3.

One question needs to be asked about the SCDA. That is whether the power of users to express their preference is constrained too much in a SCDA since their bidding prices are changed by the SCDA. For example, when a user bids with a high value (a value greater than the current minimal ask price) for a resource in a CDA, the user is likely to make an immediate deal, as long as no other bids jump in before the bid is accepted by the auctioneer. The resource can therefore be allocated to the user with minimal delay. However, if the bid is submitted to a SCDA, the bidding price may be lowered by the SCDA. As a result, the deal for the bid may be delayed. This question is answered by noting the following points:

1) The price adjustment mechanism of the SCDA is designed to prevent participants making unnecessarily high bids (low asks) while not losing their competitive edge. The better user-centric performance of the SCDA over the CDA, shown in Figure 6.12 and Figure 6.13, indicates that more
overall value is delivered to users in the case of the SCDA. Hence, more high value resource requests are allocated with minimal delay in the SCDA.

2) In the SCDA, the transaction prices follow the change in theoretical equilibrium prices closely, even when the equilibrium changes dramatically (see Figure 6.7, Figure 6.8 and Figure 6.10). Therefore, the SCDA does not limit the response to changes. Indeed, the overall changes in supply and demand are better reflected in the SCDA than in the CDA.

3) The simple TT strategy is not easily exploited by “smarter” strategies in the SCDA. In fact, the TT strategy performs better than “smarter” strategies in some cases. Therefore, participants are encouraged to bid actively – there is less need to shield their true preference as truthful bidding is not punished. When more participants understand this and bid more actively, the auction market moves faster and achieves better efficiency; the high market efficiency achieved by the SCDA under TT agents, in fact, even better efficiency than the SCDA under ZI agents, is good evidence for this hypothesis.

4) As illustrated in Figure 6.13, the SCDA is more likely to fulfil the co-allocation requirement of the Grid computing environment than the CDA. Fulfilling the co-allocation requirement is crucial to the success of Grid computing. It is in the mutual interests of all parties to promote market price stability in order to provide a better application scheduling environment. A certain amount of constraint on participants’ in expressing their preference is acceptable, especially when the overall market efficiency is improved.
In summary, the experimental results demonstrate that the SCDA is a promising market formulation for building an auction market-based Grid resource allocation mechanism. The next chapter will describe the design of an auction market for Grid resource allocation and conclude this thesis.
7 Conclusion

The central motif of this thesis is the investigation of auction market design for resource allocation in Grid, a large distributed system with competing parties. The novel Stable Continuous Double Auction has been designed in consideration of both economic efficiency and the specific requirements of Grid resource allocation. The SCDA has also demonstrated many desirable features in a set of experimental evaluation.

In this chapter, a design for a SCDA market-based Grid resource allocation mechanism will be firstly described (Section 7.1). The designed SCDA market is described in order to further demonstrate how the SCDA can be used for Grid resource allocation and to indicate future research directions. The chapter concludes with some final remarks (Section 7.2).

7.1 The Stable Continuous Double Auction Market Design for Grid Resource Allocation

This section presents a design for a SCDA market for Grid resource allocation. The SCDA market consists of five core services which are introduced next.
7.1.1 Core Services

The proposed auction market consists of five core components. In order to be compatible with the Service-oriented Grid middleware, these are designed as Web Services. The five core services are:

1. **Auction Service**

   The SCDA is implemented as an auction service. Each distinct type of resource will be dealt with in a single auction service. All requests and offers for one type of resource will be traded in one auction and be treated equally.

2. **Auction Registry Service**

   Auctions are created and accessed via the auction registry service. This service maintains information about the features and status of auctions. Market participants can retrieve information about auctions and direct their resource requests and offers to a suitable auction.

3. **Provider Agent Service**

   The provider agent service seeks to achieve maximum profit for the resource providers by assisting with revenue management. An adaptive mechanism is provided to adjust the reservation price of resources according to market conditions and information from resource providers.

4. **User Agent Service**

   The user agent bids adaptively on behalf of the user in multiple auctions, under specified budget and time constraints, in order to fulfill resource requests for bundles of resources. A bidding language representing a logical combination of desired resources, such as AND, OR, XOR and CHOOSE, will be designed and
used in the user agent service to express degrees of flexibility in resource requirements. The stable price of the SCDA plays a crucial role in supporting these complex resource requirements; it makes the decision of bidding in multiple auctions much easier. After a successful trade, the user agent service may also submit jobs directly to allocated resources without further intervention from the user.

5. Ticket Service

When the auction service detects a successful trade, it will generate a ticket via the ticket service. The ticket service issues digitally signed tickets. Copies of the ticket are sent to the provider and the user via their agents. The user (or user agent) can then present the ticket to the provider so as to access the allocated resource.

Special attention is given to allow the designed auction market to work with current local schedulers. The Grid cannot be built overnight; it will evolve from current systems. The auction market resource allocation mechanism will thus need to co-exist with local resource access mechanisms. The next section shows how to integrate the SCDA market with The Portable Batch System (PBS) (OpenPBS. 2006), a popular current local scheduler as an example.

7.1.2 Integration with a Local Scheduler

The logical architecture of a typical PBS-based scheduling system is shown in Figure 7.1.
Figure 7.1: Logical architecture of a typical local scheduler, which is responsible for scheduling jobs submitted to a resource provider on local resources. The job-resource mapper contains the policy for scheduling jobs on resources. This mapper obtains information about jobs from the job queue manager, and about resources from the resource manager. Users submit jobs to the queue manager by some job submission process that involves authorisation and authentication. Accounting and monitoring information is available to the site administrator. In the PBS system (PBS, 2006), the job queue manager is the PBS job server; the job-resource mapper is the PBS scheduler; and the resource manager is the PBS MOM.

As stated in Chapter 1, the auction market is designed for flexibility, and participants can offer or acquire resources at any time. To further increase flexibility, users will only pay for the resources they have used. Participation in the market by resource providers will require few changes to local scheduling policies, since the market scheme will remain compatible with existing batch
queues and AR provision. Hence, the auction market will be able to co-exist with the current local scheduling mechanism in a complementary fashion. The integration of the SCDA market with local schedulers is illustrated in Figure 7.2.

Figure 7.2: The integration of the SCDA market with local schedulers. The Grid consists of the resources of multiple sites. Site-local schedulers can send resource offers to the market, and Grid users can submit resource requests associated with jobs. Site-local schedulers maintain special queues for Grid jobs matched to local resources by the auction market. Site-local schedulers can still accept jobs from local users. Site-local administrators can monitor and access accounting data for both local and Grid users.

All resources that are offered to the auction market are required to be free before submitting the offer. Hence, resources in the market can be allocated
with minimal delay from the time of the request. This fulfils the requirement for immediate allocation of resource requests, and to enable bundled resource requests (simultaneous co-allocation of multiple resources) to be allocated with minimal delay from the time of the request. An important feature is that the difficulty of coordinating multi-site resource allocation agreements is reduced. That is, when a bid reaches the offer price for a resource, an agreement is made immediately – no further communication with the site-local schedulers is necessary. The market consists of multiple auctions, each auction representing bidding for a different type of resource. Similar bids and offers go to the same auction in order to achieve a competitive outcome.

A request for multiple resources will check all relevant auctions to determine whether the current sum of prices of all required resources is acceptable. If the price is acceptable, the request can bid in all pertinent auctions in order to get the bundle of resources as near to the current price. The SCDA has the property of continuous clearance, which means that requests can obtain a resource without having to wait. Thus, if the bundle of requests is not immediately completely successful, either bids for the missing resources can be made at appropriately higher prices, or the acquired resources can be released and new bids made for the whole resource bundle at some future time.

7.2 Final Remarks

Any attempt to design an efficient resource allocation mechanism for a large distributed system with competing parties must answer, at the very least, the following questions:

1. What constitutes an efficient allocation?
2. What information is available to the market participants? And what is the expected reaction of the market participants towards the information?
3. What is the complexity of the mechanism, in terms of computation, communication and complexity of participating?
4. Can the mechanism be evolved from existing systems?

For auction market-based Grid resource allocation, this thesis defines efficiency in terms of both economic efficiency and scheduling efficiency. Economic efficiency is the primary efficiency measurement and is the percentage of the actual aggregated surplus to potential aggregated surplus in the market place (Pareto efficiency). But the designed auction market must provide a desirable scheduling environment to the market participants; more specifically, it must have the ability to offer resource and resource bundles with minimal delay to high value jobs (immediate allocation and co-allocation). Scheduling efficiency is measured by the user-centric performance metric. Basically, this evaluates whether high value resource requests are allocated promptly.

This thesis focuses on the discriminated price mechanism due to the highly dynamic nature of the Grid environment. Furthermore, it relies on iterative price-convergence to achieve efficient allocation. The CDA is regarded as the most appropriate existing market model for Grid resource allocation in this thesis. The experimental and real case results of the CDA have demonstrated that the CDA is simple and yet is able to achieve high market efficiency with only little information passing to participants and low computational cost. Furthermore, it offers continuous matching and clearance, which makes it flexible and fulfils the requirement for immediate allocation. However, the CDA has unnecessarily high volatility, which causes dissatisfaction among participants and difficulty in co-allocation. The novel SCDA has been presented in this thesis as a means to overcome this undesirable high volatility while maintaining the other beneficial features of the CDA. Experiments reported in this thesis have shown that the SCDA reduces unnecessary volatility and the inherent complexity of participating in the auction while achieving nearly
100% market efficiency. The information provided to the market participants is no more than the CDA and the market supports continuous clearance. Furthermore, unlike the CDA, the simple TT bidding strategy is no longer easy to exploit in the SCDA. This strongly indicates that the SCDA reduces bidding complexity and encourages active bidding.

There is little complexity involved in the SCDA market. The matching process of the CDA is not computationally expensive, and the communication between parties is low. Participants can make their decisions mostly or entirely on their own local information. Furthermore, the open iteration price-convergence feature of the SCDA simplifies the evaluation problem of participants (see Section 5.1). In contrast, incentive compatible mechanisms, such as the GVA mechanism, typically involve complex and computationally expensive tasks and present a hard evaluation problem to market participants (see Chapter 2 for details). The major advantage of the GVA mechanism is that the co-allocation problem is solved inside the auction. In contrast, in the SCDA, users or user agents need to take responsibility for co-allocation requests. However, the immediate allocation feature and the stable environment offered by the SCDA ease the co-allocation problem. As stated in Section 1.2, immediate allocation may be more significant than co-allocation for creating the desired resource scheduling environment for Grid applications.

Furthermore, the SCDA market is flexible. It can work easily alongside other resource allocation mechanisms. This enables the incremental evolution of the Grid resource market; for such a large distributed system, it is impossible to build from scratch.

Although the experimental results of the SCDA are promising, the reactions of participants towards the SCDA still need to be evaluated in real cases. How to better support complex resource requests is an open question. Derivative
markets, or something similar to futures markets in financial trading, may offer an answer. But an efficient immediate market is the foundation. The author plans to further investigate the SCDA market Grid allocation in real Grid environments.

As highlighted several times in this thesis, the focus of this work is on a discriminated price mechanism that relies on iterative price-convergence to achieve efficient allocation. These are all choices open to debate. Indeed, the author hopes that the results of this thesis serve as a springboard to a search for further answers to the four questions posed above.
Bibliography


Maclaren, J., R. Sakellariou, et al. (2004). Towards Service Level Agreement Based Scheduling on the Grid. Workshop on Planning and Scheduling for Web and Grid Services (in conjunction with the 14th International Conference on Automated Planning and Scheduling - ICAPS-04), Whistler, British Columbia, Canada.


IPPS/SPDP 99 workshop on Job Scheduling Strategies for Parallel Processing: 80-88.


