

Foreword

The emergence of the Internet as a global platform for computation and communication has sparked the development and deployment of large-scale open distributed systems. The scale of such systems entails a large and heterogeneous set of users with often divergent motives. At the same time, open systems give users increased power to operate in pursuit of local interests, even at the expense of global system objectives. To retain the fast pace of innovation facilitated by open systems while ensuring their long-term reliability and scalability, it is essential that we design efficient protocols that are robust in the presence of self-interested agents. This need has motivated the recent interest in the study of incentives in many areas of distributed computing such as: resource allocation in computational grids, peer-to-peer systems, task scheduling, congestion control, routing and multi-agent systems. Making incentives an explicit aspect of system design represents a revolutionary change in computer science. The great challenge in this area is to design incentive compatible protocols that are computationally tractable and consistent with other application requirements. Research in this area is often cross-disciplinary, borrowing knowledge from economics, computer science, computer engineering and game theory.

Although the number of research projects and publications dealing with incentives has grown in recent years, there are still very few events dedicated exclusively to this important topic. IBC'05 represents a timely opportunity to bring together a community of researchers involved in economics, game theory and computing.

The Program Committee members have done an excellent job in selecting the papers. All submitted papers have undergone a thoroughly review process with each paper having at least two reviewers providing feedback to the authors. We selected five papers out of eight submissions for the presentation at the Workshop.

We would like to thank the members of the Program Committee who gave their time and energy to ensure the high quality of the technical program. We would also like to thank the WI/IAT 2005 organizers for their support and encouragement. Especially we would like to thank the WI/IAT 2005 Chairs, Pierre Morizet and Jiming Liu and WI/IAT Workshop Chair, Pawan Lingras. We also are very grateful to Paul Spirakis for his invited talk on congestion games.

Finally, we would like to thank the authors of the selected papers for their contributions to the new and exciting field of incentive-based computing.

Daniel Grosu and Jonathan Shapiro
IBC'05 Workshop Chairs

IBC'05 Organization

Workshop Chairs

Daniel Grosu
Department of Computer Science
Wayne State University

Jonathan Shapiro
Dept. of Computer Science and Engineering
Michigan State University

Program Committee

Sonja Buchegger (University of California Berkeley, USA)
Rajkumar Buyya (University of Melbourne, Australia)
Anthony Chronopoulos (University of Texas, San Antonio, USA)
Bruno Codenotti (IIT-CNR Pisa, Italy)
Vincent Conitzer (Carnegie Mellon University, USA)
Xiaotie Deng (City University of Hong Kong)
Daniel Figuiere (EPFL, Switzerland)
Kate Larson (University of Waterloo, Canada)
Xiang-Yang Li (Illinois Institute of Technology, USA)
Patrick Maille (ENST Bretagne, France)
David Parkes (Harvard University, USA)
Amir Ronen (Technion, Israel)
Tim Roughgarden (Stanford University, USA)
Marius Calin Silaghi (Florida Institute of Technology, USA)
Paul Spirakis (University of Patras, Greece)
Milan Vojnovic (Microsoft Research, Cambridge, UK)
William Walsh (CombineNet Inc.)
XiaoFeng Wang (Indiana University, USA)
Michael Wellman (University of Michigan, USA)

Table of Contents

Invited Talk: Network Congestion Games	1
<i>Paul G. Spirakis (Univ. of Patras)</i>	
Automated Design of Multistage Mechanisms	2
<i>T. Sandholm, V. Conitzer (Carnegie Mellon Univ.), C. Boutilier (Univ. of Toronto)</i>	
Resource Selection Games with Unknown Number of Players	12
<i>I. Ashlagi, D. Monderer, M. Tennenholtz (Technion)</i>	
Creating Incentives for Honest Rating Submission by Rating the Raters	20
<i>J. H. Lepler (Cambridge Univ.)</i>	
Scalability and Robustness of a Market-Based Network Resource Allocation System	30
<i>N. Haque, N. R. Jennings, L. Moreau (Univ. of Southampton)</i>	
An Auction-Based Pricing Scheme for Bandwidth Sharing with History Dependent Utility Functions	40
<i>P. Maille (GET/ENST-Bretagne), B. Tuffin (IRISA/INRIA Rennes)</i>	

Invited Talk

Network Congestion Games

Paul G. Spirakis

Abstract

We review here the beautiful field of congestion games and especially the case of such games played in networks. We start from classical results such as the equivalence of unweighted congestion games with exact potential games (Rosenthal's theorem). We discuss the relation of such games with the class PLS of the complexity of local search. We then move to the weighted case, where players have loads and they submit them into a network. We show some positive and negative results about existence of pure Nash Equilibria in such games, demonstrating how crucial are the latency functions of the edges. We then move to the topic of the cost of anarchy and present some recent results.

Automated Design of Multistage Mechanisms*

Tuomas Sandholm
Carnegie Mellon University
Computer Science Department
Pittsburgh, PA 15213, USA
sandholm@cs.cmu.edu

Vincent Conitzer
Carnegie Mellon University
Computer Science Department
Pittsburgh, PA 15213, USA
conitzer@cs.cmu.edu

Craig Boutilier
University of Toronto
Department of Computer Science
10 King's College Road
Toronto, Ontario, Canada M5S 3G4
cebly@cs.toronto.edu

Abstract

Mechanism design is the study of preference aggregation protocols that work well in the face of self-interested agents. We present the first general-purpose techniques for automatically designing multistage mechanisms. These can reduce elicitation burden by only querying agents for information that is relevant given their answers to previous queries. We first show how to turn a given (e.g., automatically designed using constrained optimization techniques) single-stage mechanism into the most efficient corresponding multistage mechanism given a specified elicitation tree. We then present greedy and dynamic programming (DP) algorithms that will determine the elicitation tree (optimal in the DP case). Next, we show how the query savings inherent in the multistage model can be used to design the underlying single-stage mechanism to maximally take advantage of this approach. We illustrate all of these techniques on an optimal auction example. Finally, we present negative results on the design of multistage mechanisms that do not correspond to dominant-strategy single-stage mechanisms: an optimal multistage mechanism in general has to randomize over queries to hide information from the agents.

1. Introduction

In multiagent settings, often an *outcome* (e.g., presidents, joint plans, allocations of resources) must be chosen based on the preferences of a set of agents. Since the preference aggregator generally does not know these preferences *a priori*, the agents must report their preferences to the aggregator. Unfortunately, an agent may have an incentive to misreport its preferences in order to mislead the aggregator into selecting an outcome that is more desirable to the agent than the outcome that would be selected if the agent revealed its preferences truthfully.

Mechanism design is concerned with the creation of preference aggregation rules that lead to good outcomes in spite of such strategic behavior by agents. Classical mechanism design provides some general mechanisms, which, under certain assumptions, satisfy some notion of nonmanipulability and maximize some objective. Such mechanisms do not rely on (even probabilistic) information about the agents' preferences (e.g., the Vickrey-Clarke-Groves (VCG) mechanism [24, 7, 14]), or can be easily applied to any distribution over preferences [13, 1, 18, 17]. However, these general algorithms only work in restricted settings (e.g., requiring the possibility of side payments), and may not reflect the designer's objectives.

Recently, *automated mechanism design (AMD)* has been proposed as a means to design mechanisms automatically for the setting at hand [8, 10, 12, 4]. This constrained-optimization based approach produces optimal special-purpose mechanisms even in settings for which no good general mechanisms are known, or for which an impossibility result precludes the existence of good gen-

*This material is based upon work supported by the National Science Foundation under ITR grants IIS-0121678 and IIS-0427858, and a Sloan Fellowship.

eral mechanisms for the class of instances (but not the existence of a good mechanism for the specific instance at hand). However, all prior work on general-purpose AMD has focused on single-stage mechanisms, in which all agents reveal their preferences completely and simultaneously. This is problematic for several reasons. First and foremost, agents may need to invest computational (or other) resources to determine their preferences over outcomes (e.g., when bidding on trucking tasks, an agent needs to solve, for each subset of tasks, a complex vehicle-routing problem). Second, the agents lose all privacy about their preferences. Third, it can require a large amount of communication. While the third reason applies only when the space of possible preferences is large, the first two reasons are significant even in very small examples.

Much of this computation, communication, and privacy loss is unnecessary when certain aspects of an agent’s preferences have no influence on the final outcome. For instance, if a second agent can perform a task at much lower cost than a first, we need not determine precisely how sub-optimal assigning the task to the first agent is. Unfortunately, single-stage mechanisms cannot take advantage of this: we cannot *a priori* rule out the need to know the first agent’s precise preferences for the task—this only becomes apparent after receiving information from the second.

Our solution is to use *multistage* mechanisms, where the aggregator queries the agents about certain aspects of their preferences, and chooses the next query to ask (and who to ask it of) based on answers to earlier queries. In a non-game-theoretic setting, a move to multistage protocols can yield an exponential savings in bits communicated [15]. In mechanism design settings, such a move can yield an exponential savings in communication and the aggregator’s computation [11]. In combinatorial auctions, the savings in communication can even be *super-exponential* [3].

Multistage mechanisms have been *manually* designed for several applications, such as voting [9], 1-object auctions (e.g. [6]), and combinatorial auctions (see reviews by [21, 19]). In this paper, we introduce the first general-purpose techniques for *automated* design of multistage mechanisms.¹ We adopt a specific approach: we first design a single-stage mechanism using existing techniques, and then convert it into a multistage mechanism in various ways. We also show how to design the underlying single-stage mechanism to maximally take advantage of this approach, so that the approach does not come at a loss.

¹Previous work has studied the design of multistage mechanisms in a strategic multi-party computation setting [23, 2], but the issues in that setting are very different from those that we study in this paper. For example, the key strategic issue in that work is that an agent may be tempted not to invest the effort necessary to determine its private information.

2. The model

2.1. Automated design of single-stage mechanisms

In this subsection, we review the relevant definitions and results from the single-stage AMD literature. In a single-stage AMD setting, we are given: 1) a finite set of outcomes O ;² 2) a finite set of N agents; 3) for each agent i , (a) a finite set of types Θ_i , (b) a probability distribution γ_i over Θ_i (in the case of correlated types, there is a single joint distribution γ over $\Theta_1 \times \dots \times \Theta_N$), and (c) a utility function $u_i : \Theta_i \times O \rightarrow \mathbb{R}$;³ 4) an objective function $g : \Theta_1 \times \dots \times \Theta_N \times O \rightarrow \mathbb{R}$ whose expectation the designer wishes to maximize. Possible designer objectives are many (e.g., *social welfare*, or maximizing the sum of agent utilities for the chosen outcome).

By the *revelation principle* [16], we can restrict attention to *truthful, direct revelation mechanisms*, where agents report their types directly and never have an incentive to misreport them. In general, mechanisms may choose the outcome randomly. Thus, a mechanism consists of a distribution selection function $p : \Theta_1 \times \dots \times \Theta_N \rightarrow \Delta(O)$, where $\Delta(O)$ is the set of probability distributions over O . A mechanism is a *dominant strategy mechanism* if truthtelling is optimal regardless of what other agents report. In other words, for any agent i , type vector $(\theta_1, \dots, \theta_i, \dots, \theta_N)$, and alternative report $\hat{\theta}_i \in \Theta_i$, we have $E_{o|\theta_1, \dots, \theta_i, \dots, \theta_N} u_i(\theta_i, o) \geq E_{o|\theta_1, \dots, \hat{\theta}_i, \dots, \theta_N} u_i(\theta_i, o)$. If telling the truth is optimal only *given* that the other agents are truthful, we have a *Bayes-Nash equilibrium (BNE)* mechanism. That is, in a BNE mechanism, for any i , $\theta_i \in \Theta_i$, and alternative report $\hat{\theta}_i \in \Theta_i$, we have $E_{(\theta_1, \dots, \theta_{i-1}, \theta_{i+1}, \dots, \theta_N) | \theta_i} E_{o|\theta_1, \dots, \theta_i, \dots, \theta_N} u_i(\theta_i, o) \geq E_{(\theta_1, \dots, \theta_{i-1}, \theta_{i+1}, \dots, \theta_N) | \theta_i} E_{o|\theta_1, \dots, \hat{\theta}_i, \dots, \theta_N} u_i(\theta_i, o)$.⁴

Given that the mechanism is allowed to choose the outcome at random, the problem of designing an optimal single-stage mechanism can be solved in polynomial time (given that the number of agents is constant) using linear programming [8]. The decision variables of that linear program are the following: for every type vector θ and every outcome o , there is a decision variable $p(\theta, o)$ that de-

²Payments to/from agents can be part of the outcome.

³The utility function is parameterized by type; while the u_i are common knowledge, the types encode (private) preferences [16]. The restriction to a finite type space is somewhat limiting. However, continuous spaces can be handled via suitable discretization of the type space. The discretization can be fixed in advance with an analysis of its impact on incentives and efficiency (as in recent research on limited revelation auctions [5]). Or, it may be optimized within the AMD model itself; this latter point is the subject of current research.

⁴In settings where participation is voluntary, the AMD formulation also includes participation (or *individual rationality*) constraints: no agent is worse off participating in the mechanism than not. Techniques for handling them in single-stage AMD can be applied to our multistage case without modification.

termines the probability of that outcome given that type vector. It is straightforward to check that the incentive compatibility constraints above, as well as the expectation of the objective, are linear functions of these variables, which gives us the linear program. Generating and solving this linear program is all that is required to have a basic approach to automatically designing single-stage mechanisms, and it is in fact the approach that we use in this paper to generate single-stage mechanisms.

2.2. Automated design of multistage mechanisms

In multistage AMD, the input includes—in addition to the input for single-stage AMD—a set of queries Q and a set of answers A .⁵ Each query q is associated with a particular agent i (of whom q would be asked),⁶ and the answer that the agent would give to q (when answering truthfully) is given by the function $a : Q \times \Theta_i \rightarrow A$, where $a(q, \theta_i)$ is i 's truthful answer to query q when i 's type is θ_i . This implies that there is only one truthful response to any $q \in Q$; thus, each query partitions the agent's type space. Upon receiving answer a to q from agent i , the mechanism can infer (assuming truthfulness) i 's type is in $\{\theta_i \in \Theta_i : a(q, \theta_i) = a\}$.

A multistage mechanism M corresponds to a given single-stage mechanism S if, for each type vector θ reported by the agents, both M and S choose each outcome o with the same probability. Suppose M corresponds to some S where truth-telling is dominant. It is not hard to see that M has truth-telling as an *ex-post* equilibrium, regardless of the results of previous queries revealed. That is, truth-telling is optimal (regardless of an agent's beliefs) whenever all other agents answer queries truthfully.⁷ This implies that we never need to randomize over query choice (though this no longer holds if S is not a dominant-strategy mechanism; see Section 7).

For these reasons, apart from Section 7, we focus exclusively on multistage mechanisms that correspond to dominant-strategy single-stage mechanisms. Thus, we can restrict ourselves to mechanisms that select the next query deterministically based on answers to prior queries; moreover, we need not worry about incentives.

⁵Assuming a single answer set (rather than distinct A_q for each query q) comes w.l.o.g. One set of interest is $A = \{\text{yes}, \text{no}\}$.

⁶In this paper we will restrict our attention to the case where we query one agent at a time; however, our approach is easily extended to settings where we query multiple agents at the same time. We note, however, that querying agents one at a time leads to the largest possible savings in the number of queries.

⁷Ex post is weaker than dominant strategies, but stronger than BNE. Note that even if S is a dominant-strategy mechanism, M need not be: if an agent makes her answer dependent on the history of queries asked, another agent may have an incentive to lie about her type in order to influence which queries the former is asked.

Under these restrictions, a *multistage mechanism* is defined by: 1) a tree with nodes V and edges E ; 2) for each internal (non-leaf) node v , an agent i and a query q to that agent; 3) a one-to-one correspondence between possible answers to the query at node v and children of node v ; 4) for each node v and outcome o , a probability that, given that we reach v , we stop asking queries and choose outcome o . (In the case where v is a leaf, these probabilities must sum to one.) An *elicitation tree* is a multistage mechanism without outcome probabilities. We denote by I_v the *information set* at node v (i.e., the set of type vectors consistent with the answers that lead to v). We generally assume an elicitation tree is *complete*: I_l is a single type vector for any leaf l .⁸

We study several variants of multistage AMD. First, we may either start from a given single-stage mechanism (e.g., computed by single-stage AMD software) and turn it into a corresponding multistage mechanism, or we may impose no constraints on the single-stage version of the multistage mechanism. Second, we may either assume that the elicitation tree (hence the query order) is given beforehand, or we may impose no constraints on it.

3. A small example

In this section, we illustrate various notions for automatically designing multistage mechanisms using a single, simple example. Suppose a divorcing couple jointly owns a painting, and an arbitrator has to decide the fate of the painting. There are 5 options: (1) the husband keeps the painting; (2) the wife keeps it; (3) the painting remains jointly owned, but is hung in a museum; (4) it is cut into pieces which are given to the husband; and (5) it is cut up with pieces given to the wife. The husband and wife each have two possible types: type L is associated with relative indifference toward the painting, and type H with deep attachment. Each has type L with probability 0.8 and type H with probability 0.2. To maximize social welfare, the arbitrator would like to give the painting to whomever cares for it more; but since a party who cares little would prefer having it over not, the arbitrator must design appropriate incentives to ensure truthful reporting. The utility function for each party is the “same.” Keeping the painting gives utility 2 (type L) or 100 (H). The other party getting the painting gives utility 0 (for either type). The museum outcome gives utility 1.5 (L) or 40 (H). Receiving pieces gives utility -9 while not even getting the pieces gives utility -10 .

⁸This does not imply that the mechanism will ask all queries and uniquely determine a type vector: the concrete outcome probabilities, specifically, the possibility of terminating at an interior node will typically preclude this.

(for either type).⁹

Our goal is to find a dominant-strategy (possibly randomized) mechanism (without payments) that maximizes expected social welfare. First we find the optimal single-stage mechanism. Solving this example using the methodology described in Subsection 2.1 yields the following randomized mechanism (the probabilities are rounded):

	wife L	wife H
husband L	Museum	0.96 Wife keeps; 0.04 Husband gets pieces
husband H	0.96 Husband keeps; 0.04 Wife gets pieces	0.47 Husband keeps; 0.40 Wife keeps; 0.13 Wife gets pieces

In spite of the symmetry between the husband and the wife, the mechanism is asymmetric. Of course, other optimal solutions exist (e.g., where the roles of husband and wife are interchanged).

Now we consider how to turn this single-stage mechanism into a corresponding multistage mechanism (i.e., with the same outcome probabilities). First, suppose the elicitation tree is given, with the wife's type elicited first. Fig. 1 shows the optimal multistage mechanism. (Why this is so will become apparent.) This mechanism saves one query with probability $0.2 \cdot 0.4 = 0.08$.

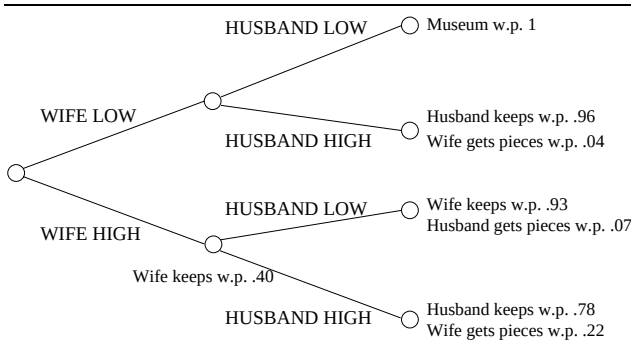


Figure 1. The optimal elicitation tree given the single-stage mechanism and given that the wife is queried first. When an internal node has a probability-outcome pair associated with it, we terminate early at that node with that probability, with that outcome; with the remaining probability, we move on to the next query.

If the elicitation tree (query order) is not fixed, the optimal mechanism is that given in Fig. 2. It turns out that

⁹This problem has some similarity to King Solomon's dilemma; however, when that dilemma is discussed in the economics literature [20], it is assumed that there is only one rightful mother, and both women know who it is—unlike our problem, where the agents do not know each others' types.

greater savings can be obtained by eliciting the husband's type first: this mechanism saves a query with probability $0.2 \cdot (0.04 + 0.47) = 0.10$. (This is due to the asymmetry of the single-stage mechanism from which we are starting.)

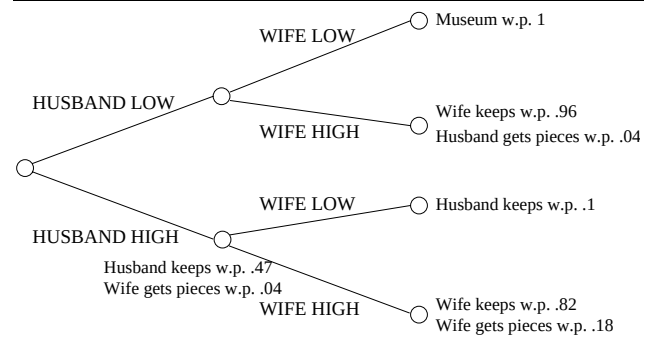


Figure 2. The optimal mechanism when asking the husband first.

Suppose queries to the husband are slightly more expensive than those to the wife, so that we would rather save on husband queries (unlike the previous mechanism). If we allow a different optimal single-stage mechanism, namely the analog of the one above with the husband and wife roles switched (which remains optimal due to the problem symmetry), then the optimal multistage mechanism that corresponds to this saves a query to the husband (rather than the wife) with probability 0.10, giving greater cost savings.

Finally, if we are willing to sacrifice optimality of the single-stage mechanism to obtain greater query savings, this may again change the mechanism. For example, if we make the cost of querying sufficiently large, it will be optimal to not ask any queries, and always choose the same outcome.

One interesting additional motivation for automatically designing multistage mechanisms is that the tree-based representation of a multistage mechanism may be *easier to understand* for a human than the tabular form of a single-stage mechanism—especially if the tree is relatively small.

4. Converting a single-stage mechanism into a multistage mechanism

In this section we develop methods for converting a given (e.g., automatically designed) single-stage mechanism into an equivalent multistage mechanism which saves on elicitation costs. In Subsection 4.1 we develop methods for the case where the elicitation tree (query order) is given. In Subsection 4.2 we generalize the approach to the case where the elicitation tree is not given, but can be chosen endogenously.

4.1. Given elicitation tree

We first solve the simplest of our problems: converting a single-stage mechanism into the most efficient multistage mechanism for a given elicitation tree. This problem can be motivated by considering exogenous constraints on query order (e.g., agents available at different times, or when the optimal ordering is readily available). More importantly, this setting serves as a stepping stone to more general techniques below. Our key technique is to “propagate up” probability from the leaves to internal nodes where this is possible.

Lemma 1 *Let multistage mechanism M correspond to single-stage mechanism S . Suppose that for some internal node v in the elicitation tree (with exit prob. e_v) and outcome o , all the leaves of the subtree T_v rooted at v assign a probability of at least $p > 0$ to outcome o . Then the following modification M' of M corresponds to S : (1) At node v , exit with o with probability $(1 - e_v)p$; (2) Subtract p from the probability assigned to o at each leaf of T_v ; (3) Divide all the outcome probabilities at leaves T_v by $1 - p$.*

Proof: Consider the probability $p(\theta, o')$ that outcome o' will be selected given type vector θ in M' . If θ does not lead to v , clearly $p(\theta, o')$ is the same in M and M' ; so assume that it does. If $o' = o$ (the outcome that we exit early with), then the probability of selecting o' at v is now the early-exit probability p , plus the probability that we do not exit early but choose outcome o' later, which is $(1 - p)(p(\theta, o')^{old} - p)/(1 - p) = p(\theta, o')^{old} - p$. Hence the total probability is $p(\theta, o')^{old}$; i.e., it did not change. If $o' \neq o$, then the probability of selecting o' at v is the probability that we do not exit early with o and choose outcome o' later, which is $(1 - p)(p(\theta, o')^{old})/(1 - p) = p(\theta, o')^{old}$. Hence for any θ , M and M' select o' with the same probability. ■

We note that the ability to propagate probability up in this manner even when the distributions at the leaves are not identical makes this different from the standard framework in communication complexity theory. (In addition, we may have a restricted query language, and we have a prior distribution over the inputs.)

If we propagate up as much probability as possible, we obtain the optimal mechanism (for a given S and tree):

Theorem 1 *Suppose we apply Lemma 1 until we can apply it no further (that is, until for any internal node v and outcome o , there is at least one leaf of the subtree rooted at v that assigns probability 0 to o). Then the resulting multistage mechanism saves the most queries (or, in the case of different query costs, the greatest query cost) among multi-*

stage mechanisms corresponding to the given single-stage mechanism and the given elicitation tree.

Proof: Any mechanism with the same elicitation tree that saves more queries (or greater query cost) must, for some node v , have a greater probability of exiting early at or before v than the mechanism generated by applying Lemma 1. It follows that for at least one outcome o , the former mechanism has a probability of exiting early at or before v with this o that is greater than $\sum_{o \in O} \min_{\theta \in I_v} p(\theta, o)$. But then, there is some $\theta \in I_v$ such that $p(\theta, o)$ is smaller than the probability of exiting early at or before v with outcome o . So the mechanism does not correspond to the given single-stage mechanism. ■

As an example, we derive the mechanism of Fig. 2. We start from a mechanism that saves no queries (Fig. 3).

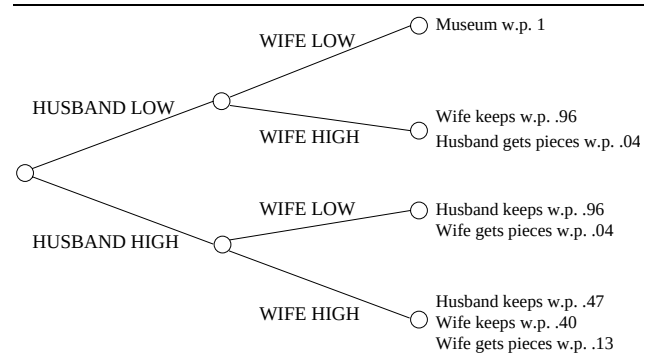


Figure 3. Multistage mechanism that saves no queries at all.

At the node after the husband reports “high”, the husband keeps the painting with $p \geq .47$ in all subsequent leaves. So we can propagate this probability up (Fig. 4). At the same

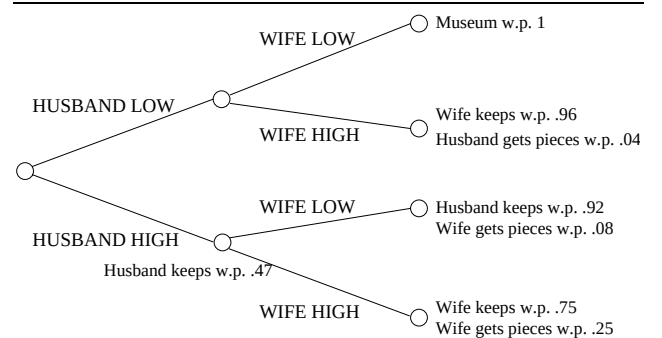


Figure 4. Some probability propagated up.

node, the wife gets the pieces of the painting with $p \geq .08$

in all subsequent leaves. Propagating this up results in the mechanism of Fig. 2.

The following corollary characterizes the probability of exiting early at or before a given node. This will be helpful in our use of the “propagating probabilities up” technique within all of the algorithms discussed later in the paper.

Corollary 1 *In a multistage mechanism that saves a maximum number of queries, for any type vector θ such that node v will be reached if the mechanism does not exit early, the probability that we will reach v and not exit early at v is $1 - \sum_{o \in O} \min_{\theta \in I_v} p(\theta, o)$. Hence, given that we have not exited early at or before node v , and we transition from node v to node w , the probability of exiting early at node w is $1 - (1 - \sum_{o \in O} \min_{\theta \in I_w} p(\theta, o)) / (1 - \sum_{o \in O} \min_{\theta \in I_v} p(\theta, o))$.*

4.2. Endogenously determined elicitation tree

In this section we develop methods for converting a single-stage mechanism into a multistage one, *without constraints on the elicitation tree (query order)*. We first provide a greedy algorithm, and show two ways in which it can “fail” (i.e., yield an arbitrarily small fraction of the query savings available). We then give an optimal dynamic program.

4.2.1. Greedy algorithm Our greedy algorithm chooses the query at each stage so as to maximize the probability of being able to exit immediately after this query given the preceding queries and responses. Letting $U(I, q, a_q)$ denote the information state that results from being in information state I and then receiving answer a to query q , we define the algorithm as follows.

Definition 1 *The greedy algorithm chooses the query to ask at node v from the set*

$$\arg \max_{q \in Q} \sum_{a \in A} P(a|I_v, q) \sum_{o \in O} \min_{\theta \in U(I_v, q, a)} p(\theta, o).$$

The greedy algorithm does what we intend:

Theorem 2 *The greedy algorithm chooses a query that maximizes the probability of exiting immediately after it.*

Proof: By Corollary 1, we know that if we are currently at node v and do not exit early, and we transition to node w , then the probability of exiting early at node w is $1 - \frac{\sum_{o \in O} \min_{\theta \in I_w} p(\theta, o)}{1 - \sum_{o \in O} \min_{\theta \in I_v} p(\theta, o)}$. Thus, if we ask query $q \in Q$ at node v , the probability of exiting immediately after q is

$$\sum_{a \in A} P(a|I_v, q) \left(1 - \frac{\sum_{o \in O} \min_{\theta \in U(I_v, q, a)} p(\theta, o)}{1 - \sum_{o \in O} \min_{\theta \in I_v} p(\theta, o)} \right) = 1 - \frac{1}{1 - \sum_{o \in O} \min_{\theta \in I_v} p(\theta, o)} \left(1 - \sum_{a \in A} P(a|I_v, q) \sum_{o \in O} \min_{\theta \in U(I_v, q, a)} p(\theta, o) \right).$$

Choosing $q \in$

Q to maximize this expression is equivalent to choosing $q \in Q$ to maximize $\sum_{a \in A} P(a|I_v, q) \sum_{o \in O} \min_{\theta \in U(I_v, q, a)} p(\theta, o)$, which is exactly what the greedy algorithm does. ■

Theorem 3 *The greedy algorithm chooses the query for node v in time $O(|Q| \cdot |A| \cdot |O| \cdot |\Theta|)$.*

Unfortunately, the greedy algorithm can be arbitrarily far from optimal (even when all queries have equal cost):

Proposition 1 *There exist single-stage mechanisms S for which the greedy algorithm achieves only an arbitrarily small fraction of the possible query savings (even when S is deterministic, there are only three players, two types per player, and three outcomes; alternatively, even when priors over types are uniform, there are only three players, two types per player, and five outcomes).*

4.2.2. Dynamic programming algorithm Unlike the greedy algorithm, the dynamic program must build the entire tree. The program works by computing, for every possible information state I , the minimum possible expected number of queries $n(I)$ from that point on, given that we have not exited early. As before, let $U(I, q, a)$ be the information state that results from receiving answer a to q at I . Let $e(I, q, a)$ be the probability of exiting immediately after receiving answer a to q at I , given that we did not exit early at I . By Corollary 1, we can compute $e(I, q, a)$ as $1 - \frac{1 - \sum_{o \in O} \min_{\theta \in U(I, q, a)} p(\theta, o)}{1 - \sum_{o \in O} \min_{\theta \in I} p(\theta, o)}$. We obtain the recurrence

$$n(I) = \min_{q \in Q} c(q) + \sum_{a \in A} P(a|I, q) (1 - e(I, q, a)) n(U(I, q, a))$$

Using the fact that $n(\{\theta\}) = 0$ for every type vector θ , we use this recurrence to compute the value of $n(I)$ for every I , starting with the small I and working up to larger ones.

Theorem 4 *The dynamic programming algorithm computes the value of $n(I)$ for all I in time $O(|Q| \cdot |A| \cdot |O| \cdot |\Theta| \cdot 2^{|\Theta|})$.*

We can retrieve the optimal multistage mechanism from this as follows: when we arrive at information state I and do not exit early, choose a query from

$$\arg \min_{q \in Q} \sum_{a \in A} P(a|I, q) (1 - e(I, q, a)) n(U(I, q, a)).$$

5. Designing optimal multistage mechanisms

So far we have discussed how a given single-stage mechanism can be converted into an equivalent multistage mechanism. Here we will no longer take the single-stage design as a constraint. We develop a method for designing the

single-stage mechanism in such a way that we get large savings in queries when we transform it into a multistage mechanism using the techniques described earlier. We focus on the case where the elicitation tree (query order) is given. It turns out that, using Corollary 1, we can directly integrate the eventual query savings into the linear programming formulation for AMD described in Subsection 2.1.

We say that node v is *on the elicitation path* for type vector θ if θ would lead us to ask the query at v (given that we do not exit early). For every internal node v in the tree, we add a term to the AMD objective (which maximizes the designer's objective) that indicates the probability of saving the query corresponding to this node.¹⁰ (We say that we *save* the query corresponding to v when v is on the elicitation path, but we exit early at or before v .) Thus, the term in the objective for v is $c(v)P(v)e(v)$ where $c(v)$ is the cost of the query at node v , $P(v)$ is the probability of v being on the elicitation path, and $e(v)$ is the probability that we will exit early at or before v , given that v is on the elicitation path. $P(v)$ is a constant, but $e(v)$ is a variable that depends on how we set the outcome probabilities for the leaves. Specifically, by Corollary 1, we know that $e(v) = \sum_{o \in O} \min_{\theta \in S_v} p(\theta, o)$. The min operator is not linear, so we cannot add this expression to the LP objective directly. We work around this by letting $e(v) = \sum_{o \in O} e(v, o)$, where $e(v, o)$ is the probability of exiting early at or before v with outcome o , given that v is on the elicitation path. Then, for every $o \in O$ and $\theta \in S_v$, we add the constraint $e(v, o) \leq p(\theta, o)$.

Because linear programs can be solved to optimality in polynomial time, and the formulation above is polynomial in the number of outcomes and the number of types per agent (but not in the number of agents), the following theorem follows immediately:

Theorem 5 *The extension of the single-stage AMD formulation described above computes the optimal multistage mechanism for the given elicitation tree, taking query costs into account, in time polynomial in the number of outcomes and the number of types per agent (but not in the number of agents).*

This also begets an (inefficient) algorithm for generating the optimal multistage mechanism when neither the single-stage mechanism nor the elicitation tree is given: apply the above algorithm to every possible elicitation tree.

¹⁰Suitable scaling to ensure commensurability with the designer's objective is straightforward; however, this does assume query costs can be accounted for additively.

6. Example application: optimal auctions

In this section, we demonstrate how our techniques can be applied to a simple auction example. In this auction, a single item is for sale. There are two bidders, A and B ; the prior over each bidder's valuation for the item is uniform over $\{0, 1, 2, 3\}$. Our objective is to maximize expected revenue.

We first used the standard (single-stage) automated mechanism design software to generate an optimal single-stage dominant strategies mechanism for this setting. This produced the following mechanism (which is effectively the Myerson auction [18]):

	B bids 0	B bids 1	B bids 2	B bids 3
A bids 0	item not sold	item not sold	B wins, pays 2	B wins, pays 2
A bids 1	item not sold	item not sold	B wins, pays 2	B wins, pays 2
A bids 2	A wins, pays 2	A wins, pays 2	A wins, pays 2	B wins, pays 3
A bids 3	A wins, pays 2	A wins, pays 2	A wins, pays 2	A wins, pays 3

We then focused our attention on turning this single-stage mechanism into a multistage mechanism. We allowed only queries of the form "Is your valuation for the item at least k ?" We observe that no matter which query is asked first, there is no chance of exiting after the first query. Hence, the greedy algorithm from Subsection 4.2 is underdetermined (and presumably will not perform very well). Instead, we used the dynamic programming algorithm from Subsection 4.2 to obtain the optimal elicitation tree for this single-stage mechanism. This produced the multistage mechanism in Figure 5.

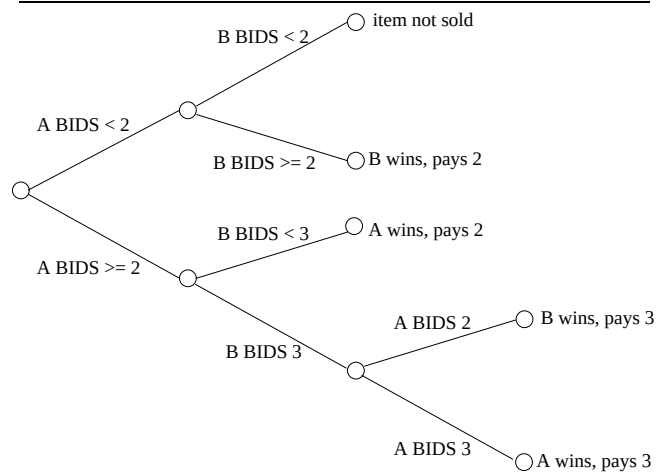


Figure 5. Single-stage auction converted to multistage.

We next studied whether any query gains could be made by changing the underlying single-stage mechanism while keeping the elicitation tree fixed, as described in Section 5.

We first placed a cost of 0.001 on each query. This produced the multistage mechanism in Figure 6.

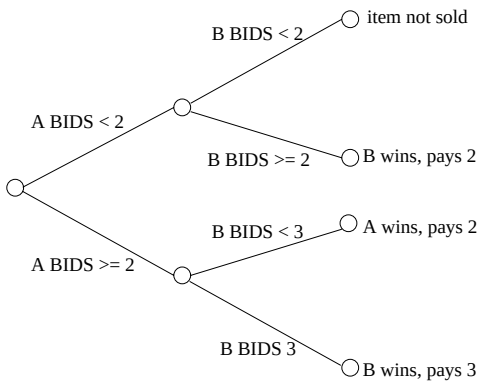


Figure 6. Underlying single-stage auction changed to save queries.

The only difference between this mechanism and the previous is how the tie is broken when both bidders bid 3. Therefore, this mechanism attains the same expected revenue as the previous mechanism. That is, the additional query savings obtained by this multistage mechanism come at no cost to the original objective.

Finally, we placed a cost of 0.5 on each query. This produced the multistage mechanism in Figure 7.

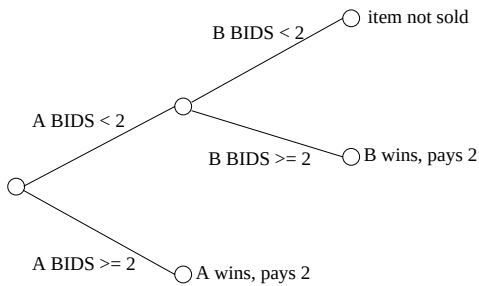


Figure 7. Underlying single-stage auction changed to save even more queries.

Effectively, this mechanism gives bidder *A* a take-it-or-leave-it offer of 2 for the item; if bidder *A* does not take this offer, the mechanism makes the same offer to bidder *B*. (Mechanisms that consist of sequences of take-it-or-leave-it offers have been studied systematically [22].) This mechanism in fact has lower expected revenue, but this loss in expected revenue is outweighed by the query savings that are obtained.

7. Mechanisms without dominant strategies

So far, we have restricted our study to multistage mechanisms whose single-stage correspondents have truth-telling as a dominant strategy. As discussed, this is helpful because in such multistage mechanisms, telling the truth is an ex-post equilibrium, so we need not worry that information revealed to agents by the mechanism will introduce strategic behavior. Nevertheless, we may also be interested in converting single-stage mechanisms that do not have dominant strategies, such as BNE mechanisms, to multistage mechanisms (e.g., because such mechanisms can achieve a higher objective value than dominant-strategy mechanisms).

Here we present initial results on converting BNE mechanisms into multistage mechanisms. These results are negative: they show that restricting ourselves to particular natural classes of multistage mechanisms may come at a loss of optimality. Thus, to design optimal multistage mechanisms, we need to search a broader space of mechanisms.

Proposition 2 *Even when the primary objective is social welfare and we use BNE as our solution concept, there exist settings in which immediately revealing the result of every query incurs a loss in objective value.*

The next result that we establish is that restricting ourselves to mechanisms that always chooses the next query deterministically can come at a loss.

Proposition 3 *There exist settings in which: 1) The primary objective is social welfare; 2) The optimal single-stage BNE incentive compatible mechanism is unique; 3) The unique optimal (in terms of query savings) elicitation tree to ask the queries for this mechanism is not (even BNE) incentive compatible; 4) There exists an elicitation tree for this mechanism that randomizes over the next query selected, is (BNE) incentive compatible, and has almost the same query savings as the optimal elicitation tree (and thus strictly greater query savings than any deterministic (BNE) incentive-compatible elicitation tree for this mechanism).*

Proof: Let there be two agents, 1 and 2; let agent 1 have type set $\Theta_1 = \{\theta_1^1, \theta_1^2, \theta_1^3\}$ and let agent 2 have type set $\Theta_2 = \{\theta_2^1, \theta_2^2\}$. Let the outcome set be $O = \{o_1, o_2, o_3, o_4\}$. Let the utility functions be as follows:

$$\begin{aligned} u_1(\theta_1^1, o_1) &= 0, u_1(\theta_1^2, o_1) = 0, u_1(\theta_1^3, o_1) = 0, \\ u_2(\theta_2^1, o_1) &= 0, u_2(\theta_2^2, o_1) = 0; \\ u_1(\theta_1^1, o_2) &= 3, u_1(\theta_1^2, o_2) = 1 + \epsilon, u_1(\theta_1^3, o_2) = -1, \\ u_2(\theta_2^1, o_2) &= 1, u_2(\theta_2^2, o_2) = -4; \\ u_1(\theta_1^1, o_3) &= -4, u_1(\theta_1^2, o_3) = 1, u_1(\theta_1^3, o_3) = 3, \\ u_2(\theta_2^1, o_3) &= 2, u_2(\theta_2^2, o_3) = 3; \\ u_1(\theta_1^1, o_4) &= -5, u_1(\theta_1^2, o_4) = -5, u_1(\theta_1^3, o_4) = 4, \\ u_2(\theta_2^1, o_4) &= -5, u_2(\theta_2^2, o_4) = 4. \end{aligned}$$

The unique social-welfare maximizing outcome in each case is: $o(\theta_1^1, \theta_2^1) = o_2$, $o(\theta_1^1, \theta_2^2) = o_1$, $o(\theta_1^2, \theta_2^1) = o_3$, $o(\theta_1^2, \theta_2^2) = o_3$, $o(\theta_1^3, \theta_2^1) = o_3$, $o(\theta_1^3, \theta_2^2) = o_4$.

Let the probability distributions over types be as follows: $\gamma_1(\theta_1^1) = 0.4$, $\gamma_1(\theta_1^2) = 0.1$, and $\gamma_1(\theta_1^3) = 0.5$. $\gamma_2(\theta_2^1) = \gamma_2(\theta_2^2) = 0.5$. Choosing the social welfare maximizing outcome in every case is BNE incentive compatible, so this is the unique optimal single-stage BNE mechanism. (We omit the proof of this fact due to space constraint.)

Let the cost of a query be a secondary objective to the social welfare objective (or let the cost of a query be very small), so that we still want to implement a mechanism that always chooses the social welfare maximizing outcome. Let there be three queries: the single query to agent 2, and two queries to agent 1: *Is your type θ_1^1 ?* and *Is your type θ_1^3 ?* No single query is enough to determine the outcome for the optimal single-stage mechanism above. Disregarding strategic considerations, the unique optimal multistage mechanism corresponding to the optimal single-stage mechanism is:

Ask agent 2 for its type first;

If it is θ_2^1 , ask agent 1 *Is your type θ_1^1 ?*

If so, choose o_2 , otherwise choose o_3 ;

If it is θ_2^2 , ask agent 1 *Is your type θ_1^3 ?*

If so, choose o_4 , otherwise ask agent 1 *Is your type θ_1^1 ?*

If so, choose o_1 , otherwise choose o_3 .

This multistage mechanism needs all 3 queries only $0.5 \cdot 0.5 = 0.25$ of the time. Unfortunately, under this multistage mechanism, agent 1 has a (slight) incentive to lie when its true type is θ_1^2 , and the first query to it is *Is your type θ_1^1 ?* In this case, agent 1 knows that agent 2 reported θ_2^1 , so falsely answering yes would give it $1 + \epsilon$ rather than 1. Any other multistage mechanism that corresponds to the optimal single-stage mechanism and does not randomize over queries will cost us significantly in queries. But, instead, we can also slightly modify the given multistage mechanism and with very small probability ask agent 1 *Is your type θ_1^1 ?* after agent 2 answers θ_2^2 . Now, when agent 1 is confronted with *Is your type θ_1^1 ?* as the first query while its true type is θ_1^2 , there is a slight chance that agent 2 answered θ_2^2 , in which case agent 1 does not want to answer yes and get o_1 rather than o_3 . Because the benefit of lying in the other case is so small, it is outweighed, and agent 1 will answer truthfully. All other queries are still answered truthfully—thus this mechanism is (BNE) truthful. ■

A potential alternative to randomization by the mechanism is to obtain the randomization from mixed (i.e., randomized) strategies of the agents in mechanisms that are not truthful direct-revelation mechanisms.

8. Conclusions and future research

We extended the constrained-optimization based techniques for automated mechanism design to the design of multistage mechanisms, allowing reduction in elicitation burden by querying agents sequentially, and only querying them for information that is relevant given previous query

responses. We focused primarily on the design of multistage mechanisms that correspond to dominant-strategy single-stage mechanisms, since these ensure truth-telling is an *ex-post* equilibrium (no matter what is revealed about other agents' answers). We described several techniques for converting single-stage mechanisms into multistage, both with and without fixed elicitation trees, and also showed how to augment single-stage AMD to produce single-stage mechanisms that can be maximally exploited in the conversion to multistage. We illustrated all of these techniques on an optimal auction example. Finally, we presented negative results on the design of multistage mechanisms that do not correspond to dominant-strategy single-stage mechanisms.

Future research includes developing techniques for designing multistage mechanisms that do not correspond to dominant-strategy single-stage mechanisms. We also hope to develop more efficient algorithms for optimal design when neither the single-stage mechanism nor the elicitation tree is given. Another direction is to extend techniques from machine learning to the query selection problem in our setting. Finally, we also plan to discover more efficient problem-specific techniques for the automated design of multistage mechanisms.

References

- [1] Kenneth Arrow. The property rights doctrine and demand revelation under incomplete information. In M Boskin, editor, *Economics and human welfare*. New York Academic Press, 1979.
- [2] Gal Bahar and Moshe Tennenholtz. Sequential-simultaneous information elicitation in multi-agent systems. In *Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence (IJCAI)*, Edinburgh, UK, 2005.
- [3] Avrim Blum, Jeffrey Jackson, Tuomas Sandholm, and Martin Zinkevich. Preference elicitation and query learning. *Journal of Machine Learning Research*, 5:649–667, 2004.
- [4] Andrew Blumberg and Abhi Shelat. Searching for stable mechanisms: Automated design for imperfect players. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pages 8–13, San Jose, CA, USA, 2004.
- [5] Liad Blumrosen and Noam Nisan. Auctions with severely bounded communication. In *FOCS*, pages 406–415, 2002.
- [6] Liad Blumrosen, Noam Nisan, and Ilya Segal. Multi-player and multi-round auctions with severely bounded communication. In *ESA*, pages 102–113, 2003.
- [7] Ed H. Clarke. Multipart pricing of public goods. *Public Choice*, 11:17–33, 1971.
- [8] Vincent Conitzer and Tuomas Sandholm. Complexity of mechanism design. In *Proceedings of the 18th Annual Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 103–110, Edmonton, Canada, 2002.
- [9] Vincent Conitzer and Tuomas Sandholm. Vote elicitation: Complexity and strategy-proofness. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pages 392–397, Edmonton, Canada, 2002.
- [10] Vincent Conitzer and Tuomas Sandholm. An algorithm for automatically designing deterministic mechanisms without payments. In *International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 128–135, New York, NY, USA, 2004.

- [11] Vincent Conitzer and Tuomas Sandholm. Computational criticisms of the revelation principle. In *The Conference on Logic and the Foundations of Game and Decision Theory (LOFT-04)*, Leipzig, Germany, 2004.
- [12] Vincent Conitzer and Tuomas Sandholm. Self-interested automated mechanism design and implications for optimal combinatorial auctions. In *Proceedings of the ACM Conference on Electronic Commerce (ACM-EC)*, pages 132–141, New York, NY, 2004.
- [13] Claude d’Aspremont and Louis-Andre Gérard-Varet. Incentives and incomplete information. *Journal of Public Economics*, 11:25–45, 1979.
- [14] Theodore Groves. Incentives in teams. *Econometrica*, 41:617–631, 1973.
- [15] E Kushilevitz and N Nisan. *Communication Complexity*. Cambridge University Press, 1997.
- [16] Andreu Mas-Colell, Michael Whinston, and Jerry R. Green. *Microeconomic Theory*. Oxford University Press, 1995.
- [17] Eric Maskin and John Riley. Optimal multi-unit auctions. In Frank Hahn, editor, *The Economics of Missing Markets, Information, and Games*, chapter 14, pages 312–335. Clarendon Press, Oxford, 1989.
- [18] Roger Myerson. Optimal auction design. *Mathematics of Operation Research*, 6:58–73, 1981.
- [19] David Parkes. Iterative combinatorial auctions. In Peter Cramton, Yoav Shoham, and Richard Steinberg, editors, *Combinatorial Auctions*, chapter 3. MIT Press, 2006.
- [20] Motty Perry and Philip J. Reny. A general solution to King Solomon’s dilemma. *Games and Economic Behavior*, 26:279–285, 1999.
- [21] Tuomas Sandholm and Craig Boutilier. Preference elicitation in combinatorial auctions. In Peter Cramton, Yoav Shoham, and Richard Steinberg, editors, *Combinatorial Auctions*, chapter 10. MIT Press, 2006.
- [22] Tuomas Sandholm and Andrew Gilpin. Sequences of take-it-or-leave-it offers: Near-optimal auctions without full valuation revelation. In *Agent-Mediated Electronic Commerce (AMEC) workshop*, Melbourne, Australia, 2003.
- [23] Rann Smorodinsky and Moshe Tennenholtz. Sequential information elicitation in multi-agent systems. In *Proceedings of the 20th Annual Conference on Uncertainty in Artificial Intelligence (UAI)*, Banff, Canada, 2004.
- [24] W Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *Journal of Finance*, 16:8–37, 1961.

Resource Selection Games with Unknown Number of Players

Itai Ashlagi Dov Monderer Moshe Tennenholtz
Technion–Israel Institute of Technology
Faculty of Industrial Engineering and Management
Haifa 32000, Israel
ashlagii@tx.technion.ac.il
dov@ie.technion.ac.il
moshet@ie.technion.ac.il

Abstract

In the context of pre-Bayesian games we analyze resource selection systems with unknown number of players. We prove the existence and uniqueness of a symmetric safety-level equilibrium in such games and show that in a linear model every player benefits from the common ignorance about the number of players. In order to analyze such games we generalize the theory of equilibrium in general pre-Bayesian games.

1. Introduction

There has been much interest in the recent years in work bridging computer science and game theory. Most of this work is concerned with the analysis of multi-agent systems, where the agents are selfish, and each agent attempts to maximize his own utility or minimize his own cost. Hence, it is natural that work in multi-agent systems adopted the notion of equilibrium as the type of solution concept to be discussed. On the other hand, when dealing with settings of incomplete information, work in computer science typically adopts non-Bayesian models, where qualitative decision making rules (without probabilistic assumptions about the environment) are used. However, in a multi-agent setup with incomplete information the non-Bayesian approach might severely restrict the type of analysis that may be carried out. Indeed, most previous work on games with incomplete information in computer science adopted dominant strategy equilibrium or ex-post equilibrium as the major solution concepts. This paper is part of a line of research that introduces non-Bayesian equilibrium concepts as appropriate solutions for dealing with incomplete information in multi-agent systems. One interesting property is that (as we show) the related (non-Bayesian) equilibrium always ex-

ist.¹ The main challenge remained is to consider concrete powerful settings, where the related (non-Bayesian) equilibrium concepts can be discussed, and lead to illuminating results. In this paper we deal with such setting, resource selection games, which is central to the computer science literature. We present an analysis of resource selection games with incomplete information, using the non-Bayesian approach, yielding several highly powerful and surprising results.

In a *resource selection system*, Γ , there is a set of m resources, $j \in \{1, \dots, m\}$. Each resource j is associated with a cost function $w_j : \{1, 2, \dots, \dots\} \rightarrow \mathbb{R}$, where $w_j(k)$ is the cost for every user of resource j if there are k users. Together with a set of n players a resource selection system defines a game in strategic form— a *resource selection game*, $\Gamma(n)$. The action set² of every player i in $\Gamma(n)$ is the set of resources M , and the cost of i depends, via the resource-cost functions on the resource she chooses and on the number of other players who choose this resource. Thus, resource selection games are special type of congestion games [22, 17]. A resource selection game is also referred to as a simple congestion game.³ In many situations the assumption that every player knows the number of players is not reasonable. The goal of the current paper is to analyze resource selection games with unknown number of players. One approach for analyzing such situations is the Bayesian approach, where it is assumed that the distribution of the random set of players is commonly known. In this approach one is interested in Bayesian equilibrium.⁴ The goal

¹ In a finite setup.

² When dealing with games in strategic form the choice set of a player is referred to as an action set or as a strategy set. We use "actions" rather than "strategies" because we keep the notion of strategy to describe the choice set of a player in "bigger" games.

³ Simple congestion games and their generalization to player-specific games were discussed in [21, 15, 8, 26, 10]. Simple congestion games were also discussed in the price of anarchy literature, e.g., [11, 2]

⁴ See [14, 13, 6] for such analysis in the context of auctions, and [18,

of this paper is to analyze resource selection games in the non-Bayesian setup in which the players do not have probabilistic information about the other players in the game. We will use the model of pre-Bayesian games.

We remark that games can be analyzed with either payoff functions or cost functions. Most of the general theory has been developed with payoff functions. However, congestion games have been mainly analyzed using cost functions. Translating results proved in one of the setups to the other setup is obvious in most cases.⁵ We follow the tradition of previous literature in the sense that the general theory is discussed with payoff functions, while resource selection games are discussed with cost functions.

In a classical pre-Bayesian game there is a fixed set of players $N = \{1, 2, \dots, n\}$, each of them is endowed with a set of actions, $x_i \in X_i$. There is a set of states $\omega \in \Omega$. The payoff of player i , $u_i(\omega, x_1, x_2, \dots, x_n)$ depends on the realized state, on her choice of action, x_i , and on the choices of all other players. However, the realized state is not known to the players. Every player receives a state-correlated signal, $t_i = \tilde{t}_i(\omega)$ on which she conditions her action. A pre-Bayesian game becomes a Bayesian game when a commonly known probability measure on the set of states is added to the system. In pre-Bayesian games one can deal with ex-post equilibrium. It is well-known that an ex-post equilibrium in a pre-Bayesian game is a Bayesian equilibrium for every choice of prior probability and vice versa. Indeed, the classical literature in economics/game theory actually discussed ex-post equilibrium in a particular Bayesian game, and defined it as a Bayesian equilibrium which is robust to changes in the prior probability. Only recently the concept of pre-Bayesian games have been explicitly defined.⁶ Unfortunately, ex-post equilibrium does not exist even in very simple pre-Bayesian games. A strategy of player i in a pre-Bayesian or Bayesian game is a function b_i that assigns an action (or a mixed action), $x_i = b_i(t_i)$ to every possible signal t_i . In all concepts of equilibrium it is assumed that a player knows the other players' strategies. Hence, when a player receives a signal, t_i , she is facing a classical decision problem with uncertainty about the true state. She knows, however, that the true state should be compatible with her signal. That is, the true state is one of the states in $\Omega(t_i) \subseteq \Omega$ that yields the signal, and she is optimizing her choice in this decision problem. In a Bayesian equilibrium, optimizing means maximizing expected utility over $\Omega(t_i)$ with respect to the given prior probability. In an ex-post equilibrium, optimizing means maximizing util-

ity for every compatible state.

Another equilibrium concept for pre-Bayesian games was recently defined in [4] - a minimax-regret equilibrium. In such an equilibrium optimizing means choosing an action that minimizes the maximal regret over compatible states. It was proved in [4] that a minimax-regret equilibrium always exist in a finite setup⁷ when players are using mixed strategies.⁸ An equilibrium concept for forms of pre-Bayesian games has been already defined in [25, 24, 23] in the context of work on artificial social systems, and [12] defined another equilibrium concept for particular pre-Bayesian auctions. Both groups of authors used distinct generalizations of maximin equilibrium, which was recently defined for classical pre-Bayesian games in [1]. In a maximin equilibrium optimizing means maximizing utility in the worst case scenario. The min in a maximin equilibrium ranges over all states that are compatible with the signal t_i , and the max ranges over all actions, X_i . Therefore, in a cost model the maximin equilibrium is actually a minimax equilibrium. We therefore prefer to use the term *safety level equilibrium* for both payoff and cost models. In [1] the authors mainly deal with the non-private type case, in which all players have the same signaling function. However, they prove existence results for the classical pre-Bayesian games described above.

The existence of multiple equilibria has always been a problematic issue in game theory. For pre-Bayesian games we also face the problem of multiple types of equilibrium. The question of which type of equilibrium in pre-Bayesian games makes a better prediction is yet to be explored. It is observed however, by [4] and [1] that every ex-post equilibrium is both a minimax-regret equilibrium and a safety level equilibrium.

In order to analyze resource selection games with unknown number of players we mainly use the safety level equilibrium solution. However, the model of pre-Bayesian games described above is not applicable for that matter, since the set of players in this model is fixed. We therefore generalize the model of pre-Bayesian games. In our generalized model, the set of players and the action sets of the players are state dependent.

We prove existence results for maximin and minimax-regret equilibrium in the generalized model.⁹ All proofs of existence (in [4], [1] and ours) make use of (variations of) Kakutani fixed point theorem. For the sake of completeness we also present and prove an existence theorem for a third, and new equilibrium concept for pre-Bayesian games

19, 20] for such analysis in the context of elections.

5 An exception is the price of anarchy theory.

6 Pre-Bayesian games have been also called *games in informational form* and *games without probabilistic information* [9, 7], *games with incomplete information with strict type uncertainty* [4], and *distribution-free games with incomplete information* [1].

7 Namely, the number of players and the number of states are finite.

8 Actually, the proof in [4] is given for a less general model of pre-Bayesian games than the one described above, but in this paper we extend their proof to the more general case.

9 We also allow compact convex sets of actions.

- the competitive ratio equilibrium. The competitive ratio approach is relatively common in computer science.¹⁰

Once we have the right tools we proceed to analyze resource selection games with unknown number of players. We focus on a model with increasing resource cost functions. In order to derive results for the case in which the number of players is unknown we prove several results, some of them are interesting for themselves, about classical resource selection games with known number of players. In particular we prove that every resource selection game possesses a unique mixed-action symmetric equilibrium. When the number of players is k , the unique symmetric equilibrium mixed-action of every player is denoted by p^k . That is, $p^k = (p_1^k, p_2^k, \dots, p_m^k)$, where p_j^k is the probability that a player chooses resource j .

In our pre-Bayesian model all active players know a common bound, n , on the number of active players, but the players do not know the true number of players, say k , $k \leq n$. Hence, the only signal a player receives is an "activity" signal. A state in this pre-Bayesian game is the set of active players. We prove that a resource selection game with unknown number of players has a unique symmetric safety level equilibrium. In this equilibrium, every active player is using the unique symmetric-equilibrium mixed-action, p^n in the game in which the number of players is commonly known and equals n .

Hence, when the number of players is k , every player uses p^k when the number of players is commonly known, and every player is using p^n when the number of players is unknown. Surprisingly, the lack of knowledge makes each of the players better off in the linear system, in which the resource cost functions are linear! That is, we show that in the linear model, when there are k players, and each of them is using p^n , the cost of each player is at most the cost he obtains in the unique symmetric equilibrium, p^k . Under very modest assumptions every player is strictly better off.

The above results are applicable to a mechanism design setup in which the organizer knows the number of active players, and the players do not know this number. If the goal of the organizer is to maximize revenue then he is better off revealing his private information.¹¹ If his goal is to maximize social surplus then he should not reveal the information. In order to estimate the gain of the players resulting from their ignorance we investigate in the last section the function $c^k(p^n)$ for $n \geq k$, where $c^k(p^n)$ is the cost for a player in a k -player setup when all players play the mixed-action associated with the symmetric equilibrium of the corresponding n -player setup.

In the full paper we also analyze the minmax-regret equilibrium of the related setting. We show that the phenomenon

described above does not hold for this solution concept. The precise relationships between the linear cost functions of the different resources determine whether knowing the number of participants is helpful or harmful for the players. Many of the proofs are omitted in this conference version of the paper.

2. Background

A game in strategic form is a tuple, $\Gamma = (N, (X_i)_{i \in N}, (u_i)_{i \in N})$. N is a nonempty set of players, X_i is a nonempty set of actions for player i , and $u_i : X \rightarrow \mathbb{R}$ is the payoff function of i , where $X = \times_{i \in N} X_i$ is the set of action profiles. Hence $u_i(x)$ is the payoff of player i when the profile of actions $x \in X$ is played. Γ is a *finite* game if N and X are finite sets.

Let $x \in X$ denote a profile of actions. For each $i \in N$ we let $x_{-i} = (x_j)_{j \in N/\{i\}}$ denote the actions played by everyone but i . Thus $x = (x_i, x_{-i})$. An action profile $x \in X$ is in *equilibrium* if $u_i(x_i, x_{-i}) \geq u_i(y_i, x_{-i})$ for every player $i \in N$ and for every $y_i \in X_i$.

A *permutation* of the set of players is a one-to-one function from N onto N . For every permutation π and for every action profile $x \in X$ we denote by πx the permutation of x by π . That is, $(\pi x)_i = x_{\pi(i)}$ for every player i . Γ is a *symmetric game* if for every player i , for every action profile x , and for every permutation π

$$u_i(\pi x) = u_{\pi(i)}(x).$$

A *symmetric action profile* is an action profile x such that $x_i = x_s$ for every $i, s \in N$. x is a *symmetric equilibrium* if it is both an equilibrium profile and a symmetric action profile.

For any finite set C , $\Delta(C)$ denotes the set of probability distributions over C . Let $\Gamma = (N, (X_i)_{i \in N}, (u_i)_{i \in N})$ be a finite game in strategic form. Every $p_i \in \Delta(X_i)$ is called a *mixed action* for i . $p_i(x_i)$ is the probability that player i plays action x_i . Every vector $p \in \Delta = \times_{i \in N} \Delta(X_i)$ defines a probability distribution over X ; The probability, $p(x)$ of $x \in X$ is $\prod_{i \in N} p_i(x_i)$.

Let u_i^m be the expected payoff function defined on Δ by u_i . That is, $u_i^m(p) = E_p(u_i)$.

The game $\Gamma^m = (N, (\Delta(X_i))_{i \in N}, (u_i^m)_{i \in N})$ is called the *mixed extension* of Γ . A mixed action for player i in the game Γ is an action for player i in the game Γ^m .

A *mixed action equilibrium* in a finite game Γ is defined to be an equilibrium in Γ^m . It is well-known (see e.g. [5]) that every finite symmetric game in strategic form possesses a symmetric mixed action equilibrium.

¹⁰ See e.g., [3].

¹¹ This is indirectly related to the Linkage Principle, in auction theory [16].

3. Resource Selection Games with Known Number of Players

In a *resource selection system*, $\Gamma = (m, (w_j)_{j=1}^m)$ there is a set of resources, $M = \{1, \dots, m\}$, $m \geq 1$. Every resource j is associated with a cost function $w_j : \{1, 2, \dots\} \rightarrow \mathbb{R}$. $w_j(k)$ is the cost for every user of resource j if there are k users. Together with the set of players $N_n = \{1, \dots, n\}$ a resource selection system defines a game in strategic form— a *resource selection game* $\Gamma(n)$. The action set of every player i in $\Gamma(n)$ is the set of resources M , and the cost of i depends on the resource she chooses and on the number of other players that choose this resource via the resource-cost functions. That is, $X_i^n = M$ for every $1 \leq i \leq n$, and $c_i^n(x) = w_{x_i}(\sigma_{x_i}(x))$, where for every resource j and for every action profile $x \in \times_{i=1}^n X_i^n = M^n$, $\sigma_j(x)$ is the number of all players s for which $x_s = j$. Obviously every resource selection game is a finite symmetric game.

Let $p \in \Delta(M)$ be a mixed action of an arbitrary player. That is, $p = (p_1, \dots, p_m)$, where p_j is the probability that a player who uses the mixed action p will select resource j . We denote the support of p by $\text{supp}(p)$. That is $\text{supp}(p) = \{j \in M | p_j > 0\}$. Denote by $c^n(p, j)$ the expected cost of a player that chooses resource j when each of the other $n-1$ players in $\Gamma(n)$ is using p . Let $c^n(p)$ be the expected cost of every player when each of the n players in $\Gamma(n)$ is choosing p .

For every $n \geq 1$, and for every $0 \leq \alpha \leq 1$. Let $Y_\alpha^n \sim \text{Bin}(n, \alpha)$ be a binomial random variable. That is, $f_\alpha^n(k) = P(Y_\alpha^n = k) = \binom{n}{k} \alpha^k (1-\alpha)^{n-k}$ for every $0 \leq k \leq n$. Let $F_\alpha^n(k) = P(Y_\alpha^n \leq k)$ be the distribution function of Y_α^n . Obviously

$$c^n(p, j) = E(w_j(1 + Y_{p_j}^{n-1})), \quad (1)$$

where E stands for the expectation operator. That is,

$$c^n(p, j) = \sum_{s=0}^{n-1} w_j(s+1) f_{p_j}^{n-1}(s). \quad (2)$$

Let $(q, \dots, q) \in \Delta(M)^n$ be a symmetric mixed-action equilibrium profile in $\Gamma(n)$. We will refer to q as a *symmetric-equilibrium action*.

Theorem 1 *Every resource selection game with at least two players ($n \geq 2$) with increasing¹² resource cost functions possesses a unique symmetric mixed-action equilibrium.*

In order to prove Theorem 1 we need some preparations. We prove two lemmas, and a corollary to these lemmas. The proofs of Theorem 1, of the lemmas, and of the corollary are given in the full version.

¹² That is, $w_j(k) < w_j(k+1)$ for all j and k .

Lemma 1 *Let $n \geq 1$. $F_\alpha^n(k)$ is a strictly decreasing function of α for every $0 \leq k \leq n-1$.*

Lemma 2 *Let $\Gamma(n)$, $n \geq 2$ be a resource selection game. Let $q, p \in \Delta(M)$ be mixed actions, and let $j \in M$ be a resource such that w_j is increasing in $\{1, 2, \dots, n\}$. If $p_j > q_j$ then $c^n(p, j) > c^n(q, j)$.*

Corollary 1 *Let $\Gamma(n)$, $n \geq 2$ be a resource selection game with increasing resource cost functions. All symmetric equilibrium actions in $\Gamma(n)$ have the same support.*

For every $n \geq 1$ we will denote the unique symmetric equilibrium in $\Gamma(n)$ by p^n , and we denote by $c^n = c^n(p^n)$ the equilibrium cost of a player in $\Gamma(n)$.

We say that a resource cost function w_j is *convex* if it can be extended to a convex function on $[1, \infty)$.

The following lemma will be useful later.

Lemma 3 *Let $\Gamma = (m, (w_j)_{j=1}^m)$ be a resource selection system, with increasing and convex cost functions. There exists an integer $K \geq 2$, $K = K(\Gamma)$ such that for every $n \geq K$, the unique symmetric-equilibrium action in the game $\Gamma(n)$, $p^n \in \Delta(M)$ has a full support. That is, $p_r^n > 0$ for every $1 \leq r \leq m$.*

Proof: Recall that p^n is the unique symmetric-equilibrium action in $\Gamma(n)$, and that $c^n = c^n(p^n)$ is the symmetric-equilibrium cost of every player. As p^n is in equilibrium, $c^n(p^n, j) = c^n$ for every $j \in \text{supp}(p^n)$. For every resource j we denote by w_j the convex extension of w_j to $[0, \infty)$. As w_j is convex,

$$c^n(p^n, j) = E(w_j(1 + Y_{p_j^n}^{n-1})) \geq w_j(1 + E(Y_{p_j^n}^{n-1})) = w_j(1 + p_j^n(n-1)),$$

where the first equality follows from (1), the inequality follows from the convexity of w_j , and the last equality follows from the well-known fact that

$$E(Y_\alpha^n) = \alpha n. \quad (3)$$

Obviously, there exists $j \in \text{supp}(p)$ for which $p_j^n \geq \frac{1}{m}$. For this resource j

$$c^n = c^n(p^n, j) \geq w_j(1 + \frac{1}{m}(n-1)) \geq$$

$$\min_{r=1}^m w_r(1 + \frac{1}{m}(n-1)).$$

Since w_j is increasing and convex, $\lim_{n \rightarrow \infty} w_j(n) = \infty$ for every resource j . Therefore $\lim_{n \rightarrow \infty} c^n = \infty$. Hence, there exists K such that for every $n \geq K$ $c^n > \max_{j=1}^m w_j(1)$. We claim that for every $n \geq K$, $p_r^n > 0$ for every $1 \leq r \leq m$. Indeed, if $p_r^n = 0$ for some r then because $c^n > w_r(1)$, a player will decrease her cost by deviating from p^n to r (assuming every other player is using p^n). This contradicts p^n being a symmetric-equilibrium action. ■

4. Equilibrium in Pre-Bayesian Games

In this section we define a general model of pre-Bayesian games, and we present and prove existence results for minimax-regret, safety-level, and competitive-ratio equilibria in such games.

A *pre-Bayesian game* is a tuple $G = (N, \Omega, (X_i)_{i \in N}, (S_i)_{i \in N}, (T_i)_{i \in N}, (\tilde{t}_i)_{i \in N}, (u_i)_{i \in N}, (e)_{i \in N})$, where

- N is the set of potential players.
- Ω is the set of *states*.
- X_i is the set of possible actions of agent i — Only a subset of X_i will be available for i at a given state ω .
- The symbol e_i stands for a dummy action for i . It does not belong to X_i . Let $Z_i = X_i \cup \{e_i\}$, and let $Z = \times_{i \in N} Z_i$.
- $T_i \subseteq 2^{X_i} \times S_i$ is the set of *types* of i . Hence, every type of i is a pair $t_i = (t_i^1, t_i^2)$, where $t_i^1 \subseteq X_i$ and $t_i^2 \subseteq S_i$. t_i is an *active* type if $t_i^1 \neq \emptyset$.
- $\tilde{t}_i : \Omega \rightarrow T_i$ is the *signaling function* of i ; $\tilde{t}_i(\omega) = (t_i^1(\omega), t_i^2(\omega)) = (X_i(\omega), s_i(\omega))$. For every t_i we denote by $\Omega(t_i)$ the set of states that can generate t_i . That is $\Omega(t_i) = \{\omega \mid \tilde{t}_i(\omega) = t_i\}$. We assume that $\Omega(t_i) \neq \emptyset$ for every $t_i \in T_i$.

Player i is *active* at ω if $\tilde{t}_i(\omega)$ is an active type. The set of all active players at ω is denoted by $N(\omega)$.

- $u_i : \Omega \times Z \rightarrow \mathbb{R}$ is the *payoff function* of i .

The pre-Bayesian game proceeds as follows. Nature chooses $\omega \in \Omega$. Every player $i \in N(\omega)$ chooses an action $x_i \in X_i(\omega)$. Every non-active player j does nothing. It is modelled by choosing e_j . Hence, the players form a profile $z \in Z$. Every active player i receives $u_i(\omega, z)$. A *strategy* of i is a function $b_i : T_i \rightarrow Z_i$ such that $b_i(\emptyset, s_i) = e_i$ for every $s_i \in S_i$, and $b_i(t_i) \in t_i^1$ whenever $t_i^1 \neq \emptyset$. We denote by Σ_i the set of strategies of player i , and let $\Sigma = \times_{i \in N} \Sigma_i$.

Let G be a pre-Bayesian game. We say that G is a *finite* if the set of potential players, the set of states, and the strategy sets are finite. We say that G is a *compact-continuous pre-Bayesian game* if the set of players and the set of states are finite, X_i is a compact subset of some Euclidean space¹³, for every type t_i , t_i^1 is a compact subset of X_i , and for every $\omega \in \Omega$, the restriction of $u_i(\omega, *)$ to $X_i \times Z_{-i}$ is continuous in x_i .

A compact continuous pre-Bayesian game is *concave* if X_i is a convex subset of some Euclidean space, every $t_i^1 \subseteq X_i$ is convex, and for every $\omega \in \Omega$, the restriction of $u_i(\omega, *)$ to $X_i \times Z_{-i}$ is concave in x_i .

Let $G = (N, \Omega, (X_i)_{i \in N}, (S_i)_{i \in N}, (T_i)_{i \in N}, (\tilde{t}_i)_{i \in N}, (u_i)_{i \in N}, e)$ be a finite pre-Bayesian game. Denote

$\Delta_i = \Delta(X_i)$, and for every $Y_i \subseteq X_i$ let $O_i(Y_i)$ be the set of all probability distributions in Δ_i that vanish outside Y_i . Note that $O_i(\emptyset) = \emptyset$. Let $T_i^m = \{(O_i(t_i^1), t_i^2) \mid (t_i^1, t_i^2) \in T_i\}$. Define $\tilde{t}_i^m : \Omega \rightarrow T_i^m$ as follows: $\tilde{t}_i^m(\omega) = (O_i(\tilde{t}_i^1(\omega)), \tilde{t}_i^2(\omega))$. The pre-Bayesian game $G^m = (N, \Omega, (\Delta_i)_{i \in N}, (S_i)_{i \in N}, (T_i)_{i \in N}, (\tilde{t}_i^m)_{i \in N}, (u_i^m)_{i \in N}, e)$ is called the *mixed extension* of G , where u_i^m is the expected-payoff function defined by u_i . Obviously G^m is a concave pre-Bayesian game. Every strategy of i in G^m is called a *mixed-strategy* for i in G .

4.1. Safety-level equilibrium

Safety-level equilibrium were defined by Aghassi and Bertsimas in ([1]). We use their definition for our general model of pre-Bayesian games.

Let G be a compact-continuous pre-Bayesian game. Let $b_{-i} = (b_j)_{j \in N \setminus i}$ be a profile of strategies of all players but i , and let $t_i = (Y_i, s_i)$ be an active type for i . For every $y_i \in Y_i$ the worst case payoff of i is

$$V_i(t_i, b_{-i}, y_i) = \min_{\omega \in \Omega(t_i)} u_i(\omega, y_i, b_{-i}(\tilde{t}_{-i}(\omega))).$$

Obviously V_i is continuous on Y_i . We say that $y_i^* \in Y_i$ is *optimal* for an active type $t_i = (Y_i, s_i)$ given b_{-i} if the maximal value of $V_i(t_i, b_{-i}, y_i)$ over $y_i \in Y_i$ is attained at y_i^* . A strategy of player i , b_i , is a *safety-level best-response* to b_{-i} if for every active type t_i , $b_i(t_i)$ is optimal for t_i given b_{-i} . A strategy profile $b = (b_i)_{i \in N}$ is called a *safety-level equilibrium* if for every i , b_i is a safety-level best-response to b_{-i} .

Hence, b is a safety-level equilibrium if and only if for every ω , and for every player i , which is active at ω , $b_i(\tilde{t}_i(\omega))$ is optimal for $\tilde{t}_i(\omega)$ given b_{-i} .

A safety-level equilibrium in a pre-Bayesian game with exactly one state is simply a Nash equilibrium in this game. We next show that safety-level equilibria exist in every concave game.

Theorem 2 *Every concave pre-Bayesian game possesses a safety-level equilibrium.*

Proof: For every i , let T_i^a be the set of active types, and let $T_i^d = T_i \setminus T_i^a$ be the set of dummy types. Every $b_i \in \Sigma_i$ is determined by its restriction to T_i^a . Note that Σ_i is a closed convex subset of $X_i^{T_i^a}$, which is a convex and compact set because it is a cartesian products of such sets. In this sense, every Σ_i and Σ are convex and compact spaces. For every $b \in \Sigma$ and for every $i \in N$ let $B_i(b) \subseteq \Sigma_i$ be the set of all $d_i \in \Sigma_i$, which are best response to b_{-i} . Let $B(b) = \times_{i \in N} B_i(b) \subseteq \Sigma$. It is standard to check that the correspondence $b \rightarrow B(b)$ satisfies the conditions of Kakutani's fixed point theorem. That is, it is upper hemicontinuous, and $B(b)$ is a nonempty compact convex subset of Σ .

¹³ Or of some linear topological space.

for every $b \in \Sigma$. Therefore there exists a fixed point b , that is, $b_i \in B_i(b)$ for every $i \in N$. Obviously such a fixed point is a safety-level equilibrium. ■

Hence, if G is a finite game, G^m possesses a safety-level equilibrium. Every such an equilibrium is called a *mixed-strategy safety-level equilibrium* in G .

4.2. Minimax-regret equilibrium

Minimax-regret equilibrium were defined by Hyafil and Boutilier in [4]. We use their definition for our general model of pre-Bayesian games.

Let G be a compact-continuous pre-Bayesian game. Let $b_{-i} = (b_j)_{j \in N \setminus i}$ be a profile of strategies of all players but i , let t_i be an active type of i , and let $\omega \in \Omega(t_i)$. The *regret* of $x_i \in t_i^1$ at w is defined as:

$$R(x_i, w, t_i, b_{-i}) = \max_{z_i \in t_i^1} [u_i(w, z_i, b_{-i}(\tilde{t}_{-i}(\omega))) - u_i(w, x_i, b_{-i}(\tilde{t}_{-i}(\omega)))].$$

The *maximal regret* of $x_i \in t_i^1$ over all $\omega \in \Omega(t_i)$ is denoted by $MR(x_i, t_i, b_{-i})$. That is:

$$MR(x_i, t_i, b_{-i}) = \max_{\omega \in \Omega(t_i)} R(x_i, w, t_i, b_{-i}).$$

We say that $y_i \in t_i^1$ is a minimax regret strategy at t_i given b_{-i} if the minimal value of $MR(x_i, t_i, b_{-i})$ over $x_i \in t_i^1$ is attained at y_i . A strategy b_i is a minimax regret best response to b_{-i} if for every active type t_i , $b_i(t_i)$ is a minimax regret strategy at t_i given b_{-i} . b is a *minimax regret equilibrium* if for every player i b_i is a minimax regret best response to b_{-i} . The next theorem shows that minimax-regret equilibria exist in every concave pre-Bayesian game. The proof is similar to the proof of theorem 2.

Theorem 3 *Every concave pre-Bayesian game possesses a minimax-regret equilibrium.*

4.3. Competitive ratio equilibrium

Competitive ratio equilibrium resembles the minimax-regret equilibrium. They differ only in the definition of regret.

Let G be a compact-continuous pre-Bayesian game, where all payoff functions are positive. Let $b_{-i} = (b_j)_{j \in N \setminus i}$ be a profile of strategies of all players but i , let t_i be an active type of i , and let $\omega \in \Omega(t_i)$. Replace the definition of $R(x_i, w, t_i, b_{-i})$ in the previous section with the following:

$$\hat{R}(x_i, w, t_i, b_{-i}) = \max_{z_i \in t_i^1} \frac{u_i(w, z_i, b_{-i}(\tilde{t}_{-i}(\omega)))}{u_i(w, x_i, b_{-i}(\tilde{t}_{-i}(\omega)))}.$$

Note that $\hat{R}$ is well defined since all payoff functions are positive. A strategy profile b is a competitive-ratio equilibrium if for every player i b_i is a minimax-regret best response to b_{-i} with respect to the regret function $\hat{R}$.

Theorem 4 *Every concave game in pre-Bayesian game with positive payoffs functions possesses a competitive ratio equilibrium.*

The proof follows from Theorem 3 by applying the logarithmic function to the payoff functions.

5. Resource Selection Games with Unknown Number of Players

Consider a fixed resource selection system, Γ with the set of resources $M = \{1, \dots, m\}$, $m \geq 1$, and resource cost functions $(w_j)_{j=1}^m$.

We proceed to describe our model of resource selection games with unknown number of players. Let $N = \{1, 2, \dots, n\}$, $n \geq 1$ be the set of potential players. The set of states, Ω is the set $2^N \setminus \{\emptyset\}$ of all nonempty subsets of N . The set of actions of player i is the set of resources. That is, $X_i = M$ for every player i . The sets S_i will have no use, and therefore we ignore them. The set of types of i is $T_i = \{\emptyset, M\}$. The signaling functions are defined as follows: $\tilde{t}_i(J) = M$ if $i \in J$ and $\tilde{t}_i(J) = \emptyset$ if $i \notin J$. Recall that $Z_i = M \cup \{e_i\}$, where e_i denotes a dummy action. The cost function of i is $c_i : \Omega \times Z \rightarrow \mathbb{R}$, where for $i \in J \subseteq N$, $c_i(J, z) = w_{z_i}(\sigma_{z_i}(z))$, where $\sigma_{z_i}(z)$ is the number of all players $l \in N$ for which $z_l = z_i$. The values of $c_i(J, z)$ for subsets of N that do not contain i are not relevant and should not be specified. The above pre-Bayesian game is finite. We denote its mixed extension by $G_\Gamma(n)$, and we referred to $G_\Gamma(n)$ as a *resource selection game with unknown number of players*. A strategy of player i in $G_\Gamma(n)$ can be described by a mixed action $q[i] \in \Delta(M)$. That is, when receiving the signal M , i uses $q[i]$ and when receiving the signal $\emptyset$, i uses her dummy action, e_i . Since we deal with costs and not with payoffs we use minimax rather than maximin in the definition of safety-level equilibrium. Let $\mu = (q[1], \dots, q[n])$ be a strategy profile in $G_\Gamma(n)$, and let i be a player. Let $\mu[-i] = (q[l])_{l \in N \setminus \{i\}}$ be the profile of strategies of the other players. If i is active, that is she received the signal M , then the set of states that are compatible with i 's signal, $\Omega_i(M)$ is the set of all nonempty subsets of N that contain i . If all cost functions are non-decreasing, and if i believes that all other players are using the profile $\mu[-i]$, then it is obvious that the worst case scenario for i is obtained in the state N . Thus we have:

Lemma 4 *Let Γ be a resource selection system in which the resource cost functions are non-decreasing. Let $\mu \in \Delta(M)^n$. μ is a safety-level equilibrium in $G_\Gamma(n)$ if and only if μ is a mixed-action equilibrium in $\Gamma(n)$.*

Proof: Assume μ is a mixed-action equilibrium in $\Gamma(n)$. Let i be an active player. By the comment we made before the statement of the lemma,

$$\min_{p[i] \in \Delta(M)} \max_{S \in \Omega(M)} c_i(J, p[i], \mu[-i]) = \min_{p[i] \in \Delta(M)} c_i(N, p[i], \mu[-i]).$$

Because μ is a mixed-action equilibrium in $\Gamma(n)$, the min in the right hand-side of the above formula is attained at $q[i]$. Therefore μ is a safety-level equilibrium in $G_\Gamma(n)$. An analogous argument proves the if part of the lemma. ■

Theorem 5 *Let Γ be a resource selection system in which the resource cost functions are non-decreasing. $G_\Gamma(n)$ has a unique symmetric safety-level equilibrium. In this symmetric safety-level equilibrium every player is using the strategy p^n , where p^n is the unique symmetric-equilibrium action in $\Gamma(n)$.*

Proof: The proof follows directly from Theorem 1 and Lemma 4. ■

By Theorem 5, each of the players in $G_\Gamma(n)$ is using the strategy p^n , where p^n is the unique symmetric-equilibrium mixed action in $\Gamma(n)$. However, the cost of each active player in $G_\Gamma(n)$ is not $c^n = c^n(p^n)$, it depends on the true state. If the true state is J , that is J is the set of active players, and $|J| = k$, the cost of each active player i is $c^k(p^n)$. It is worthy to compare this cost with the cost $c^k = c^k(p^k)$ that every player in J would have paid had the players in J known the state. We make these comparison in linear models. We say that a resource selection system is *linear* if for every resource j there exists a constant d_j such that $w_j(k) = w_j(1) + (k-1)d_j$ for every $k \geq 1$. For every number of players, n , the associated resource selection game, $\Gamma(n)$, as well as the associated resource selection game with unknown player set, $G_\Gamma(n)$ will be called *linear* too. Note that in a linear system, w_j is increasing if and only if $d_j > 0$, and w_j is non-decreasing if and only if $d_j \geq 0$. The proof of the following theorem, Theorem 6 is given in the full version:

Theorem 6 *Let Γ be a linear resource selection system with increasing resource cost functions. For every $k \geq 2$ let p^k be the unique symmetric equilibrium in $\Gamma(k)$. There exist an integer $K = K(\Gamma)$, $K \geq 2$ such that for all $n > k \geq K$:*

1. $c^k(p^k) \geq c^k(p^n)$.
2. All inequalities above are strict if and only if there exists $j_1, j_2 \in M$ such that $w_{j_2}(1) \neq w_{j_1}(1)$

Theorem 6 is applicable to a mechanism design setup in which the organizer knows the number of active players, and the players do not know this number. If the goal

of the organizer is to maximize revenue then he is better off revealing his private information. If his goal is to maximize social surplus, then he should not reveal that information. In order to estimate the gain of the players resulting from their ignorance we analyze the function $c^k(p^n)$. The proof of the following theorem is given in the full version:

Theorem 7 *Let Γ be a linear resource selection system with increasing resource cost functions. There exist K such that for every $n \geq K$ the following assertions hold:*

1. $p_j^n = \frac{n-1+B-w_j(1)A}{Ad_j(n-1)}$, where $A = \sum_{j=1}^m \frac{1}{d_j}$, and $B = \sum_{j=1}^m \frac{w_j(1)}{d_j}$.
2. The minimal social cost in $\Gamma(k)$ attained with symmetric mixed-action profiles is attained at p^{2k-1} . Consequently, $c^k(p^n)$ is minimized at $n = 2k - 1$.

References

- [1] M. Aghassi and D. Bertsimas. Robust Game Theory. 2004.
- [2] I. Ashlagi, D. Monderer, and M. Tennenholtz. On the value of correlation. In *Proceedings of the 21th Annual Conference on Uncertainty in Artificial Intelligence (UAI-05)*, pages 34–41, Corvallis, Oregon, 2005. AUAI Press.
- [3] A. Borodin and R. El-Taniv. *On-Line Computation and Competitive Analysis*. Cambridge University Press, 1998.
- [4] C. Boutilier and N. Hyafil. Regret minimizing equilibria and mechanisms for games with strict type uncertainty. In *Proceedings of the 20th Annual Conference on Uncertainty in Artificial Intelligence (UAI-04)*, pages 268–277, Arlington, Virginia, 2004. AUAI Press.
- [5] S. Cheng, D. M. Reeves, Y. Vorobeychik, and M. Wellman. Notes on Equilibria in Symmetric Games. Technical Report, University of Michigan, 2004.
- [6] R. M. Harstad, J. Kagel, and D. Levin. Equilibrium bid functions for auctions with an uncertain number of bidders. *Economics Letters*, 33:35–40, 1990.
- [7] R. Holzman, N. Kfir-Dahav, D. Monderer, and M. Tennenholtz. Bundling equilibrium in combinatorial auctions. *Games and Economic Behavior*, 47:104–123, 2004.
- [8] R. Holzman and N. Law-yone (Lev-tov). Strong equilibrium in congestion games. *Games and Economic Behavior*, 21:85–101, 1997.
- [9] R. Holzman and D. Monderer. Characterization of Ex Post Equilibrium in the VCG Combinatorial Auctions. *Games and Economic Behavior*, 47:87–103, 2004.
- [10] H. Konishi, M. L. Breton, and S. Weber. Equilibrium in a Model with Partial Rivalry. *Journal of Economic Theory*, 72:225–237, 1997.
- [11] E. Koutsoupias and C. Papadimitriou. Worst-Case Equilibria. In *Proceedings of the 33rd Symposium on Theory of Computing*, pages 404–413, 1999.
- [12] D. Levin and E. Ozdenoren. Auctions with Uncertain Number of Bidders. To appear in 'Journal of Economic Literature', 2004.

- [13] S. Matthews. Comparing auctions for risk averse buyers: a buyers point of view. *Econometrica*, 55:633–646, 1987.
- [14] R. P. McAfee and J. McMillan. Auctions with a stochastic number of bidders. *Journal of Economic Theory*, 43:1–19, 1987.
- [15] I. Milchtaich. Congestion Games with Player-Specific Pay-off Functions. *Games and Economic Behavior*, 13:111–124, 1996.
- [16] P. Milgrom and R. Weber. A Theory of Auctions and Competitive Bidding. *Econometrica*, 50:1089–1122, 1982.
- [17] D. Monderer and L. Shapley. Potential Games. *Games and Economic Behavior*, 14:124–143, 1996.
- [18] R. B. Myerson. Population uncertainty and Poisson games. *International Journal of Game Theory*, 27:375–39, 1998a.
- [19] R. B. Myerson. Extended Poisson games and the Condorcet jury theorem. *Games and Economic Behavior*, 25:111–131, 1998b.
- [20] R. B. Myerson. Large Poisson games. *Journal of Economic Theory*, 94:7–45, 2000.
- [21] T. Quint and M. Shubic. A Model of Migration. working paper. Cowles Foundation, Yale University, 1994.
- [22] R. Rosenthal. A Class of Games Possessing Pure-Strategy Nash Equilibria. *International Journal of Game Theory*, 2:65–67, 1973.
- [23] Y. Shoham and M. Tennenholtz. Artificial Social Systems. *Computers and Artificial Intelligence*, 14:533–562, 1995.
- [24] Y. Shoham and M. Tennenholtz. On Social Laws for Artificial Agent Societies: Off-Line Design. *Artificial Intelligence*, 73:231–252, 1995.
- [25] M. Tennenholtz. *Efficient Representation and Reasoning in Multi-Agent Systems*. PhD Thesis, Weizmann Institute of Science, Israel, 1991.
- [26] M. Voorneveld, P. Borm, F. van Meegen, S. Tijs, and G. Facchini. Congestion Games and potentials reconsidered. *International Game Theory Review*, 1:283–299, 1999.

Creating Incentives for Honest Rating Submission by Rating the Raters

Jörg H. Lepler

Cambridge University

Computer Laboratory Faculty of Economics

Joerg.Lepler - at - cl.cam.ac.uk

Abstract

Reputations of electronic parties are important for their economic viability in business settings. Potential clients of e-services cannot rely solely on their own experiences to gain knowledge about e-services' reputations. Plainly aggregating rating comments from other clients can often be misleading. Therefore, we are presenting a reputation aggregation model which seeks for the subgroup of raters that (1) contains the largest degree of overall agreement and (2) derives the resulting reputation scores from their comments. We complement the reputation system by adding policies that promote both truth revelation for the rater commentary, and also competition over service quality between providers. Then we describe the recursive algorithm of the model, which judges a rater's commenting quality based on the divergence between his and the other raters' comments. This algorithm feeds the commenting quality back into the aggregation function as a weight on the rater's comments. We evaluate our algorithm in simulations of three challenging threat scenarios. To this end, we show that our aggregation model can be effectively used to deselect weak rating accuracy, and to filter out a malicious collective of raters which makes up almost half of the rating population. Finally, we show that the model is able to reveal provider behavior that is biased in favor of specific groups of raters.

1 Motivation for a Reputation Reporting Service

Clients choose a service provider on the basis of (1) price versus promised performance aspects such as, for example, a provider's contractual terms in an SLAs and (2) the client's confidence in how well this provider will deliver on the negotiated performance. Aggregating the confidence of many clients resembles the reputation of a provider. The importance of reputation in the clients' decisions often is

underestimated in the designs of electronic market places. This has been the case in, for example, Giovanetti and Ristuccia's [6] analysis on the band-x backbone bandwidth market, where the researchers found that clients did not rely much on the reported performance numbers, but more so on the reputations of large, well-known providers.

Introducing widely accepted reputation systems for e-services requires addressing a collection of pitfalls that are inherent to them. Most of these pitfalls are not of a technical nature. For technical policies such as "we want to keep commentators anonymous", we are able to devise technical solutions. The questions that are difficult to address are in the nature of the design philosophy.

For once, one needs a definition of the scales by which to measure the reputation of a provider - a definition which includes the understanding of the pragmatic meaning of these scales. The reputation model we are presenting makes only one assumption about the choice of the reputation valuation function, namely that the reputation system operator has chosen an appropriate function.

The question, then, arises: Can we trust the rating submissions from all raters? Should we give more weight to those raters who are seemingly more trustworthy than those who might be less well-informed? To address these questions, our model has the ability to redistribute the influence it assigns to a rater on the aggregated scores, and to do so in favor of certain better informed raters. In addition, we are able to produce different views of the reputation results, depending on which rater, or set of raters, we ex ante trust more.

A separate threat model category exists on the side of the providers: Do they treat all their clients equally? Can we recognize provider discrimination? Can we distinguish discrimination behavior from biased client behavior? We address these different threat models by simulation them in challenging rating scenarios.

Finally, do clients have some incentive to submit ratings at all? Do they have a tangible incentive state their experiences

truthfully? We create these incentives by listing the clients as contributors to the reputation system and adapt policies that discourage undesired behavior.

A more detailed description of the reputation system’s model, as well as further examples and demonstrations of its applicability can be found in [8]. This includes a detailed investigation of the technical model properties, as well as a practical example. In the following section we discuss some of the design choices and prior observations we made that led to the development of the reputation system as we will present it in section 3.

2 Reputation System Philosophy

2.1 Asymmetric Agents

Our system distinguishes between raters and rated entities, and allows only for uni-directional ratings. Our design requires this distinction because our market model assumes a regular business-style scenario with separate clients and providers. In this market model, clients are usually *not* providers for the same service, as in peer-to-peer networks or eBay-style trading transactions. This distinction has significant drawbacks, because it is easier to assess the credibility of raters when judging their rating behavior alongside their providing performance. For example, in peer-to-peer systems, I can distrust both the ratings submitted by a peer who has a poor provision performance, and also all the peers who rated this peer highly. In a client-server market model we lack such a high degree of interaction and rating connectivity. Nevertheless, we want to use such a model to calibrate raters’ comments by the quality of comments they are submitting.

2.2 Quantitative Scales

We require that all users of our reputation system agree on one scale and assigned meaning of the reputation variable, which together characterize the recommended ordering of preferred providers. This means that the reputation variable has to be an objectively measurable, numeric value $\in \mathbb{R}$ which can be observed by all users in the same way. This $\mathbb{R}$ -reputation variable may be a technically measurable metric, such as the fact that the service has a response latency of a few seconds. But the metric could also be a percentage variable which notes providers’ performances in the form of a qualitative statement such as “90% of all transactions with this service were successful”. With transformation functions like this, one can adapt many qualitative observations, including binary ones, to a numeric $\mathbb{R}$ -variable.

Why should we not operate more with qualitative observations though? A qualitative assessment usually produces a more accurate picture, but leads to difficulties at the point of comparison and aggregation of observations. The motivation, for this requirement is that most of the times, when evaluating and judging an object - which is what reputations are about - one does so in order to make a choice, or better to produce a ranking of preferences. Thus, our desired outcome is not qualitative, but actually a quantitative description of precise orderings. In order to obtain the quantitative results, we need at some point a mapping from qualitative to quantitative observations. It is our assertion that we achieve more transparent aggregation results, if this transition from qualitative to quantitative observations is done rather earlier than later in the aggregation process.

The technical reason for demanding objectively measurable reputation input parameters is that our rating model is not able to adjust for generally biased raters, i.e. raters who judge entities in the same way as other raters, but who nonetheless assign all entities one full grade lower than the unbiased raters. The model would recognize behavior like this as the grounds for a “poor rating”, and would reduce the influence that such a rater would have on the final scores. On a more general level, though, as pointed out by Pennock [10], a reputation system would need to be independent of the scales. Economists in general believe that the absolute magnitude of one user’s scale cannot be compared with another user’s (see Arrow [2] and Sen [11]). This is due to the fact that users inescapably are biased, and implies that users’ utilities cannot be added up because they are invariant under linear transformations. Thus, it is necessary to require agreement among all the users on the scale of the reputation values. This agreement allows clients to have a certain variance with regard to the technical applications of the metric.

2.3 Rating Commentators

The reputation system calculates the performance scores of the services by averaging the comments submitted by the rating-clients. After collecting the divergences between all the comments given by a rater and the overall scored obtained, we derive a quality rating about the rater. Then the rater’s quality variable is fed back into the rating of the performance scores, as a weight on his comments.

2.4 Truth Revelation Incentives

Given the uni-directional rating design of the system, we need explicit incentives which will entice clients to submit comments about their experiences with the services, and

moreover, to report on these truthfully. We create this incentive for each client by publishing, as a part of the reputation system, a list of clients who contributed comments and the number of comments they did present. We expect that a client who is listed as a rater can expect to be treated well by the service providers, because his listing is a signal to them that in future he also will submit reports on their performance. The number of comments that a client has submitted reveals whether a client, relative to his activity level, has supplied more than just a token commentary.

To entice truth revelation from the clients, we have to apply another measure to our reputation system. We considered the possibility of publishing the rater reputation that our system calculates in the process of aggregating scores. There, a rater's reputation reflects his rating accuracy and the influence assigned to his comments by the reputation system. Making this rater reputation public would increase the rater's incentive to provide high ranking commentary, because this would send a stronger signal to the providers. However, the rater reputation should be kept undisclosed by the reputation system, since knowing this would also create an incentive for the clients to modify or even fabricate their comments in a targeted way. They could do so by submitting comments that reflect the currently published scores for the services. So actually, in order to promote truthful comment submissions, the reputation system adopts a policy of striking clients off its list of raters if they submit erratic or modified comments. We can recognize such clients from their particularly low rating reputations. We should point out that the clients' reputations and their influence on the ratings will decline and remain unnoticed outside of the reputation system before we strike them off the rater register. We strike them off once the coherency of their ratings underpasses a certain low standard. Figure 3 in section 4.1 gives an example for how inaccurate raters can be identified through the reputation system. With this policy, raters have no incentive to submit untruthful comments. Instead, they have an incentive against this, particularly because they are not able to gauge how high their own rating reputation is. Rating reputations depend on all the other comments and thus are only known by the reputation system.

Another advantage in adopting this policy of striking off random or malicious raters is that doing so reassures providers that the reputation system is meaningful and that they should accept the published rankings. This acceptance should increase their willingness to improve their service quality where necessary. Service quality improvements should actually be self-evident if the reputation system creates competition over quality aspects.

3 The Rating Mechanism

We have a set of l raters $R = \{r_1, \dots, r_l\}$ who rate k service providers $P = \{p_1, \dots, p_k\}$ on the performance they experienced as clients. The providers perform at a certain 'real' performance level and this is the value we ideally would like to obtain as a final score for this provider in the reputation reporting system. However, this 'real' level is an unknown variable, possibly even unknown to the provider himself. Every time a client requests services from this provider the client experiences a probabilistic value of this performance. However, these individual 'experiences' may be observed slightly differently by each individual client. The individual differences arise from a number of reasons such as them applying different measurements, or aggregating their experiences on different scales.

Every so often, at time t rater $r_A \in R$ will make a comment: $c_{t, r_A \mapsto p_B} \in C$, about provider $p_B \in P$, reporting his observations in form of a numerical value: $val(c) \in \mathbb{R}$. The following calculations are performed for all raters and providers, however in order to simplify the notation, we will demonstrate the calculations for an exemplary rater r_A and an exemplary provider p_B . As the comments are collected at a central point in the reputation system, we define all comment time stamps t to be distinct. Ratets can submit more than one comment about a provider and these can be distinguished by the time the comment was made.

After collecting a sufficient number of comments¹ on a provider: $|\{c_{t, r_* \mapsto p_B}\}|$, we can calculate an initial score for this provider. In our terminology '*' represents any client or provider applicable. Before entering the iterative loop, we initialize the provider scores to obtain the *initialization vector* $s_{p_1, \dots, p_k}^{(0)}$. This initialization simply averages the values of all the comments:

$$\forall p_B \in \{p_1, \dots, p_k\} : s_{p_B}^{(0)} = \begin{cases} \frac{\sum_{c \in C'} val(c)}{|C'|}, & C' = \{c_{t, r_* \mapsto p_B} \in C\} \\ 0, & \text{if } C' = \emptyset. \end{cases} \quad (1)$$

C' is the set of all comments made by any rater about provider p_B and if this set is empty, we assign a score of null. The (0) represents the iteration number², indicating the initialization procedure at this point, where we simply average the received comments.

¹It is possible to develop a confidence value that indicates how many comments are sufficient to achieve statistical integrity. However, we decided to omit developing and analyzing such a variable.

²In our notation, where variables (e.g. s) are indexed by the iteration number (e.g. n), such as $s^{(n)}$, the iteration number is kept in parenthesis in the superscript in order to distinguish these from power operations.

Next, we calculate the score each rater would assign by himself for each provider. For each provider, we take the collection of raters having submitted comments on this provider and then for each rater within this set we average his comments on this provider to obtain a local score $local.s_{r_A \mapsto p_B}$. Here, C' is the set of all comments made by rater r_A at different times about provider p_B and if this set is empty, $C' = \emptyset$, we assign a score of null, $local.s_{r_A \mapsto p_B} = 0$:

$$\forall r_A \in \{r_1, \dots, r_l\} : \forall p_B \in \{p_1, \dots, p_k\} :$$

$$local.s_{r_A \mapsto p_B} = \begin{cases} \frac{\sum_{c \in C'} val(c)}{|C'|}, & C' = \{c_t, r_A \mapsto p_B \in C\} \\ 0, & \text{if } C' = \emptyset. \end{cases}$$

After calculating initial and local scores for all providers, we enter the iterative loop and calculate for all the raters their reputations $q_{r_A \in R}$: The reputation is calculated from the difference between rater A 's comments and the above calculated initial scores $s_p^{(0)}$. Then we calculate the standard deviation of this difference. Of this difference we take the norm by dividing it by the number of comments submitted by this client. This yields our reputation value, except that, as it stands, a good rater with a high rating reputation would be assigned a smaller value q . In order to correlate good reputations to high reputation values, we invert this final reputation value. C' is the set of all comments made by rater r_A about any provider and if this rater did not submit any comments at all, he is assigned a rater reputation of null, $q_{r_A} = 0$:

$$\forall r_A \in \{r_1, \dots, r_l\} :$$

$$q_{r_A}^{(n)} = \begin{cases} \frac{|C'|}{\sqrt{\sum_{c \in C'} (s_{p_*}^{(n)} - val(c))^2}}, & C' = \{c_t, r_A \mapsto p_* \in C\} \\ 0, & \text{if } C' = \emptyset. \end{cases}$$

Following, we use the raters' reputation values as a weight on their comments, and therefore obtain an updated score for the provider ratings.

This weight each rater obtains is directly proportional to their reputation value q , and is represented as their share of influence $infl$ on the reputation scores. From the above collection of all raters $\{r_1, \dots, r_l\}$ of the providers $\{p_1, \dots, p_K\}$, we take their reputation values $\{q_{r_1}, \dots, q_{r_l}\}$, and divide 100% of available influence into slices $\{infl_{r_1}, \dots, infl_{r_l}\}$, proportionally to these reputation values. Additionally we introduce what we will label the rating model's *selectivity weight* variable w , which reinforces the selective effect of the model for $w > 1$ and dampens the model effect for $0 < w < 1$. For $w = 0$ the model effect is entirely disabled

and the resulting scores are result of plainly averaging the input comments. This selective weight variable is essential to the utility of the rating model, as we need it to adjust the model's balance between selection and inclusiveness to any certain market scenario. In case the model was run on an empty input set without any comments submitted to the system, we assign all influence values to null, $infl_{r_A} = 0$:

$$\forall r_A \in \{r_1, \dots, r_l\} :$$

$$infl_{r_A}^{(n)} = \begin{cases} \frac{(q_{r_A}^{(n)})^w}{\sum_{i=1}^l q_i^w}, & C \neq \emptyset \\ 0, & C = \emptyset. \end{cases} \quad (4)$$

(2)

Finally, we compute an updated performance score rating $s_{p_B}^{(n+1)}$ for the next iteration $n + 1$ by summing up the averaged comments from 2 multiplied with the influence shares from the previous equation (4). Note that if in the previous equation (4) all influence values have been set to null, $infl_{r_A} = 0$, the resulting scores automatically also are all null:

$$\forall p_B \in \{p_1, \dots, p_k\} :$$

$$s_{p_B}^{(n+1)} = \sum_{i=1}^l (local.s_{r_i \mapsto p_B} \times infl_{r_i}^{(n)}). \quad (5)$$

This concludes iteration n , the next will continue at step (3), with these now updated provider scores and raters' influence values. This iterative process will continue, until the updated differences of the calculated performance scores fall below a convergence threshold δ . Convergence is usually very rapid, and we found a fixed threshold of $\delta = 0.0001$ to be practically suitable in most conceivable scenarios. The algorithm is robust such that if one runs this rating algorithm with no or only one comment as input, the algorithm converges immediately in the first iteration.

(3)

3.1 Confidence Value

If only a subset of raters has submitted ratings on a particular provider, the resulting score of the provider is only based on the comments from these raters. This may mean that a resulting score can be derived from raters who have a low overall rating reputation. There is little to be done about the score in such a case, since there are no other ratings that we have more confidence in. However, from equation (5) we can calculate the sum influence values that we used to compile a resulting score and therefore obtain a confidence value on this score. The confidence value $confidence_{p_B}$ is calculated for every provider, after the algorithm has converged:

$$\begin{aligned}
&\forall p_B \in \{p_1, \dots, p_k\} : \\
&\text{confidence}_{p_B} = \sum_{j \in R'} \text{infl}_j, \\
&R' = \{r_* \in R \mid \exists c_t, r_* \mapsto p_B\}. \quad (6)
\end{aligned}$$

The confidence value $\text{confidence}_{p_B} \in [0, \dots, 1]$ is a measure that states how many per cent of the total accumulated rater reputation participated in deriving the score for this provider p_B .

3.2 Multiple Convergence Points

The algorithm converges very well, but is able to produce more than one convergence point. In this section we first present the convergence properties of the algorithm and then explain the implications of multiple convergence points.

For a scenario of non-convergence to exist at all, it is necessary that in the course of the algorithm's iterations, we reach in equation (5) an influence vector which is identical to a vector reached in a previous iteration. Otherwise, the shifting of influence shares will progressively and eventually converge to a stable set of scores, because the local scores *local.s* are fixed after the initialization and the algorithm cannot avoid convergence without traversing the same influence share vector twice. It is hard to construct a scenario that leads into an infinite cycle, the most promising approach would be to devise a scenario that fulfils the arrow impossibility theorem [2]. However we did not encounter any such scenarios and in general the algorithm converges rapidly, within several dozen iterations. Only when the solution approaches discontinuities, when the solution falls onto an altogether different convergence point, the convergence can take up to hundreds of iterations. The convergence properties are investigated in greater detail in [8].

Depending on the setting of the selectivity weight factor w , any given scenario set of comments has one unique or multiple possible sets of scores to converge to. In fact, for $w = 0$ all scenarios have exactly one convergence point, this being the simple average of all comment values, and for a very high setting of w , as many convergence points exist as there are raters. To reach all these convergence points it is necessary to replace the initialization vector $s_{p_1, \dots, k}^{(0)}$ of equation (1) by the values of the comments of each rater. However, as soon as a second convergence point exists, the results from the reputation model are questionable. The intuition behind this is, that if for the same comments and selectivity weight, we are able to obtain more than one

possible outcome, there is no model-inherent way to determine, which of the outcomes to choose. The motivation for choosing simple averaging as an initialization vector in equation (1) was to speed up the convergence. Therefore we suggest to limit the settings of w for to the range of the unique solution scores, unless one has an application suitable method for selecting one of the convergence points over the others. One such method could be to user customize the scores and to choose the convergence point from the perspective of this user's comments.

If starting from the same averaging initialization vector, and continuously changing the the setting of the selectivity weight factor w , usually also results in continuous shifts of the convergence point. Commonly, when changing from one to the next convergence point, transitions are continuous. However, discontinuous jumps are possible if the two convergence paths are connected by discontinuous gradients. In our extreme example the two voting blocks correspondingly have two convergence points, and scores gravitate to either of the two voting blocks, as long as w is set strong enough to promote such selectivity. In this example, between $w = 1.77$ and $w = 1.78$ the convergence assumes a slightly different slope and then changes the direction of the convergence slope. Figure 1 and 2 show the distribution of influence shares changing from iteration to iteration for the two borderline values of $w = 1.77$ and $w = 1.78$ and display how this transition comes to place. The very same clients who dominate in setting A, are marginalized in setting B and vice versa.

4 Threat Model Analysis

The scenarios in this section contain 100 raters and 20 providers. In these scenarios, the providers are set to perform at a certain average performance level, which is characterized by a numeric value. The goal of our rating model is to identify these performance levels from the comments submitted by the raters. The scenario assumes that raters are not able to capture the performance level of a provider perfectly, because performance is compiled from the quality of interactions over a time period. But these scenarios assume that the providers will on average perform to a certain measurable level, therefore comments are drawn from a random variable with a normal distribution where the mean equals this performance level. While each provider will have a different performance level to show the figures more clearly, one should note that the choice of the actual level is irrelevant to the the rating model, only the deviations from this level are relevant.

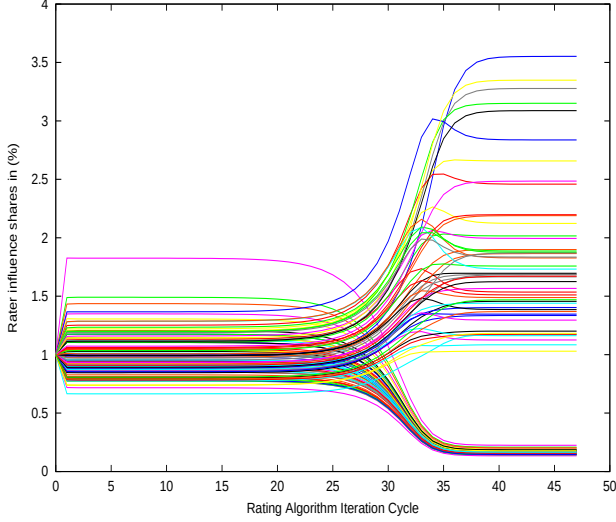


Figure 1. Slope of influence shares for the divergent example at $w = 1.77$, converging to voting block A.

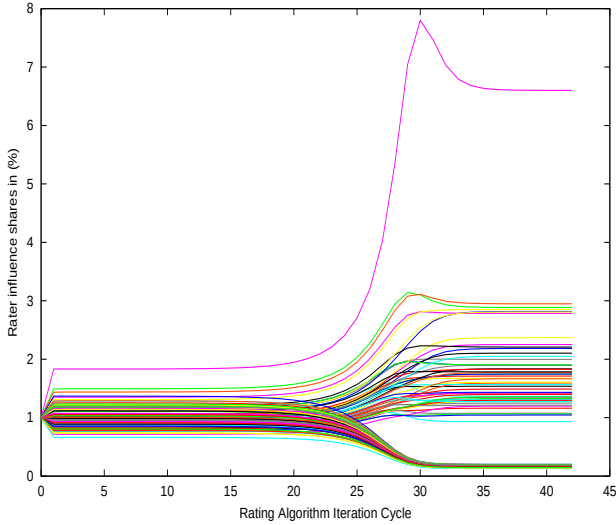


Figure 2. Same example during convergence with $w = 1.78$ and gravitating to voting block B.

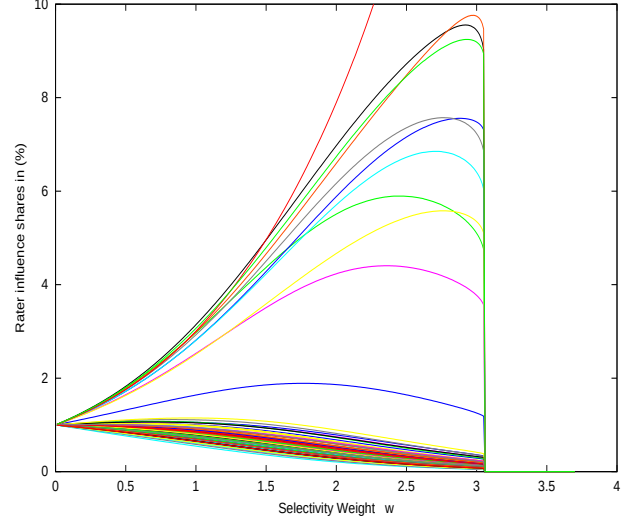


Figure 3. Influence Shares for a Mix of Ten Raters with $\sigma = 0.25$ and 90 Raters at $\sigma = 1.0$.

4.1 Different Rating Accuracy

We want to confirm if the rating model is able to discern raters with a weak rating consistency and set up a scenario with sets of raters of different rating accuracy. One group of 10 raters has a low standard deviation of $\sigma = 0.25$ and a second group of 90 raters has a relatively high standard deviation of $\sigma = 1.0$. In figure 3 we can identify the two distinct bunches of graphs, the top bunch corresponding to the group of low deviation raters and the lower bunch relating to the ten raters with the high deviation. In this scenario, one would choose a selectivity factor of about $w = [2.5, \dots, 3.0]$ in order to put the balance in favor of the ten raters with a low deviation. In doing so, we would ensure that the scores are only drawn from raters with the higher accuracy.

4.2 Malicious Rater Collective Manipulating One Provider's Score

The most important threat model of “bad” ratings is a malicious collective of raters who attempt to influence the score outcome in a coordinated and directed fashion. For such a scenario 40 raters apply a deviating commenting agenda by adding an offset of -2.5^3 to their comment values. If this set of deviating raters would apply their rating bias to all

³The offset value of -2.5 was chosen such that one could visibly recognize the impact of the deviators on the scores, with the combined scores effectively lowered by an offset of -1 . One could also question if a relatively smaller offset value for such a deviation would amount to a threat model or would have to consider such comments valid and rightfully include these in the scores.

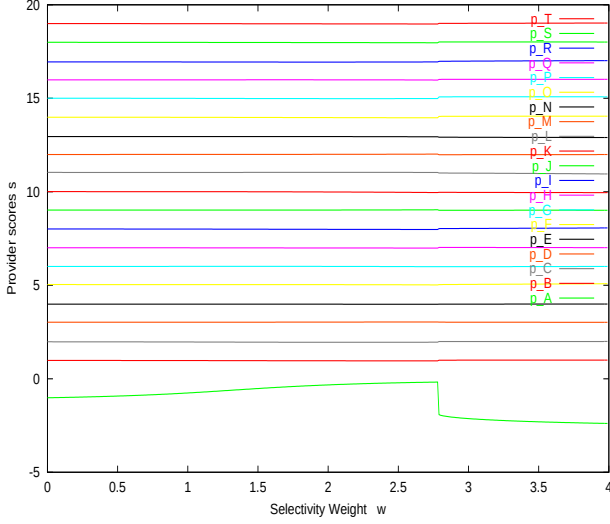


Figure 4. Scores, where all raters agree on all ratings, except for 40%, who disagree on one provider, p_A , and apply a rating offset of -2.5 . For $2 < w < 2.7$ this offset is canceled out by the reputation system.

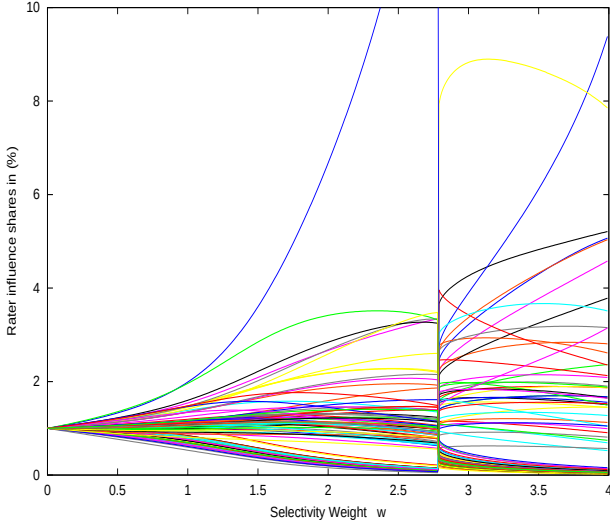


Figure 5. Influence shares for the scenario of figure 4. The 40% raters who apply the rating offset are given by the reputation system a distinctly lower influence in the range of $2 < w < 2.7$ than the other raters.

the rated providers, it would be easier to filter them out. As it is much harder to identify and deselect these deviators if they attempt to artificially alter the rating of only one of the providers, the bias is applied only to provider p_A . In fig. 4 we see how the scores are effected by this deviation, because for $w \leq w < 0.5$ the scores are lowered by one unit from the “correct” values. Increasing w makes the model more selective and eventually for $w > 2.5$ the rating scores reflect the rating pattern of the 60 “good” raters only. From figure 5 we see though, that the gap between the graphs of the 40 deviating raters (represented by the graphs close to 0 in the range of $2 < w < 2.78$) and the 60 other ones is very narrow. This shows that the rating model is pushed to its limit, although it still does achieve the goal of removing the effect of the deviators for a selectivity weight value of about $w > 2.5$.

At this point we would like to determine an appropriate setting for w in this scenario, such that deviators are rightfully disabled, but the other raters form a meaningful aggregation. Therefore we search for a second convergence point by replacing the initialization vector in equation (1) with the comment values of the deviators. Figures 4 and 5 actually show the model results drawn from the altered initialization vector that is favoring the deviators. At $w = 2.78$, the second convergence point is available, the scores follow a discontinuous jump to the scores of the deviator group and the influence distribution is reversed for both groups. With the existence of the second convergence point, the solution is not unique and it could be debatable which of the convergence points should be chosen. Therefore we choose to disregard results that offer two convergence points.⁴ Within the range of possible unique model solutions, we want to maximize the rating model’s selectivity effect in order to enable the model to filter out the deviating raters. Choosing $w = 2.78$ satisfies both of these criteria and yields us model solution scores that are practically free of influence taken by the deviators, which at this point collectively only amounts to 4.41%.

4.3 Deliberately Inconsistent Provider Performance

The scenarios we discussed in the previous sections assume that providers display a consistent performance behavior. This does not imply that they have to deliver identical performance every time, but we assume that clients base their ratings on a number of transactions with a service and that

⁴We can choose to take into account results that allow for more than one convergence point, if we have a method for choosing the desired initialization vector to reach the desired convergence point. One such method is to take a client-specific view, and select the convergence point according to this client’s comments.

statistically the clients experience on average a similar performance from a provider and that this average is characteristic for this provider. Deriving a reputation with the model under such assumptions for a provider who displays a high variability in his performance would still result in the scores correctly reflecting his average performance.

However, what happens if a provider does not deliver a varied performance by statistical chance, but rather has opted to treat a specific set of clients with a specifically worse or better performance level? For example, provider p_A could decide that he delivers a guaranteed-level 99.99% of the agreed performance terms to a majority of 75% of preferred customers and delivers as low as only 50% of the agreed performance terms to the minority of 25% of the remaining contracting clients, when he needs to use these clients as a best-effort-served buffer for his performance demand variances. With the clients then submitting their observations as comments, this would result theoretically in a performance score of 87.49%, if applying a plain average. We would argue that this does not represent the performance of the provider well at all. Such a result neglects the good and the bad of this provider's strategy.

In order to make this scenario statistically more realistic and relevant, the majority group experiences a normal distribution with no performance above 100% and a very low standard deviation of $\sigma = 0.25$, simulating the guaranteed performance level. The minority group's normal distribution of comments on the other hand will never fall below 50% and displays an extremely high standard deviation of $\sigma = 25.0$, reflecting the best effort characteristic of their service experience.

Figure 6 compares the model's score outcomes for the two possible convergence points of p_A 's rating and practically, with the statistical circumstances mitigating p_A 's performance bias, a plain averaging of comments results in a flattering rating score of 91.4%. The remainder of this scenario contains again 100 raters, with 25 reporting the lower performance and a total of 20 providers, where the other 19 all behave consistently and predictable. While the first convergence point is found from the overall average as the initialization vector, the second is derived from an initialization vector formed from the comments the minority group is submitting. As in the previous scenarios, as soon as the selectivity weight is high enough such that the resulting scores are not compromising between the two groups anymore, the second convergence point appears, that is for $w \geq 1.85$. At that point we can clearly identify the two general performance levels (99.99%; 50%) this provider is supplying to the two groups of clients. We can individually identify which of the clients belong to either of these two groups by observing the exact reversal of the influence shares held by either group for the different convergence points. Ironically

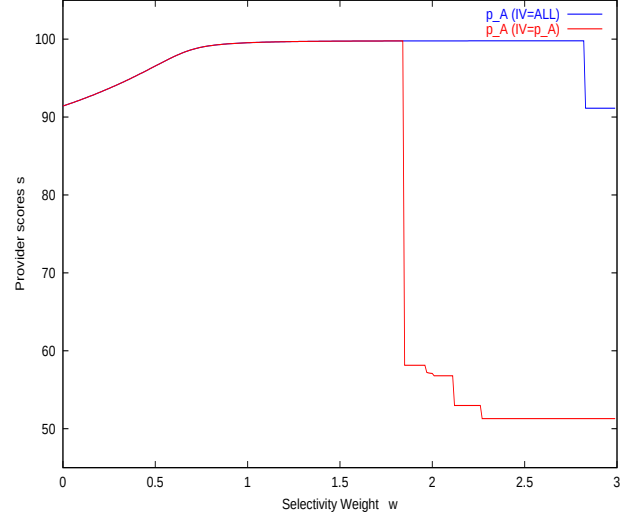


Figure 6. Scores for both convergence points where provider p_A is deliberately delivering inconsistent performance.

though, when increasing the selectivity weight, eventually for $w \geq 2.83$ the resulting scores center on one of the minority group raters who happens to be very close again to the plain average which we obtained with $w = 0$.

From the point of view of the scores, this scenario is identical to a set of clients with a rating bias, as we described in section 4.2. By analyzing the submitted comments it is not possible to distinguish between a provider actually treating a client differently than others or a client turning out a different rating than the performance he has received. The model however is able to reveal these divergences, particularly if these are part of a general pattern and not random flukes. However, to tell which of the both cases of deviance we are dealing with is external to this rating model and needs to be resolved with other methods.

5 Related Work

The idea to introduce reputations for raters in our reputation aggregation algorithm was inspired by the way page rankings are calculated in Google (transfer of endorsement[9]), and has been picked up by other reputation system research in various ways. Chen and Singh[4] integrated this in their versatile reputation system, which allows for either bi-directional or unidirectional ratings and is able to integrate plain text comments from raters. Chen and Singh's research is closely related and supersedes ours in its functionality. However, our algorithm and the threat models are more specifically tailored to the uni-directional recommen-

dation case of client-server oriented business scenarios.

Collaborative filtering systems are reputation systems that use similarity-based approaches to provide a function that aggregates recommendations of other clients in relation to a specific client's preferences. Pennock et. al. [10] analyze the theoretical foundations of collaborative filtering systems under the aspect of social choice theory. Dellarocas [5] presents a cluster filtering algorithm, which he evaluates in terms of its ability to eradicate the effects of unfair ratings.

For peer-to-peer reputation systems, Kamvar et. al. [7] develop the EigenTrust algorithm, which uses a similar trust propagation concept, transfer of endorsement, which recursively feeds the trust information back into the aggregated scores. Given that EigenTrust has been designed to suit a more specific application than our reputation system, it is interesting to note that both are able to identify similar sized proportions of malicious peers.

Bayesian estimation is another approach to classify sets of agents by their behaviour [1][3]. Buchegger [3] develops a routing protocol for mobile ad-hoc networks, which uses a modified Bayesian estimation procedure to eliminate false information among the second-hand reputation information and to isolate misbehaving nodes. The main difference between our approach and Buchegger's is that Buchegger calculates the reputation aggregation from the viewpoint of an individual node, where our approach is the one of a service that does not use first-hand information. Being able to use first-hand information strengthens the filtering of misinformation, but it always makes the resulting recommendations partial to the collector of the first-hand information.

6 Financing the Operations of the Reputation System

Operating such a reputation system needs to be funded in some way. However, the way we choose to fund the reputation system has implications on the incentives put forth by the reputation system. With an reputation system we aim to resolve some of the market inefficiencies, which presumably would yield benefits that could be tapped for refinance. Clients benefit directly from the higher reliability of services from better providers, and the providers benefit from the market's ability to drive "lemon"-providers out of business and the therefore generally higher confidence by the clients in the services market. Some providers may benefit by charging a premium for delivering service with higher reliability.

Possibly the easiest route to obtain the funds to run the reputation service would be to demand subscription fees from the listed providers, as these could afford to write off

the fees as a form of advertisement for themselves. The other incentive for the providers to pay for a subscription would be to lock out competitors out of the market, as they now would be identifiable through the reputation system. Effectively they would be forming a guild of "accredited" providers and thereby maintain a clean market for the clients. However, we would choose against funding the system by the providers, as this would create the wrong incentives for the operation of the reputation service. The operators of the reputation system would lack the incentive to promote transparent performance comparisons between providers and thereby promote competition among them, but would have an incentive to set a high bar for the level for market entry. Our reputation system seeks to aid the clients' choice to make the market in general more efficient, and a provider funded system would naturally follow different goals.

The economically cleaner solution would be to demand payment from the clients and thereby provide a direct incentive for the reputation system to cater to the needs of the clients' choice. A client funded reputation system does not have an incentive to alter the objectivity of its provider reputation rankings and if it is subject to competition itself, where clients choose a reputable reputation system, the reputation system also has a disincentive to accepting bribes (or advertisement) from the providers. However, demanding clients' to pay is likely to meet their resistance, as it can be expected that some of the clients rather not use such an advisory service at all than having to pay for it, even if such could be shown to pay off in the long run. Moreover, clients could undermine this revenue model by forwarding or even publicizing the reputation system's ranking lists. Further, the actual pricing model would be a sensitive issue, as a fixed fee might be too high for some small one-time client and yet negligible for a high-turnover one. Addressing this through discriminatory pricing schemes is likely to introduce distortions and using micro-payments that scale proportionally with the volume of use is likely to introduce unreasonably high transaction overheads. A specific problem is system startup, as the value of the reputation system increases along with the number of gathered clients' review comments. Therefore, one could not use the client funding to cover the startup phase. On a general level, it may turn out that the clients' individual per-transaction-profit from using the reputation system is not sufficient to cover the running cost of the system, although amortized they themselves or the whole market would profit. This is not an unlikely case and one common solution is to call for a government sponsored funding solution.

Where a system's net benefits need to be amortized either over time or aggregated over a number of market participants, it is common to seek for government sponsoring, as

the system meets the criteria to be considered an infrastructure element. It is our anticipation that the reputation system would be such a candidate due to all the complications stated in our discussions about client funding and rejecting provider funding on the ground of incentive incompatibility.

Another possible solution is the mix of government and client funding. In that case one would draw on client charges for enquiries about high stakes services that yield client benefits clearly outweighing the charges and government funds for the remainder, in particular any startup periods. In the UK's energy market, the government supports such price comparisons in order to make the market more dynamic and transparent. Even in the absence of direct government funding for a reputation system, it may well be necessary to introduce government regulations that protect the reputation reporting system. Otherwise providers not only can lack cooperation with such a reputation system setup, but could actively obstruct its work, for example by demanding from their clients not to disclose their experiences that stem from using the provider's service. Such obstructions would most likely be expected from providers who have built their reputations through other means, such as advertising, or sheer size and now attempt to protect their advantageous profile.

7 Conclusions

At a pragmatic level, we showed that our reputation aggregation system is able to counter three hard threat scenarios. The algorithm highlights the difference between raters with a high and a low rating variance and is able to shift influence to the raters with a low rating variance, even if the vast majority of raters has a high variance. Further, we show that the model is able to filter out a malicious collective of raters, even if this group singles out only one provider and is close to being the majority. The third model demonstrates how the model can be used to identify unfair behavior on the part of the providers.

We were able to demonstrate that the rating model is able to find the largest subgroup of raters with the closest agreement and through this to filter out various forms of undesirable rating behavior. Most commonly, we recommend to set the selectivity weight variable w to the maximum setting that leads to a unique solution (ones that have only one convergence point). However, it is possible to adapt the reputation model's aggressiveness as a selective device to application-specific demands, which is explained in greater detail in [8].

The reputation aggregation model is very versatile, and so has been applied to the academic peer review process.

Chapter 8 in [8] demonstrates on the example of a conference review, how the model is able to numerically capture the notion that some reviewers are better at contributing to consensual review results than others. We show how the review results can be gracefully adjusted to reflect the quality of the reviewers, without lowering too much the confidence value of the resulting review scores. This example also displays the scalability of our reputation model, which is able to produce useful scores even when applied to a small number of entities with peculiar characteristics.

References

- [1] K. Aberer and Z. Despotovic. Managing Trust in a Peer-2-Peer Information System. In *Proceedings of the Ninth International Conference on Information and Knowledge Management (CIKM)*, 2001.
- [2] K. J. Arrow. *Social choice and Individual Values*. Yale University Press, 1963.
- [3] S. Buchegger. Coping with Misbehavior in Mobile Ad-hoc Networks. PhD Dissertation, Ecole Polytechnique Fédérale de Lausanne, January 2004.
- [4] M. Chen and J. P. Singh. Computing and using reputations for internet ratings. In *Proceedings of the 3rd ACM conference on Electronic Commerce*, October 2001.
- [5] C. Dellarocas. Immunizing online reputation reporting systems against unfair ratings and discriminatory behavior. In *Proceedings of the ACM Conference on Electronic Commerce*, pages 150–157, Minneapolis, Minnesota, USA, 2000.
- [6] E. Giovannetti and C. A. Ristuccia. Estimating Market Power in the Internet Backbone Using Band-X data. Cambridge Working Papers in Economics, Department of Applied Economics, University of Cambridge, June 2003.
- [7] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina. The eigentrust algorithm for reputation management in p2p networks. In *Proceedings of the Twelfth International World Wide Web Conference (WWW)*, Budapest, Hungary, 2003.
- [8] J. H. Lepler. Cooperation and Deviation in Market-Based Resource Allocation. PhD Dissertation - <http://www.cl.cam.ac.uk/TechReports/UCAM-CL-TR-622.html>, Cambridge University, November 2004.
- [9] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.
- [10] D. M. Pennock, E. Horvitz, and C. L. Giles. Social choice theory and recommender systems: Analysis of the axiomatic foundations of collaborative filtering. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence (AAAI-2000)*, 2000.
- [11] A. Sen. *Handbook of Mathematical Economics*, volume 3, chapter Social Choice Theory. Elsevier Science Publishers, 1986.

Scalability and Robustness of a Market-Based Network Resource Allocation System*

Nadim Haque, Nicholas R. Jennings, Luc Moreau
School of Electronics and Computer Science
University of Southampton
Southampton, UK.

{n.a.haque, n.r.jennings, l.moreau}@ecs.soton.ac.uk

Abstract

In this paper, we consider issues related to scalability and robustness in designing a market-based multi-agent system that allocates bandwidth in a communications network. Specifically, an empirical evaluation is carried out to assess the system performance under a variety of design configurations in order to provide an insight into network deployment issues. This extends our previous work in which we developed an application that makes use of market-based software agents that compete in decentralised marketplaces to buy and sell bandwidth resources. Our new results show that given a light to moderate network traffic load, partitioning the network into a few regions, each with a separate market server, gives a higher call success rate than by using a single market. Moreover, a trade-off in the number of regions was also noted between the average call success rate and the number of messages received per market server. Finally, given the possibility of market failures, we observe that the average call success rates increase with an increasing number of markets until a maximum is reached.

1. Introduction

Resource allocation is a central problem in effectively managing networks. Specifically, this covers the process by which network elements try to meet the competing demands that applications have for network resources — primarily link bandwidth and buffer space in routers or switches [12]. This is a challenging problem since resources become scarce when there is a high demand for them. In this work, we consider a circuit switched meshed network where nodes communicate with their immediate neighbours

using radio links [11], where by design, they consume as little power as possible and are targeted for rapid and cost-efficient deployment in poor countries. Such low power consumption implies that there is limited bandwidth available in the network. In this context, we provide an empirical evaluation of a multi-agent system that allocates end-to-end (source-to-destination) bandwidth in such communications networks to set up calls. More specifically, based on our previous solution [7], in which we developed an application using markets, we consider a network which is partitioned into regions, each with a market server, from where resources are allocated. Using regions and decentralised markets in this way means that there is no central point of failure from where all resources are allocated. In more detail, we look at the region scalability in a fixed size network, as well as robustness where failures are induced in the network. That is, we evaluate how the system performance changes with varying number of markets. By investigating these issues, we provide a network designer with an insight into how such a network can be deployed in practice.

In recent years, market-based approaches have been investigated and used to solve various problems in areas of computing, where applications and systems have also successfully been developed. Areas in which markets have been studied include allocating resources in computational grids [1], development of peer-to-peer systems [14], supply-chain management [10], scheduling [17], congestion control [8], routing [6], workflow automation [9] and recommender systems [16], amongst other areas. The solution that we developed consists of software agents that compete in a marketplace to buy and sell bandwidth. Our previous work [7] described the system and indicated the broad feasibility of our approach. Here, buyer agents represent callers and seller agents represent the owners of the resources. We decided to base our solution on agents for a number of reasons. First, their autonomous behaviour allows them to carry out their tasks in the decentralised control regime

*The research in this paper is part of the EPSRC funded Mohican Project (Reference no: GR/R32697/01). We would also like to acknowledge the contribution of Steve Braithwaite who provided us with domain expertise.

of distributed marketplaces. Second, the reactive nature of agents is needed to respond to requests quickly so that calls within the network can be made with minimum delay. Third, agents have the ability to flexibly interact which is important in our system because the agents need to bid against a variety of different opponents in an environment where the available resources vary dynamically. A market-based approach was chosen for the following reasons. First, markets are effective mechanisms for allocating scarce resources in a decentralised fashion [2]. Second, they achieve this based on the exchange of small amounts of information (such as prices). Finally, they provide a natural way of viewing the resource allocation problem because, generally speaking, they ensure the individual who values the resources the most will obtain them.

Our system in [7] was a distributed market mechanism in which allocations of *interrelated* resource bundles were sold in multiple markets. The marketplace protocol incorporates a reservation and commitment mechanism that provides a guarantee that resources will not be bought unnecessarily. Now in this paper, we extend our previous work by analysing the system behaviour after performing scalability and robustness tests, with respect to increasing the number of regions in a fixed size network. The remainder of this paper is structured as follows: the marketplace design and components are recapped in section 2. The system evaluation and experimental results are presented in section 3. Section 4 describes related work and, finally, section 5 concludes.

2. Marketplace design

This section describes the design of the system. Specifically, the basic components and the network model are outlined in section 2.1. Section 2.2 describes the constituent agents and, finally, section 2.3 provides a brief description of how resources are acquired in a multi-region call.

2.1. Network model

The system consists of three types of agents: seller, buyer and auctioneer (see figure 1). Seller agents are responsible for selling node bandwidth capacity resources and buyer agents are responsible for buying these resources. The auctioneer agents accept *asks* from seller agents and *bids* from buyer agents and conduct auctions so that resources can be allocated using a market-based protocol. As can be seen in the figure, the overall network is divided into a number of regions (3 in this case). Callers are used to initiate calls via the use of handsets. When a call request takes place, the destination location to where the caller wishes to make the call is passed to the buyer agent on the local node. This agent then starts the process of setting up the call. For each

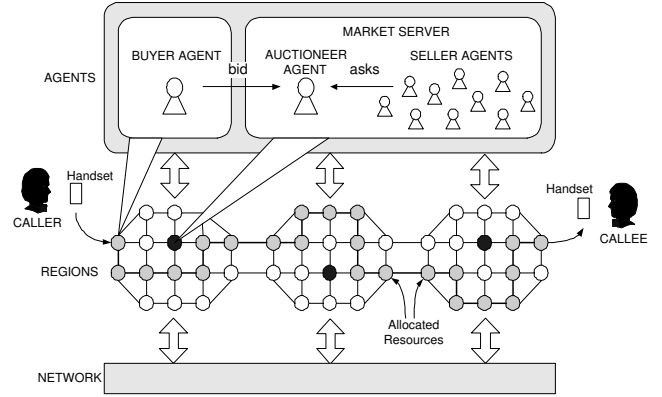


Figure 1. An overview of the system architecture. Black nodes in regions represent market servers and grey nodes represent allocated resources for a particular call from the caller to the callee.

call attempt, a buyer agent in each required region tries to reserve a resource bundle (i.e. set of interrelated resources in a single region) from its local market server. Buyer agents work together to collectively make a complete source-to-destination path across the regions using the bundles. If a resource bundle cannot be obtained, then a backtracking mechanism is used which allows alternative allocations to be made if currently reserved bundles cannot lead to the final destination.

It is desirable for resources to be bought and sold in the network from various points and not from a central location (since a single server would constitute a central point of failure). With this in mind, we partitioned the network into regions where only resources within those regions are sold (i.e. there are multiple market servers in the network, one placed in each region). Thus, if we are to look into the scalability of the number of regions within a fixed size network (i.e. the region scalability), we must consider how the network should be divided. Here we regard a network region as a group of nodes that are situated geographically close together. In this application, we consider a static network configuration as defined at deployment time. Nodes on the edge of regions can communicate with other edge nodes in neighbouring regions. In using local markets, resource information does not have to be replicated across all markets in the network. This is good since the market server that receives a bid from a buyer does not need to contact *all* other markets to make sure that the same resources are not being sold elsewhere, for each bid placed.

To model the network, each node has a fixed total bandwidth capacity that is split logically into several *equal* parts, where these are the resources that are bought and sold by

the market mechanism. These resources are used in relaying several calls at the same time through the nodes. Each node has a fixed number of handsets attached from where calls originate. A handset that is currently in use is assumed to be engaged and, thus, cannot be used for any other calls at the same time. Our current work assumes that control capacity is separate from the bandwidth capacity used for relaying calls. The resources we consider are for calls and not for control messages. In this work, we do not look into the usage of control capacity and leave this investigation for future work. However, in section 3.3, we do look at the number of messages received per market server, since it is on these servers that the majority of the processing takes place for running the auctions.

2.2. The agents and the markets

Auctioneer agents conduct auctions using a combinatorial reverse auction protocol [15] to allocate goods (units of node bandwidth) to buyers (i.e. they allocate a *combination of goods* that consist of the cheapest possible bundles). There is one auctioneer agent per region in the network, each on their respective market server nodes. Auctioneer agents execute a *winner determination protocol* that determines which resources are allocated to which parties, for each bid submitted.

There are several seller agents per region, one owning each node, where they each submit an individual ask price to their local markets. The implication of each seller agent owning a node is that they can attempt to compete against each other by pricing their respective resources competitively. To minimise communication in the network, all seller agents are physically deployed on their local market server nodes and we assume that, currently, they all use the same pricing strategy. A seller agent begins with a total of y resource units initially priced at one price unit each. For each unit sold, the price increases by one price unit (i.e. when there is only one resource unit left, it should cost y price units). Conversely, for each unit reclaimed by a seller, the price reduces by one price unit.

The initial low price of one price unit is chosen so that sellers can sell resources more easily to begin with. As demand for resources increases, the price per unit increases so that buyer agents have to bid more for resources. Given this, seller agents can maximise their utilities by making as much profit as possible. They also reduce the price of resources by one price unit when they have reclaimed the resource so that they can lure more buyers to purchase resources from them in the future. This allows seller agents to remain competitive against each other when pricing their resources.

A buyer submits a bid composed of several bundles, of which only one is required. The winner determination algorithm then attempts to allocate resources by minimising

the amount spent. From these bundles, the cheapest available one is allocated to the buyer agent. If a buyer agent's bid is successful, resources are sold at the asking prices of the seller agents. There is one buyer agent placed on each individual node where they await call requests, from callers, from any point in the network. We assume that all buyer agents use the same purchasing strategy. Thus, when a buyer agent receives a request for purchasing node bandwidth, it formulates its bid. It attempts to find the cheapest set of routes that lead from its current node to a destination node within its own region. Now, in this work, we assume that buyer agents select a set of bundles that minimise the length of their desired routes. The intuition here is that the buyer believes shorter routes are generally cheaper since they contain fewer resources.

We make the assumption that buyer agents are only allowed to submit up to a certain number of bundles for each bid. The value chosen here must be enough to allow some flexibility in the bundle that a buyer could be allocated, but it should not be so high that the market algorithm has to do significant amounts of unnecessary processing. If the final destination node is within the same region, that node is the destination node. The bundles selected by a buyer agent are sent as a bid to the buyer's local market. Finally, if the buyer agent is successful in reserving resources, it is informed by the local market. Callers are assumed to pay a fixed amount per region for calls that are successfully established where the cost per region is proportional to the number of resources in the region.

2.3. Acquiring resources across regions

The number of buyer agents required in setting up a call is the same as the number of regions in which resources are required for a given call. If the final destination for a call is in a different region from the one in which a buyer is located in, then this buyer will attempt to find routes that lead to a node within its region that is connected to a node in a neighbouring region that leads to the final destination. Then, in a multi-region call, once a buyer agent has successfully reserved a bundle of resources, the market server in that region is responsible for contacting a buyer agent that is on the edge of the next region. The node on which this second buyer agent resides must be in reach of the last node in the bundle of resources that have been reserved in the previous region. Thus, these boundary nodes will relay the call from the initiating source node to the final destination node, such that there is a continuous path when/if the call eventually takes place.

Figure 2 illustrates part of the reservation process. This figure shows a buyer agent, b_1 , which has already successfully been allocated a resource bundle (shown by the grey nodes) by its local market server within its region (region 2

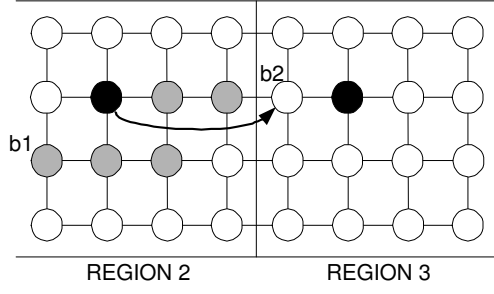


Figure 2. A market server in region 2 (black node) contacting a buyer agent, b2, in the following region for setting up a multi-region call. Grey nodes represent a reserved resource bundle for b1 in region 2.

in this case). This market server in region 2 then attempts to contact another buyer agent, b2, which is on a boundary node in the following region so that b2 can then bid for a resource bundle within its own region (region 3). The reservation process continues in each required region until the final node is reached and the call is set up. In general, the system algorithm allows buyers to choose which region to contact next when there is a choice of following regions in a multi-region call. In such a case, the regional route that involves the use of the fewest regions for the set up of a call is preferred (i.e. the shortest regional route).

Once the final destination has been reached, the market server in the last region sends a *commit* message to the buyer agent within its own region. This buyer agent then contacts the market server in the previous region which, in turn, informs its buyer agent and so on, until the initial region is reached. Eventually, the originating buyer agent receives the commit message and the call can be placed. Thus, there is a reservation and commitment process that takes place in the system, where payment for resources only takes place during the commit phase once all of the necessary resources have been acquired. When the call has completed, a message is sent from the initial buyer to its local market (and to all other markets and buyers involved in this call in the direction of the final region) to signal that resources can be released. The markets then resell the resources to buyers that place bids for them in the future.

The system uses a backtracking mechanism that allows alternative allocations to be made if currently reserved resource bundles cannot lead to the final destination. Thus, if a buyer agent in an intermediate region fails in reserving a bundle of resources, it can resubmit another bid to its local market which contains bundles that lead to another destination node within its own region (i.e. to a different boundary node). This continues until a bundle has been re-

served or there are none available. Thus, agents perform a distributed search for resource bundles. An example of the backtracking mechanism used to find alternative routes via different boundary nodes and regions was given in our previous work [7].

3. Experimental evaluation

This section describes the experimental work that was carried out in evaluating the scalability of the system with respect to increasing number of regions as well as the robustness of the system. Section 3.1 describes the methodology and parameters used, while results are outlined in sections 3.2 to 3.5.

3.1. Experimental methodology and settings

Previously [7], we looked at the initial results of the system algorithm (i.e. the *average call success rate* and *average call set up time* when compared against optimum and random strategies). This provided us with an insight into a fundamental measure of the percentage of successful calls and set up times, respectively, given one particular network setting. However, the aim of our current work is to evaluate the system in terms of region scalability and robustness. Looking at the former, will give us an understanding of how the structuring of the network into regions impacts the performance for a given network size. Testing for the latter, will show how well the system performs with market failures.

In more detail, to test for scalability, we measured the *average call success rate* when progressively increasing the number of regions in a fixed size network (see section 3.2). This was chosen as a measure because it provides a fundamental insight into the percentage of calls that are successfully established from all calls attempted, with call establishment being the primary aim of our system. We also look at the number of messages received per market server in the network when scaling up the number of regions (described in section 3.3). This is also important since this is where resources are auctioned and thus, where the majority of processing occurs. Thus, these are two measures that a network designer would be interested in considering when deploying a network. By obtaining results from these two experiments, we then look at the trade-off between the average call success rate and the number of messages received per market server in order to find an optimum number of regions in which the network should be partitioned into (see section 3.4).

Finally, the average call success rate was also measured for robustness testing, after a market failure was introduced within the network (described in section 3.5). Again, as with scalability testing, the average call success rate is an equally applicable measure to investigate for robustness

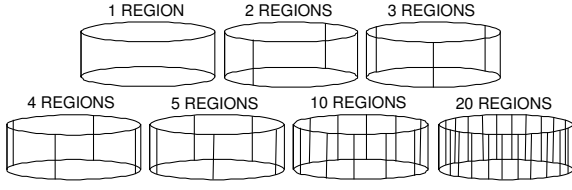


Figure 3. Set of seven tori used where each consists of an 80-node (20-by-4) network but is split into a different number of regions.

testing, since looking at the successful establishment of calls is just as important given a market failure. For the sake of consistency, we chose to investigate a *single* market failure throughout all of the robustness testing in this particular experiment. Here, we assume that all buyers have knowledge of the failed market server and, thus, we do not deal with failure detection. In all experiments discussed in sections 3.2 to 3.5, the network load was increased by varying the *call origination probability* (i.e. the probability of a call originating from any given unused handset).

For our experiments, several different network set ups were used. In each set up, the same underlying network topology was used but it was partitioned into a different number of regions with a market server in each region (market servers are placed manually within a central location in their regions where there is a high connectivity of neighbouring nodes). Thus, all of the experimental set ups use an 80-node (20-by-4) network. This was chosen because it demonstrates a topology which can be partitioned vertically with ease into several regions such that it is easier to evaluate the system for scalability. For each set up, the network was wrapped around and shaped into a taurus, as shown in figure 3. Also, all calls made were unidirectional (i.e. calls travel in only one direction around the taurus). The combination of joining the beginning and end of the network to form a taurus and allowing calls to traverse the network in one direction ensures fairness in our experiments for two reasons. Firstly, the number of types of different distance calls made in any set up are the same. For example, given the five region set up, there are exactly five different types of calls that require a single region, five different types that span across two regions, five types of triple region calls, and so on. Secondly, by using a taurus, the average load in any region is the same, since there is no central region through which any extra calls traverse. This provides an even setting for testing the system.

The experimental settings we used in this evaluation were obtained from a domain expert. Specifically, each experiment was run for a total of 100,000 time steps and

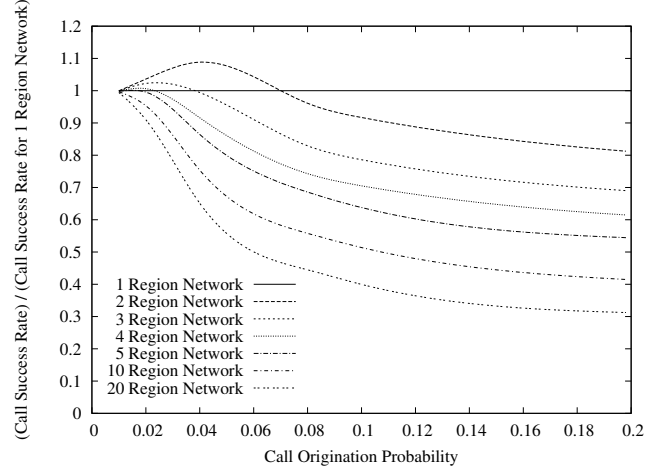


Figure 4. Average call success rate over the 1 region set up average call success rate, as the call origination probability is varied.

probed after every 1,000 time steps. The duration of a call was set to 1,000 time steps. We assume that each node has 2 handsets attached to it and has a total of 10 units of bandwidth capacity available, allowing each node to relay up to 10 simultaneous calls at any one time. Calls were made to originate after every 50 time steps. Buyer agents were allowed to submit bids that contained a choice of 5 bundles from which an attempt was made to allocate the cheapest one available. Finally, the number of simulation runs for each experiment was sufficient for the results to be *statistically significant* at the 95% confidence level.

3.2. Average call success rate

The purpose of the first experiment was to investigate the proportion of calls that could successfully be connected for each network set up, shown in figure 3, when varying the call origination probability. (The network is regarded as being heavily loaded when the call origination probability is 0.2 (20%). This gives a typical network occupancy of 82%). As can be seen from figure 4, in general, call success rates are higher when there are fewer regions. (For the sake of clarity in figure 4, all call success rates are divided by the 1 region call success rate). However, another observation can be made from figure 4. Given a network load that is light to average where the call origination probability is below 0.07 (7%), we can see that the 2 region set up gives a higher call success rate than the single region case of up to 10% more. Also, when the call origination probability is below 0.04 (4%), the 3 region network set up provides a higher average call success rate than the 1 region centralised case,

by about 3% more.

The reason for our observations can be explained as follows. When the network is light to moderately loaded (i.e. when the call origination probability is below 0.07), there are sufficient resources available for setting up a large proportion of the calls. For the multi-region set ups (i.e. when there are 2 or more regions in the network), resources need to be acquired across *several* regions. Here, the search for resource bundles is exhaustive across boundary nodes in intermediate regions — buyers will continue to bid until either a bundle is found or there are none available, except in the final region where only a single bid is made to the final destination node. The search for resources in the 1 region set up is similar to the search that takes place in the final region of a multi-region call (except that on average, the required resource bundle is larger in the 1 region case because the size of the region is larger). Thus, no exhaustive searching takes place for resource bundles across boundary nodes for the 1 region case since there is only a single region and, therefore, if the initial attempt fails, then the call set up is unsuccessful.

When the call origination probability increases, resource contention increases for all network set ups (i.e. node bandwidth becomes more scarce). For multi-region cases, the search becomes more exhaustive, where resources are reserved for longer periods of time, while a call is being set up. Consequently, this reduces the chance of other calls being made. This is not an issue with the 1 region network and therefore, it is only at this point that the single region network set up begins to perform better than all of the multi-region network set ups.

3.3. Market server load

Our next experiment investigates the number of messages received per market server, as the number of regions is scaled up. There are several different types of messages that are received by the market servers. These include, buyer bids, seller asks, commit messages and messages that tell the market servers to release resources when the resource usage is complete. We do not differentiate between these, but rather simply sum them across all market servers and divide by the number of market servers, for each network set up, to find the average number of messages received by each one. Our hypothesis was that there would be a lower number of messages received by a market server, per simulation run, as the number of network regions increases. This is important because if market servers receive fewer messages, the load on a server is less and, therefore, less auction processing needs to take place.

Figure 5 shows that our hypothesis is true and that when there are more market servers in the network, on average, the number of messages received per market server does

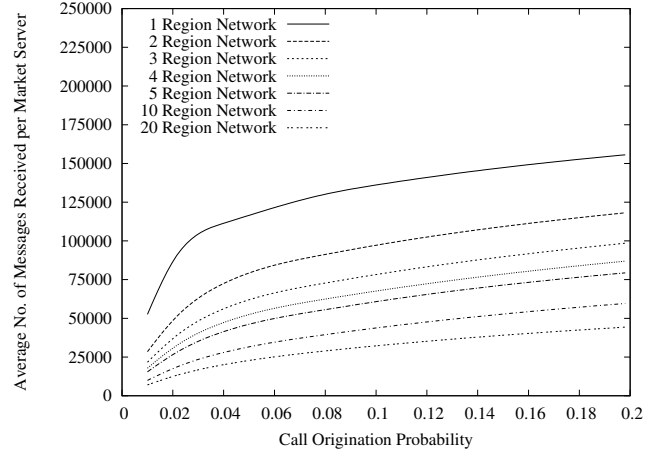
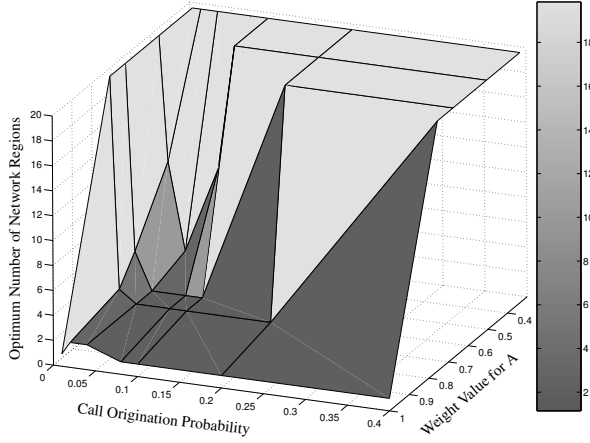


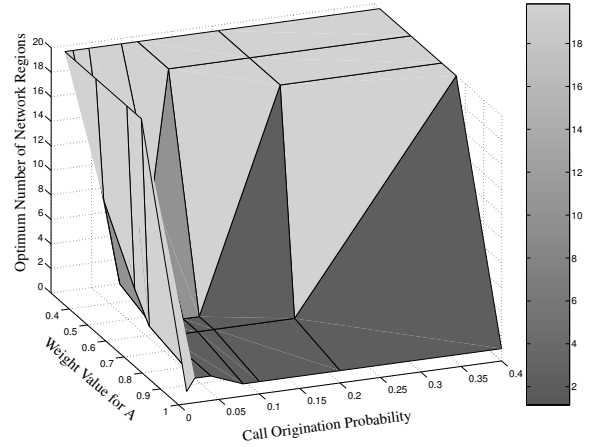
Figure 5. Average number of messages received per market server as the call origination probability is varied.

indeed decrease. Using a 1 region network, with a call origination probability of 0.1, 136,000 messages are received by the single market server whereas, in contrast, with 2 regions, there are 97,000 messages per server and with 20 regions, there are just 32,000 messages per server. A similar pattern is observed at all call origination probability values. The reason for this result can be explained as follows. As the number of regions increases, the number of nodes per region decreases, since the overall size of the network remains fixed. Since buyers and sellers submit bids and asks respectively, only to their own local market servers, the number of such messages becomes less per market. Thus, with more regions, the number of messages received per market is less (i.e. the load is distributed more amongst market servers, which is desirable, since less processing is required per server).

Whilst the total number of messages received by *all* market servers in each network set up is higher with an increasing number of regions, this sum of messages does not increase by a disproportionate amount when, say, the number of regions doubles. For example, at a call origination probability of 0.04, there is only a 35% increase in the total number of messages received by all market servers from the 5 region network set up to the 10 region case. Also, at a lower call origination probability of 0.01, there is only an 8% increase in the total number of messages from the single region case to the 2 region network set up. However, it still remains the case that the average load per market server decreases with increasing number of regions and this is a useful result, since it is the processing *per* market server that we wish to minimise.



(a) 3-Dimensional, left-sided view



(b) 3-Dimensional, right-sided view

Figure 6. Optimum number of regions, as call origination probability and weight value for A are varied.

3.4. Optimum number of network regions

Our result from figure 4 shows that, given specific call origination probabilities, the average call success rate is generally higher when the network is partitioned into a smaller number of regions. Alternatively, figure 5 showed that the number of messages received per market server decreases per market when there are more regions in the network. Receiving less messages is better because fewer auctions need to take place and this, therefore, requires less processing power and fewer input buffers. Thus, figure 4 (average call success rate), shows that less regions is better and figure 5 (messages received per market server), indicates that more regions gives a better performance.

Given these results, we would like to find the trade-off between the average call success rate and the number of messages received per market server. Our motivation is to help engineers in deploying such a network to find an optimum number of regions in which to partition the network. In order to achieve this, we first normalise the y-axis on both figures 4 and 5 so that they both range between 0 and 1, where we consider these as the utilities for each measurement. (In the case of figure 5, we subtract each normalised value from 1 to obtain the utility values, since receiving fewer messages per market server should give a higher utility). Then, for each of the network set ups and varying call origination probabilities, we consider the overall utility defined as the weighted sum of the utilities of the average call success rate and the number of messages received per market server: $A \times u(c) + B \times u(m)$ such that $u(c)$ is the utility of the average call success rate with weight A, $u(m)$ is the utility of the number of messages received per market

server with weight B and where $A + B = 1$. The values for A and B reflect the importance that a network designer would assign to the two different measures. Figure 6 shows a 3-dimensional surface plot where the optimum number of regions is plotted against the call origination probability with varying values of weights A and B. For the sake of clarity, two different views are shown of the same plot in figure 6. Several observations can be made from this figure.

When the network designer believes that receiving fewer messages per market server is more important than the average call success rate in the system (i.e. when $B > A$), the optimum number of regions to deploy is 20, across all call origination probabilities. This observation can be explained by the fact that a lower value for A means more importance is given to the number of messages received per market server, where the utility is higher when there are more regions.

When the network designer gives an equal importance to the average call success rate in the system and to receiving fewer messages per market server (i.e. when $A = B = 0.5$), the optimum number of regions that should be deployed decreases from 20 to 10, 3, 10 and then back to 20, given call origination probabilities of 0.01, 0.02, 0.04, 0.08 and 0.1, respectively. Thus, a minimum is seen across the optimum set of regions when A is 0.5. A similar pattern is observed when the importance given to average call success rate increases, thereby preserving the minimum. When a higher weighting is given to the average call success rate, the optimum number of regions drops further. The reason for this is that with intermediate call origination probabilities between 0.02 and 0.08, the difference in the average call success rates begins to widen with an increasing number of regions.

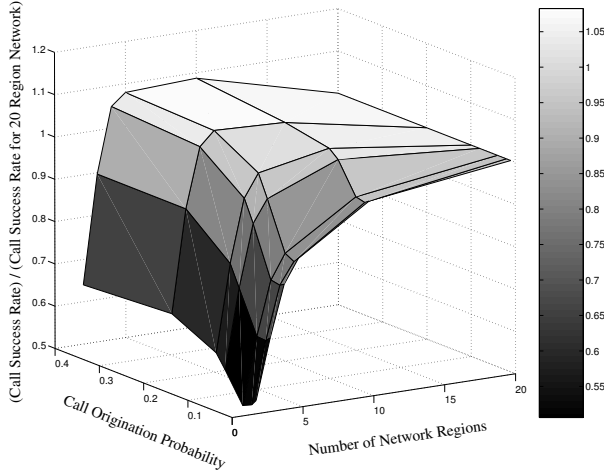


Figure 7. Average call success rate over the 20 region set up average call success rate, as the call origination probability is varied and where there is a single market failure.

This is also shown in figure 4 where the largest change in average call success rate occurs between these values of call origination probability. Thus, it becomes more evident that the fewer regions there are, the higher the average call success rate. In addition to this, because there is a higher weight value for A than there is for B, the optimum number of regions to deploy begins to decrease.

Finally, when exclusive importance is given by the network designer to the average call success rate (i.e. when $A = 1$ and $B = 0$), deploying a single region gives the best overall performance at all call origination probabilities except at 0.07 or below, where the 2 region network set up is best. Figure 6 shows that at these call origination probabilities, there is a maximum when the 2 region network provides the optimum solution. (Section 3.2 provided an explanation for why the 2 region network gave a higher average call success rate at these call origination probabilities).

3.5. Average call success rate with failures

The purpose of our final experiment was to investigate how a single market failure in the network can affect the average call success rate when the number of regions is scaled up. Our hypothesis was that the average call success rate would be higher as the number of regions is scaled up, for any value of call origination probability. Figure 7 shows that this was indeed the case when the number of regions was increased between 2 and 10 regions. (For the sake of clarity in figure 7, all call success rates are divided by the

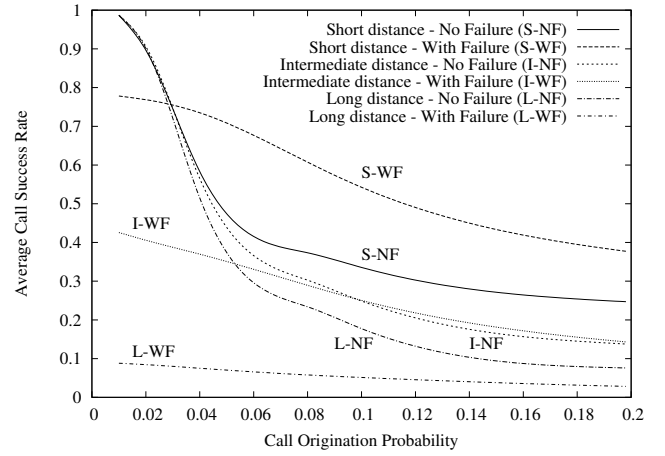


Figure 8. Average call success rate for different distance calls for the 10 region network, with and without a single market failure, as the call origination probability is varied.

20 region call success rate). In the 1 region set up, there are no calls set up since there is no market server functioning. As the number of regions is scaled up between 2 and 10, the average call success rates increase, albeit at a progressively lower rate (i.e. the difference in the average call success rates between the 5 and 10 region set ups is very close when the call origination probability exceeds 0.12). When doubling the number of regions from 10 to 20, given a call origination probability below 0.06, the average call success rate increases. However, when the call origination probability exceeds 0.06, the call success rate decreases from the 10 region set up to the 20 region set up. As a result of this, a maximum can be seen between 5 and 10 regions (as shown in figure 7).

These results can be explained by the following. When the number of regions increases and there is a *single* market server failure, the *maximum* percentage of calls that can take place increases, but at a progressively lower rate. For example, with a 2 region set up, up to 25% of calls can take place when there is a market server failure, a maximum of 40% of calls can take place with 5 regions and 45% with 10 regions. Thus, there is a steady increase in the call success rates between 2 to 10 regions, with the difference between 5 and 10 regions being marginal. This shows that the impact on the performance is affected more, with a market failure, when there are fewer regions in the first instance. However, in the 20 region case, even though a maximum of 47.5% of calls can be made, the 10 region case performs better because contention for resources increases more in the 20 re-

gion set up than in the 10 region case. Thus, given that there is a maximum between the 5 and 10 region set ups, a network designer should ideally deploy a network which has between 5 and 10 regions if a market failure is anticipated.

During our investigation, we also discovered an additional result when looking at the 10 and 20 region set ups, in the absence and presence of failures. For the 10 and 20 region set ups, we found that the overall call success rate is marginally higher with a failure than when there is no failure. This can be explained by means of figure 8, which shows calls of different distances in the network for the 10 region set up, with and without failures. Short distance calls span across 1 and 3 regions, intermediate calls between 4 and 7 regions and long distance calls cover 8 to 10 regions. The percentage of long distance calls for the failure case is less than without failures because the number of these calls is restricted more in the failure case (i.e. when there is a single failure, no calls which span across all 10 regions can be made, for example). Thus, there are more resources available in the network for shorter distance calls to be more successful and indeed, figure 8 shows that the success rate of such calls is greater for the failure case beyond a call origination probability of 0.03. Finally, figure 8 also shows that the number of intermediate distance calls are approximately the same when the call origination probability exceeds 0.08.

4. Related work

There are several market-based architectures that have been proposed for allocating resources in a distributed environment. Gibney and Jennings [5] describe a system in which agents compete for network resources in distributed markets so that calls can be routed in a telecommunications network. The system used a double auction protocol [18] with sealed bids and provided good utilisation of the network where the load was also balanced. However, a drawback of this system was that if some resources on a path were already bought and the next desired resource could not be obtained, then the resources already bought could become redundant and money would be spent unnecessarily. In contrast, our reserve/commit mechanism ensures that this situation is avoided by releasing unused resources immediately and allowing payment to occur only after all necessary resources have been successfully reserved.

The *Global Electronic Market System* (GEM) [13] is a framework for decentralised markets across the Internet. GEM has a single market which is distributed on which goods are sold. In GEM, the markets are replicated and the order for goods is distributed across these markets. Looking at GEM provided an insight into one method of how servers in a market-based resource allocation system could be distributed. However, the approach taken by GEM of replicating the resource information is not suitable for our system

because it would induce more messages in the network than our partitioned approach (as was outlined in section 2.1).

MIDAS [3] is an auction-based mechanism that allocates link bandwidth in a network for making paths. Simultaneous multi-unit Dutch auctions were used as the protocol for allocating resources. This protocol would be inadequate for our requirements since it is not capable of allocating several interrelated goods at the same time.

Finally, Ezhilchelvan and Morgan [4] have looked at how an auction system can be distributed across several servers in a network of servers. However, their approach assumes that communication takes place using a high-bandwidth network which is an assumption that does not hold within our work.

5. Conclusions and future work

In this paper, a system was described that allocates end-to-end bandwidth to set up calls in a network using market-based agents. The system used a combinatorial reverse auction where bundles of interrelated resources were allocated. Scalability testing was performed with respect to increasing the number of regions in a fixed sized network. In conclusion, results show that if light to average network loads are anticipated, then the network designer should consider deploying a decentralised approach with a few number of regions (i.e. 2 or 3 regions) rather than opting for a completely centralised system with a single market. We also see that the average number of messages received per market server is less as the number of regions is scaled up, and thus, allowing the processing of bids and the message load to be distributed better amongst the market servers when there are more of them.

In addition, we found the trade-off between the average call success rate and the number of messages received per market server, with respect to the optimum number of regions in which the network should be partitioned. Results showed that if the network designer assigns a higher priority to receiving fewer messages per market server than the average call success rate in the system, then the optimum number of regions to deploy is higher, at all values of call origination probability. When the level of importance for the average call success rate and the number of messages received per market server are considered to be the same by the designer, the optimum number of regions to deploy decreases progressively, at intermediate call origination probabilities. If a network designer wishes to associate a greater importance to the average call success rate than the number of messages received per market server, the general trend that shows this minimum continues.

Robustness testing was also performed by inducing a single market server failure. This showed that the call success rate was higher as the number of regions increased until a

maximum was reached between 5 and 10 regions, beyond which the call success rate decreased. Therefore, if it is envisaged that a market server failure is likely to occur, then the network designer should consider deploying a network which has between 5 and 10 regions. Finally, during the course of our investigations, we also found that the average call success rate is slightly higher given a single market failure than when there is no market server failure in the network at all. This was explained by the fact that more longer distance calls were restricted in the failure case, thereby allowing many other shorter distance calls to take place with the remaining resources.

There are several investigations that we would like to look into for future work. Firstly, we aim to perform further scalability tests but using networks with various different topologies. Currently, we are investigating the use of other network topologies where it is envisaged that this will provide an insight into how well the system scales, in terms of regions, given these different topologies. Secondly, we plan to look at more robustness tests by introducing multiple market failures in the network to see what additional affect this has on the call success rate. We will also investigate the amount of control capacity used on the nodes in the network, since this will also be in contention. Performing such an investigation would provide a network designer with insight into how much control capacity nodes in the network should with deployed with in the first instance. Thirdly, we aim to impose a restriction on the number of messages that a market server can receive and process within a given amount of time. The purpose of this will be to see how well the system performs, with varying number of markets, when such a limit is imposed. Finally, we would also like to look into issues related to resource pricing in more detail in order to investigate the gains made by buyer and seller agents as a result of calls being set up.

References

- [1] L. ChunLin and L. Layuan. A two level market model for resource allocation optimization in computational grid. In *CF '05: Proceedings of the 2nd conference on Computing frontiers*, pages 66–71, New York, NY, USA, 2005. ACM Press.
- [2] S. H. Clearwater. *Market-Based Control: A Paradigm For Distributed Resource Allocation*. World Scientific Publishing Co. Pte. Ltd, Covent Garden, London, 1996.
- [3] C. Courcoubetis, M. Dramitinos, and G. D. Stamoulis. An auction mechanism for bandwidth allocation over paths: New results. Technical report, London, UK, June 2001.
- [4] P. D. Ezhilchelvan and G. Morgan. A dependable distributed auction system: Architecture and an implementation framework. In *International Symposium on Autonomous Decentralized Systems*, pages 3–10, 2001.
- [5] M. A. Gibney and N. R. Jennings. Dynamic resource allocation by market-based routing in telecommunications networks. In S. Albayrak and F. J. Garijo, editors, *Intelligent Agents for Telecommunication Applications — Proceedings of the Second International Workshop on Intelligent Agents for Telecommunication (IATA'98)*, volume 1437, pages 102–117. Springer-Verlag: Heidelberg, Germany, 1998.
- [6] M. Goemans, L. E. Li, V. S. Mirrokni, and M. Thottan. Market sharing games applied to content distribution in ad-hoc networks. In *MobiHoc '04: Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing*, pages 55–66, New York, NY, USA, 2004. ACM Press.
- [7] N. Haque, N. R. Jennings, and L. Moreau. Resource allocation in communication networks using market-based agents. *International Journal of Knowledge Based Systems*, 18(4-5):163–170, August 2005.
- [8] T. M. Heikkinen. On congestion pricing in a wireless network. *Wireless Networks*, 8(4):347–354, 2002.
- [9] J. Hulaas, H. Stormer, and M. Schnhoff. Anaisoft: An agent-based architecture for distributed market-based workflow management. In *Proceedings of the Software Agents and Workflows for Systems Interoperability workshop of the Sixth International Conference on CSCW in Design*, June 2001.
- [10] P. W. Keller, F. Duguay, and D. Precup. Redagent-2003: An autonomous market-based supply-chain management agent. In *AAMAS '04: Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 1182–1189, Washington, DC, USA, 2004. IEEE Computer Society.
- [11] P. Nicopolitidis, M. S. Obaidat, G. I. Papadimitriou, and A. S. Pomportsis. *Wireless Networks*. John Wiley & Sons Ltd, Chichester, England, 2003.
- [12] L. L. Peterson and B. S. Davie. *Computer Networks: A Systems Approach*. Morgan Kaufmann Publishers Inc, San Francisco, California, 2003.
- [13] B. Rachlevsky-Reich, I. Ben-Shaul, N. T. Chan, A. W. Lo, and T. Poggio. GEM: A global electronic market system. *Information Systems*, 24(6):495–518, 1999.
- [14] O. Ratsimor, T. Finin, A. Joshi, and Y. Yesha. incentive: a framework for intelligent marketing in mobile peer-to-peer environments. In *ICEC '03: Proceedings of the 5th international conference on Electronic commerce*, pages 87–94, New York, NY, USA, 2003. ACM Press.
- [15] T. Sandholm, S. Suri, A. Gilpin, and D. Levine. Winner determination in combinatorial auction generalizations. In *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 69–76, Bologna, Italy, July 2002.
- [16] Y. Z. Wei, L. Moreau, and N. R. Jennings. A market-based approach to recommender systems. *ACM Transactions on Information Systems (TOIS)*, 23(3), 2005.
- [17] M. P. Wellman, J. K. MacKie-Mason, D. M. Reeves, and S. Swaminathan. Exploring bidding strategies for market-based scheduling. In *EC '03: Proceedings of the 4th ACM conference on Electronic commerce*, pages 115–124, New York, NY, USA, 2003. ACM Press.
- [18] P. Wurman, W. Walsh, and M. P. Wellman. Flexible double auctions for electronic commerce: Theory and implementation. *Decision Support Systems*, 24:17–27, 1998.

An Auction-Based Pricing Scheme for Bandwidth Sharing with History-Dependent Utility Functions

Patrick Maillé
GET/ENST-Bretagne
2, rue de la Châtaigneraie CS 17607
35576 Cesson-Sévigné Cedex, France
patrick.maille@enst-bretagne.fr

Bruno Tuffin
IRISA/INRIA
Campus de Beaulieu
35000 Rennes Cedex, France
btuffin@irisa.fr

Abstract

Pricing telecommunication networks has become a topic of high interest in order to deal with the increasing number of subscribers as well as more and more demanding applications. Users' behavior (or preferences) is usually represented by means of the so-called utility function, but in most cases this function expresses the instantaneous level of satisfaction for the quality of service provided. In this paper, we aim at extending a previous work on auctions for bandwidth to the case where users (or applications) are sensitive to the history of their previous allocations. We introduce a mechanism which takes bids in the form of a finite set of three-(or more)-dimensional points, indicating the bidders willingness to pay for a given quantity, contingent on a given history of allocations or prices to the bidder. Bids are partial representations of continuous utility surfaces. The mechanism computes prices and allocations that are approximately (within specified bounds) and myopically incentive compatible and approximately (within specified bounds) and myopically efficient. The results are approximate because we use piecewise linear approximations of the unknown true continuous utility functions. This extension also includes a refinement of the scheme previously published by providing a closer approximation of real user demand functions.

1 Introduction

During the last decade, the Internet has suffered from congestion due to an exponentially increasing number of subscribers, while capacity did not increase in the same way. At the same time, applications have become increasingly bandwidth and quality of service (QoS) demanding. Many solutions have been proposed to cope with congestion, but a natural one is to introduce pricing schemes to

control demand: if prices increase, demand decreases, and conversely. Although many people argue that pricing is not likely to be implemented thanks to the use of optical fiber so that capacity will still be far away from demand, this does not seem to be the case for wireless networks, where available frequencies are limited, as well as for many access networks, where switching from traditional copper lines to optical fiber would be very expensive (this is traditionally called the last mile problem [2]).

Pricing is thus the subject of a large literature; the reader is advised to look at [7, 9, 12, 22] and the references therein for overviews on the range of methods that have been developed so far. We deal here with auction schemes for bandwidth. Auctions have been first used in the *smart market* proposal [14] at the packet level in such a way that packets with the highest bid are served as far as capacity is not exceeded, and the remaining ones are discarded. The per-packet price almost follows the second-price principle: each admitted packets is charged the lowest bid among all admitted packets. In order to alleviate the per-packet management, progressive second price (PSP) auctions have been developed [13]. In that scheme, users submit two-dimensional bids composed of the amount of bandwidth asked, and the unit price for it. Users with the highest unit price are allocated the desired quantity. The scheme is studied using the framework of *game theory* [10]: the game is played until a Nash equilibrium is reached, meaning that no user (or player) has an incentive to deviate from his current allocation. Here again, charges obey the second price principle. Incentive compatibility (users' interest is to declare their real valuation of bandwidth), individual rationality (users will always gain by entering the game) and efficiency (the social welfare of the resulting allocation is maximized) are proved to be verified. Nevertheless, that scheme (still) presents the drawbacks of requiring a convergence phase (meaning a loss of steady state efficiency, especially if players leave the game or new ones enter [15]), and it also

requires that the bid profile (the list of users' bids) be broadcasted to all players at each step, resulting in signalization overhead. Moreover, the properties of the scheme in terms of incentives and efficiency rely on the strong assumption that users are short-sighted, i.e. they do not take into account the convergence phase of the auction game [16]. To cope with those problems, the authors have developed the so-called multi-bid scheme where players submit a set of two-dimensional bids for once when they enter the game [17], meaning that they do not *need* to send new bids afterwards. Allocations and charges are then immediately computed. It is thus a one-shot scheme where players do not need to know the bid profile before submitting. Again, incentive compatibility, individual rationality and efficiency are proved to be verified. Note that other auction schemes exist in the literature, but we do not describe them here for sake of conciseness [1, 3, 5, 6, 19, 20, 21].

In all those pricing schemes, the user behavior is modeled by the so-called utility function, representing users' preferences. This function may depend on various QoS measures, but the obtained throughput or allocated bandwidth is often the measure of interest. In all the previously presented pricing schemes, the utility function depends on the instantaneous allocation, and is therefore independent of the allocations a player obtained previously. However, we can easily imagine applications for which the utility of getting a given amount of bandwidth depends on what was previously obtained: for instance, it is the case of applications targeting a given average bandwidth.

Our goal is thus to extend the work on multi-bid auctions to the case of history-dependent valuation functions. We additionally provide a refinement of the results in [17] by using a better approximation of real valuation functions, resulting in improved error bounds. This work has been inspired in part by ATHENA [8] where a history-dependent utility function has been developed. Nevertheless, ATHENA considers only users with strict requirements, whereas we consider here elastic users, and it does not include the analytical results in terms of incentives or efficiency for instance that are provided here.

This paper is organized as follows. In Section 2 we present the modeling of history-dependent utility functions and discuss its practical validity. Section 3 presents the multi-bid scheme applied to this model. This can be seen as a generalization of the work on multi-bids in [17] where it is applied only to history-independent utility functions, with tighter bounds thanks to a modification of the allocation procedure. Section 4 then describes the properties verified by the scheme, that are individual rationality, (conditional) incentive compatibility and (conditional) efficiency. Note that those inter-temporal results are independent of future allocations. This is typically justified when connection durations are random and unknown, and may end at any

time. The consequences of the three aforementioned properties in terms of user behavior and efficiency/complexity trade-off are investigated in Section 5. Section 7 gives our conclusions and directions for future work.

2 A history-dependent model to represent users' preferences

We describe here the mathematical representation we will consider to take into account the fact that the utility of a user at a given time is a function not only of her current allocation and charge, but also of what happened from her arrival in the network.

Time is divided into slots: allocations and prices cannot change within a time slot, which implies that slots are short enough to prevent a user entering the game from waiting too long before her request be treated, and long enough to allow synchronization among the different parts of the auction game and control computational complexity.

We assume that utility functions are quasi-linear, which means that for a user i whose resource allocation and charge at a given time slot t are respectively $a_{i,t}$ and $c_{i,t}$, the utility $U_{i,t}$ is the difference between what user i thinks the resource $a_{i,t}$ is worth to her (her willingness-to-pay) and the price $c_{i,t}$ she is charged.

In this paper, we assume that a user willingness-to-pay for the resource at a given time slot may depend on her previous allocation and/or charges. Formally, the utility of a user i at time t is

$$U_{i,t} = \theta_i(a_{i,t}, f(X_{i,t-1})) - c_{i,t}, \quad (1)$$

where θ_i is user i 's valuation (or willingness-to-pay) function, $X_{i,t-1}$ is the history of user i 's allocations and costs starting from her arrival until time slot $t - 1$, and f is a function that represents what criteria user i is sensitive to. Following are examples of such criteria.

- $f(X_{i,t-1}) = \emptyset$: this is the case when the user is only sensitive to her current allocation and price. This (simple) model was considered in [13, 17].
- $f(X_{i,t-1}) = a_{i,t-1}$: the bandwidth allocated in the previous slots. This case may correspond to a user that is sensitive to the "continuity" of her allocation. For example, consider a user whose target is to experience a certain average throughput. Then this user will value more the resource at time t if she did not obtain enough at the previous time slot: θ_i in this case is non-increasing in its second argument.
- $f(X_{i,t-1}) = (\sum_{k=t_i^0}^{t-1} a_{i,k}, \sum_{k=t_i^0}^{t-1} c_{i,k})$, where t_i^0 is the time slot when user i entered the game: here the

user is sensitive to the cumulated resource she has obtained and the total charge she has paid until the current slot. This model can apply for example to users downloading a file: $\sum_{k=t_0}^{t-1} a_{i,k}$ is proportional to the amount of data that user i has downloaded from the beginning of her connection, and she may value more or less the resource depending on the amount of data that remains to be transferred, and on the amount of money she has spent.

In this paper, we will treat the case when $f(X_{i,t-1})$ is one-dimensional, because it is more intuitive and enables graphical interpretations. However, all next definitions and properties also hold when $f(X_{i,t-1})$ has several dimensions. Consequently, the utility experienced by a user i during time slot t depends on three parameters, that are her current allocation $a_{i,t}$, the price she is charged $c_{i,t}$, and the value of $f(X_{i,t-1})$ that we call the *history-relevant criterion* for user i . To simplify the notations, we will note ξ_i the value of the history-relevant criterion for a user i at the current time slot, and Ξ_i the set of possible values of ξ_i .

We assume that users have elastic demand, that is to say their valuation function satisfies some regularity assumptions. Moreover, we assume that the dependency of θ_i on the history-relevant criterion is monotone¹. Those properties are summarized in Assumption A.

Assumption A $\forall i \in \mathcal{I}$,

- $\forall \xi_i \in \Xi_i, \theta_i(0, \xi_i) = 0$ and $\theta_i(\cdot, \xi_i)$ is non-decreasing,
- $\forall \xi_i \in \Xi_i, \theta_i(\cdot, \xi_i)$ is concave.
- $\forall q \in \mathbb{R}^+, \theta_i(q, \cdot)$ is a monotone function over Ξ_i .

Such valuation functions are displayed in Figure 1 in the case when $f(X_{i,t-1}) = a_{i,t-1}$. In the following, $\theta'_{i,\xi}$ will denote the partial derivative $\frac{\partial \theta_i(q, \xi)}{\partial q}$ with respect to the first argument q , i.e. the marginal valuation conditionally on the past.

3 Multi-bid auctions to compute allocations at each time slot

In this section, we describe how the definition of the multi-bid auction scheme introduced in [17] can be extended to take into account the time dependency of users' valuation in the history. The mechanism we suggest here implies that at her arrival in the game, each user i submits a certain number M_i of 3-dimensional bids² of the form

$$s_i \triangleq \{(q_i^m, \xi_i^m, \rho_i^m), 1 \leq m \leq M_i\}, \quad (2)$$

¹when $f(X_{i,t-1})$ is a K -dimensional vector, then we need θ_i to be monotone in each of its K components.

²In the general case where $f(X_{i,t-1})$ is a K -dimensional vector, then each s_i^m should be of dimension $K + 2$.

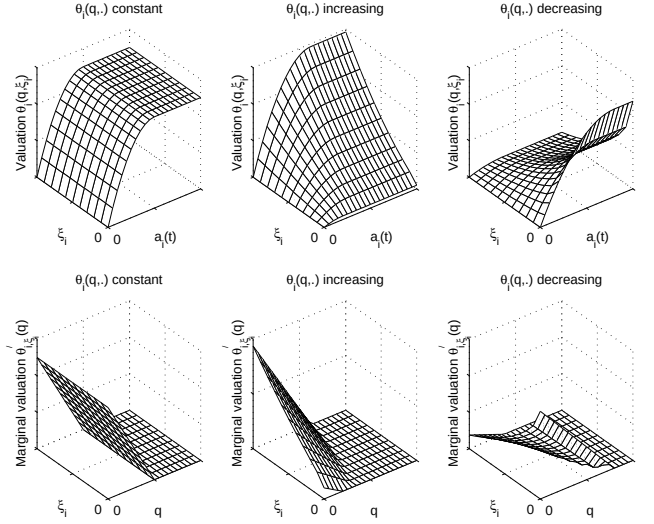


Figure 1. Valuation (top) and marginal valuation (bottom) functions for three types of users: indifferent to the history-relevant criterion (left), increasing (center) and decreasing (right) in that criterion.

that are interpreted as follows: a user submitting such a bid declares she is willing to pay $\rho_i^m \geq 0$ to obtain an allocation $q_i^m > 0$ at the current time slot, if the value of her history-relevant criterion is ξ_i^m . The set $s_i = (s_i^m)_{1 \leq m \leq M_i}$ is called the *multi-bid* submitted by user i . Remark that the scheme we define here is a one-shot scheme, in the sense that a user does not modify her bid once entered the auction game: the multi-bid s_i will be taken into account by the mechanism until the departure of user i whatever the modifications in the network conditions. A user is also asked the sense of variation of her valuation function in the history-relevant criterion; when $f(X_{i,t-1})$ is one-dimensional, one bit added to the multi-bid s_i is sufficient to specify whether $\theta_i(q, \cdot)$ is increasing or decreasing for all q .

Definition 1 We say that user i bids truthfully, or submits a truthful multi-bid, if

- all points $(q_i^m, \xi_i^m, \rho_i^m), 1 \leq m \leq M_i$ are on her valuation function curve, i.e. $\rho_i^m = \theta_i(q_i^m, \xi_i^m)$ for all m ,
- user i reveals her true sense of variation in the history-relevant criterion.

At time slot t , we denote $\mathcal{I}_t$ the set of users that are in the game: this set may change over time, as users may enter and leave the network (like connections starting and ending).

At a given time slot t , the auctioneer will then compute an allocation $a_{i,t}$ and a price to pay $c_{i,t}$ for each user i present in the game, based on the submitted multi-bids, and taking into account the history of the game through the computation of the history-relevant criterion $f(X_{i,t-1})$ of each user.

3.1 Reserve price

We assume that the seller sets a reserve unit price p_0 under which she prefers not to sell the resource. For this to be taken into account by the mechanism, the seller (who will be denoted as player 0) may be seen as a player submitting a bid $s_0 = (q_0, q_0 p_0)$ (with $M_0 = 1$), where $q_0 > Q$. Remark that if we want that bid to have the same form as the multi-bids submitted by the users, then every multi-bid of the form $(q_0, \xi_0, q_0 p_0)$, where the seller declares she is indifferent to the history-relevant criterion, is appropriate. In this paper, we will denote $\mathcal{I}_t^0 \triangleq \mathcal{I}_t \cup \{0\}$ the set of all users present at time slot t (including the seller), and $s_t \triangleq (s_i)_{i \in \mathcal{I}_t}$ the set of all competing multi-bids, that we will call the *multi-bid profile* at time t . Throughout the paper, for $i \in \mathcal{I}_t^0$, we denote $s_{t,-i} \triangleq (s_j)_{j \in \mathcal{I}_t^0 \setminus \{i\}}$ the multi-bid profile without user i .

3.2 Conditional multi-bid depending on the history

When users' valuation functions are independent of the history (which is the case treated in [17]), the multi-bid of each user i is a set of two-dimensional bids from which the functions used to compute allocations and prices are defined. We describe here how the mechanism extracts a set of 2-dimensional points from the multi-bid s_i and a given value ξ_i of the history-relevant criterion $f(X_{i,t-1})$. This set of 2-dimensional points, that we call the *conditional multi-bid* and denote s_{i,ξ_i} , is computed as follows from the 3-dimensional bids:

- if user i declared that $\theta_i(q, \cdot)$ is an **increasing** function for all given q , the conditional multi-bid corresponding to the history-relevant criterion ξ_i is

$$s_{i,\xi_i} \triangleq \{(q_i^m, \rho_i^m) : \xi_i^m \leq \xi_i\}; \quad (3)$$

- if user i declared that $\theta_i(q, \cdot)$ is **decreasing** for all given q , the conditional multi-bid corresponding to the history-relevant criterion ξ_i is

$$s_{i,\xi_i} \triangleq \{(q_i^m, \rho_i^m) : \xi_i^m \geq \xi_i\}. \quad (4)$$

Therefore only bids with history criterion under/above ξ_i , depending on the form of monotonicity, are considered by

just skipping the ξ_i^m values. The conditional multi-bid s_{i,ξ_i} is designed in a way that each pair $(q, \rho) \in s_{i,\xi_i}$ is interpreted as meaning that at the current time slot, user i is willing to pay less than ρ to obtain q units of resource because her history-relevant criterion has value ξ_i . Figure 2 illustrates the conditional multi-bid s_{i,ξ_i} for a truthful bidder whose valuation function $\theta_i(q, \cdot)$ is decreasing for all fixed q .

If player i declared she is indifferent to the history-relevant criterion, then the mechanism simply ignores the ξ_i^m in the bids, and takes $s_{i,\xi_i} \triangleq \{(q_i^m, \rho_i^m) : 1 \leq m \leq M_i\}$ as in [17].

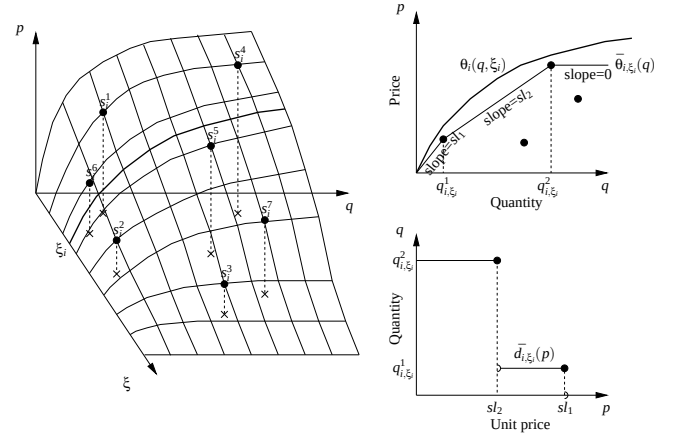


Figure 2. Valuation function and a truthful multi-bid submitted by a user (left), conditional multi-bid s_{i,ξ_i} for a given value ξ_i of the history-relevant criterion. The bids from s_i that appear in s_{i,ξ_i} are s_i^2, s_i^3, s_i^5 and s_i^7 . On the right are the conditional pseudo-valuation function $\bar{\theta}_{i,\xi_i}$ (top-right), and associated conditional pseudo-demand function $\bar{d}_{i,\xi_i}$ (bottom-right) as described in Section 3.3.

3.3 Allocation rule

We now describe how allocations are computed at each time slot t . The mechanism works as follows:

- the value ξ_i of the history-relevant criterion is calculated for each player i , and the conditional multi-bid s_{i,ξ_i} is derived from s_i and ξ_i , as described in subsection 3.2.
- The auctioneer then computes a *conditional pseudo-valuation function* for each player i :

Definition 2 At a given time slot, the conditional pseudo-valuation function for a user i , who submitted

the multi-bid s_i and whose history-relevant criterion equals ξ_i is the function $\bar{\theta}_{i,\xi_i} : \mathbb{R}^+ \rightarrow \mathbb{R}^+$, defined as the **lowest positive and concave function** such that $\bar{\theta}_{i,\xi_i}(q) \geq \rho$ for all pair $(q, \rho) \in s_{i,\xi_i}$.

The conditional pseudo-valuation function is displayed in Figure 2 for a given value of $\xi_i(t)$. The goal is to obtain an approximation of the true valuation function from the multi-bid points. The valuation function θ_i being concave in its first argument, the conditional pseudo-valuation function $\bar{\theta}_{i,\xi_i}$ associated with a truthful multi-bid is such that

$$\forall \xi_i, \forall q \geq 0 \quad \bar{\theta}_{i,\xi_i}(q) \leq \theta_i(q, \xi_i). \quad (5)$$

- c) From the conditional pseudo-valuation function, which is left-differentiable, we compute the *conditional pseudo-marginal valuation function*, denoted by $\bar{\theta}'_{i,\xi_i}$, as the left derivative function of $\bar{\theta}_{i,\xi_i}$. Since $\bar{\theta}'_{i,\xi_i}(q)$ is not defined at $q = 0$, we define $\bar{\theta}'_{i,\xi_i}(0)$ as the right limit of $\bar{\theta}'_{i,\xi_i}$ at 0 (which exists since $\bar{\theta}_{i,\xi_i}$ is piecewise linear), so that $\bar{\theta}'_{i,\xi_i}$ is continuous at 0.
- d) Then a *conditional pseudo-demand function* $\bar{d}_{i,\xi_i}$ is computed for each user:

$$\forall p \in \mathbb{R}^+ \quad \bar{d}_{i,\xi_i}(p) \triangleq \sup\{q : \bar{\theta}'_{i,\xi_i}(q) \geq p\}, \quad (6)$$

with the convention $\sup \emptyset \triangleq 0$. $\bar{d}_{i,\xi_i}$ may also be defined as the largest quantity q of resource that maximizes $\bar{\theta}_{i,\xi_i}(q) - pq$, i.e. the quantity that a user with valuation function $\bar{\theta}_{i,\xi_i}$ would buy to optimize her utility if the resource were sold at a fixed unit price p . Like the conditional pseudo-marginal valuation, the conditional pseudo-demand function $\bar{d}_{i,\xi_i}$ is stair-step, left-continuous, positive and non-increasing.

Those functions are different from the ones initially published in [17] where users were asked to declare their marginal valuation in their multi-bid. This new choice allows to get a closer approximation of the valuation and demand functions, while keeping the properties proved in [17], yielding a reduced gap with respect to the optimal values. Indeed, the new choice computes a concave function for the pseudo-valuation function that better approaches the actual one than the stair-step function in [17].

- e) Based on all conditional pseudo-demand functions of the users present in the game $\bar{\theta}_{i,\xi_i}$, $i \in \mathcal{I}_t$, allocations are determined (using the same rule as in [17]): the aggregated conditional pseudo-demand function $\bar{d}_\xi$ is computed:

$$\bar{d}_\xi \triangleq \sum_{i \in \mathcal{I}_t^0} \bar{d}_{i,\xi_i}. \quad (7)$$

Then the pseudo-market clearing price corresponding to $\bar{d}_\xi$ is defined as

$$\bar{u}_t \triangleq \sup\{p : \bar{d}_\xi(p) > Q\}, \quad (8)$$

and the total capacity Q is shared among flows according to their conditional pseudo-demand functions: the allocation $a_{i,t}(s_t)$ for a user $i \in \mathcal{I}_t^0$ can be written

$$a_{i,t}(s_t) \triangleq \bar{d}_{i,\xi_i}(\bar{u}_t^+) + \frac{\bar{d}_{i,\xi_i}(\bar{u}_t) - \bar{d}_{i,\xi_i}(\bar{u}_t^+)}{\bar{d}_\xi(\bar{u}_t) - \bar{d}_\xi(\bar{u}_t^+)}(Q - \bar{d}_\xi(\bar{u}_t^+)), \quad (9)$$

where, for every function f , $f(x^+)$ denotes the right limit at x (which exists here since conditional pseudo-demand functions are stair-step). The first term of the allocation (9) corresponds to the quantity player i asks at the lowest price $\bar{u}_t^+$ for which supply exceeds pseudo-demand. The second term is strictly positive if all the resource is not allocated at $\bar{u}_t^+$, the surplus $Q - \bar{d}_\xi(\bar{u}_t^+)$ being shared among players who submitted a bid at price $\bar{u}$, with weights proportional to the ‘‘hops’’ of the pseudo-demand functions $\bar{d}_{i,\xi_i}(\bar{u}_t) - \bar{d}_{i,\xi_i}(\bar{u}_t^+)$.

Let us now introduce some remarks that will be helpful in the proofs of properties. Since $\bar{\theta}'_{i,\xi_i}$ is left-continuous, the sup in (6) is a max when $\{q : \bar{\theta}'_{i,\xi_i}(q) \geq p\} \neq \emptyset$, i.e. when $p \leq \bar{\theta}'_{i,\xi_i}(0)$. It implies that

$$p \leq \bar{\theta}'_{i,\xi_i}(0) \Rightarrow \bar{\theta}'_{i,\xi_i}(\bar{d}_{i,\xi_i}(p)) \geq p. \quad (10)$$

Moreover,

$$\forall p \in \mathbb{R}^+ \quad \bar{d}_{i,\xi_i}(p^+) = \sup\{q : \bar{\theta}'_{i,\xi_i}(q) > p\}, \quad (11)$$

(still using $\sup \emptyset = 0$). Therefore

$$\forall p \in \mathbb{R}^+, \quad \bar{\theta}'_{i,\xi_i}(\bar{d}_{i,\xi_i}(p^+)^+) \leq p, \quad (12)$$

since if it were not the case then there would exist a $q > \bar{d}_{i,\xi_i}(p^+)$ such that $\bar{\theta}'_{i,\xi_i}(q) > p$, which would contradict (11). Equation (12) also holds for $p \geq \bar{\theta}'_{i,\xi_i}(0)$.

3.4 Pricing rule

The price $c_{i,t}$ each user $i \in \mathcal{I}_t$ is charged at time slot t is computed according to the multi-bid pricing rule defined in [17], based on the conditional pseudo-valuation functions:

$$c_{i,t}(s_t) \triangleq \sum_{j \neq i} \bar{\theta}_{j,\xi_j}(a_{j,t}(s_t, -i)) - \bar{\theta}_{j,\xi_j}(a_{j,t}(s_t)), \quad (13)$$

where $a_{i,t}(s_t, -j)$ is the allocation that the mechanism would have given to user i at time slot t if player j had just leaved the game, i.e. if the bid profile at time t had been

$s_{t,-i}$. This pricing rule, inspired by the Vickrey-Clarke-Groves pricing rule [4, 11, 23], can be interpreted as an “exclusion-compensation” principle: at time slot t , each user pays for the loss of valuation she imposed on the others by staying in the game.

4 Properties of the scheme

We prove in this section that the auction scheme described above verifies three important properties (at least up to pre-determined constants): (conditional) incentive compatibility which states that users’ best interest is to truthfully declare their valuations, individual rationality which states that entering the game will never yields a negative utility, and (conditional) efficiency, stating that the resulting allocation is the one providing the highest social welfare (that is the sum of valuations of users, auctioneer included).

Before proving those three properties, let us prove a lemma stating that the auction mechanism allocation maximizes the declared “pseudo-social welfare”, i.e. the sum of pseudo-valuations of all users, at each time slot t , given the history of allocations. Define $a_t(s_t) \triangleq (a_{i,t}(s_t))_{i \in \mathcal{I}_t}$ as the vector of allocations at time slot t .

Lemma 1 *The multi-bid allocation $a_t(s_t)$ maximizes the sum of the pseudo-valuations of all users (including the seller) under the capacity constraint:*

$$\forall s_t, \quad a_t(s_t) \in \arg \max_{\tilde{a} \in \mathcal{A}_t} \sum_{i \in \mathcal{I}_t} \bar{\theta}_{i,\xi_i}(\tilde{a}_i) \quad (14)$$

where $\mathcal{A}_t \triangleq \{\tilde{a} = (\tilde{a}_i)_{i \in \mathcal{I}_t^0} : \sum_{i \in \mathcal{I}_t^0} \tilde{a}_i \leq Q\}$.

Proof: We start by showing that for every multi-bid profile s_t , $i \in \mathcal{I}_t$ and $y \in \mathbb{R}^+$, we have

$$\bar{\theta}_{i,\xi_i}(a_{i,t}(s_t)) - \bar{\theta}_{i,\xi_i}(y) \geq \bar{u}_t(a_{i,t}(s_t) - y), \quad (15)$$

where $\bar{u}_t$ is the pseudo-market clearing price computed in (8).

- if $y < a_{i,t}(s_t)$, then

$$\begin{aligned} \bar{\theta}_{i,\xi_i}(a_{i,t}(s_t)) - \bar{\theta}_{i,\xi_i}(y) &\geq \underbrace{\bar{\theta}'_{i,\xi_i}(a_{i,t}(s_t))}_{\geq \bar{\theta}'_{i,\xi_i}(\bar{d}_{i,\xi_i}(\bar{u}_t))} \underbrace{(a_{i,t}(s_t) - y)}_{\geq 0} \\ &\geq \bar{u}_t(a_{i,t}(s_t) - y), \end{aligned}$$

where (10) is applied, holding because $0 \leq y < a_{i,t}(s_t)$ by hypothesis (thus $\bar{d}_{i,\xi_i}(\bar{u}_t) \geq a_{i,t}(s_t) > 0$ and consequently $\bar{u}_t \leq \bar{\theta}'_{i,\xi_i}(0)$).

- If $y \geq a_{i,t}(s_t)$, then

$$\begin{aligned} \bar{\theta}_{i,\xi_i}(y) - \bar{\theta}_{i,\xi_i}(a_{i,t}(s_t)) &= \int_{a_{i,t}(s_t)}^y \bar{\theta}'_{i,\xi_i} \\ &\leq \underbrace{\bar{\theta}'_{i,\xi_i}(a_{i,t}(s_t)^+)}_{\leq \bar{\theta}'_{i,\xi_i}(\bar{d}_{i,\xi_i}(\bar{u}_t)^+)} \underbrace{(y - a_{i,t}(s_t))}_{\geq 0} \\ &\leq \bar{u}_t(y - a_{i,t}(s_t)), \end{aligned}$$

by using (12).

Therefore (15) holds, and Lemma 1 immediately follows: if we denote by $\tilde{a}$ another allocation vector satisfying the capacity constraint, we have

$$\begin{aligned} \sum_{i \in \mathcal{I}_t^0} \bar{\theta}_{i,\xi_i}(a_{i,t}(s_t)) - \sum_{i \in \mathcal{I}_t^0} \bar{\theta}_{i,\xi_i}(\tilde{a}_{i,t}) &\geq \bar{u}_t \left(\underbrace{\sum_{i \in \mathcal{I}_t^0} a_{i,t}(s_t)}_{=Q} - \underbrace{\sum_{i \in \mathcal{I}_t^0} \tilde{a}_{i,t}}_{\leq Q} \right) \\ &\geq 0. \end{aligned}$$

4.1 Incentive compatibility

This subsection aims at studying an important notion called incentive compatibility. It states that a selfish user’s best interest reacting to the auction scheme we have defined in the previous section, in order to optimize her utility, is to play truthfully by declaring her valuation of bandwidth, whatever the bids submitted by the other players be. Indeed, we are going to prove that at each time slot t , in the worst case, a truthful user is ensured that the gap between the utility brought by the multi-bid she submitted and the maximum utility that she could have obtained by bidding differently is less than $\max_{q \in [0, Q]} \theta_i(q, \xi_i) - \bar{\theta}_{i,\xi_i}(q)$, where $\xi_i = f(X_{i,t-1})$.

Proposition 2 (incentive compatibility) $\forall t, \forall i \in \mathcal{I}_t, \forall s_{t,-i}, \forall s_i, \forall \xi_i, \forall \xi_i = f(X_{i,t-1})$,

s_i truthful $\Rightarrow$

$$U_{i,t}((s_i, s_{t,-i}), \xi_i) \geq U_{i,t}((\tilde{s}_i, s_{t,-i}), \xi_i) - C_{i,\xi_i}, \quad (16)$$

with

$$C_{i,\xi_i} \triangleq \max_{q \in [0, Q]} \theta_{i,t}(q, \xi_i) - \bar{\theta}_{i,\xi_i}(q). \quad (17)$$

Proof: Let $s_t = (s_i, s_{t,-i})$. Note that the conditional pseudo-valuation functions $\theta_{j,\xi_j}, j \neq i$, are the same if the multi-bid profile is s_t or $(\tilde{s}_i, s_{t,-i})$ and that, from the definition of C_{i,ξ_i} , $\bar{\theta}_{i,\xi_i}(q) \leq \theta_{i,t}(q, \xi_i) \leq \bar{\theta}_{i,\xi_i}(q) + C_{i,\xi_i}, \forall q \in [0, Q]$. We therefore have, using (13):

$$\begin{aligned}
& U_{i,t}(s_t, \xi_i) - U_{i,t}(\tilde{s}_i, s_{t,-i}, \xi_i) \\
& \geq \bar{\theta}_{i,\xi_i}(a_{i,t}(s_t)) + \sum_{j \neq i} \bar{\theta}_{j,\xi_j}(a_{j,t}(s_t)) - C_{i,\xi_i} \\
& \quad - \bar{\theta}_{i,\xi_i}(a_{i,t}(\tilde{s}_i, s_{t,-i})) - \sum_{j \neq i} \bar{\theta}_{j,\xi_j}(a_{j,t}(\tilde{s}_i, s_{t,-i})) \\
& \geq -C_{i,\xi_i} + \underbrace{\sum_{j \in \mathcal{I}_t^0} (\bar{\theta}_{j,\xi_j}(a_{j,t}(s_t)) - \bar{\theta}_{j,\xi_j}(a_{j,t}(\tilde{s}_i, s_{t,-i})))}_{\geq 0 \text{ (Lemma 1)}}
\end{aligned}$$

which establishes Proposition 2.

Bidding truthfully therefore ensures that, at each time slot t , player i maximizes her utility up to a constant that is the largest difference between her conditional valuation and pseudo-valuation functions. Notice that incentive compatibility is proved only when reasoning per time slot, conditionally on the past; there may exist untruthful strategies that lead in some situations to a larger overall utility during the presence period of a player, given the fact that your current allocation affects your future utility. However, playing truthfully may anyway be the best strategy for users who do not have an idea of their (random) connection duration, so that it could stop at any time, or of the evolution of network conditions in the future (with other users randomly entering/leaving the game).

4.2 Individual rationality

A second interesting property is individual rationality, stating that a user who submitted a truthful multi-bid is ensured to obtain a non-negative utility, i.e. to be charged a price equal to or below her valuation for her allocation. This property implies that a rational user will always enter the game and continue to play at each time slot.

Proposition 3 $\forall t, \forall i \in \mathcal{I}_t, \forall s_{t,-i}, \forall \xi_i = f(X_{i,t-1})$, if user i submitted a truthful multi-bid s_i then

$$U_{i,t}(s_i, s_{t,-i}) = \theta_i(a_{i,t}(s_t), \xi_i) - c_{i,t}(s_t) \geq 0. \quad (18)$$

Proof: Equation (5) implies

$$\begin{aligned}
U_{i,t}(s_t, \xi_i) &= \theta_{i,t}(a_{i,t}(s_t)) - c_{i,t}(s_t) \\
&= \theta_{i,t}(a_{i,t}(s_t)) + \sum_{j \neq i} [\bar{\theta}_{j,\xi_j}(a_{j,t}(s_t)) - \bar{\theta}_{j,\xi_j}(a_{j,t}(s_{t,-i}))] \\
&\geq \sum_{j \in \mathcal{I}_t} \bar{\theta}_{j,\xi_j}(a_{j,t}(s_t)) - \sum_{j \neq i} \bar{\theta}_{j,\xi_j}(a_{j,t}(s_{t,-i})) \\
&\geq \sum_{j \in \mathcal{I}_t} \bar{\theta}_{j,\xi_j}(a_{j,t}(s_t)) - \bar{\theta}_{j,\xi_j}(a_{j,t}(s_{t,-i})) \geq 0,
\end{aligned}$$

where the last line comes from Lemma 1.

4.3 Efficiency

The last important property we wish to show is (conditional) efficiency. We prove that, at each time slot t , conditionally to the past and independently of the future, the auction scheme allocates efficiently the available resource among users. The efficiency measure that we consider here is *social welfare*, that is the total valuation of users (including the seller) for the allocation: $\sum_{i \in \mathcal{I}_t^0} \theta_i(a_{i,t}, \xi_i)$. This quantity is also the sum of utilities of all users, if we consider that the utility of the seller is her valuation for her allocation plus her total revenue, $U_{0,t} = \theta_0(a_{0,t}) + \sum_{i \in \mathcal{I}_t} c_{i,t}$ for each t , where $\theta_0(q) = p_0 q$.

Like for the incentive compatibility property, efficiency is instantaneous (i.e. at each time slot t), which means that allocations are not necessarily efficient if we consider periods of several time slots (then in general a discount factor has to be introduced to compare valuations at different times). However, once again, such a property indicates that the mechanism behaves in a good fashion, and may be chosen if the network conditions of the future (number of users and submitted multi-bids) cannot be predicted.

To establish this property, we add some regularity assumptions on valuation functions:

Assumption B $\exists \kappa > 0 : \forall i \in \mathcal{I}, \forall \xi_i \in \Xi_i$,

- $\theta_i(\cdot, \xi_i)$ is differentiable in its first argument and θ'_{i,ξ_i} (the derivative in the first argument) is continuous,
- $\forall z, z', z > z' \geq 0, \theta'_{i,\xi_i}(z) - \theta'_{i,\xi_i}(z') > -\kappa(z - z')$,
- $\theta'_{i,\xi_i}(Q) = 0$.

The first and second points of Assumption B were introduced by Lazar and Semret in [13] for valuation functions independent of the history to prove the efficiency of Progressive Second Price auctions. The last point states that the available resource of the link is sufficient to fully satisfy a user if she is the only one using that link.

For such valuation functions, we have the following efficiency result:

Proposition 4 Under Assumptions A and B, the auction allocation at each time slot is close to the social welfare optimum, conditionally on the past, when players bid truthfully: $\forall \mathcal{I}_t, \forall (\xi_i)_{i \in \mathcal{I}_t}$ and truthful multi-bid profile s_t ,

$$\sum_{i \in \mathcal{I}_t^0} \theta_i(a_{i,t}(s_t), \xi_i) \geq \sup_{\tilde{a} \in \mathcal{A}_t} \left(\sum_{i \in \mathcal{I}_t^0} \theta_i(\tilde{a}_i, \xi_i) \right) - Q \sqrt{8\kappa \max_{i \in \mathcal{I}_t} C_{i,\xi_i}}$$

with $\mathcal{A} \triangleq \{\tilde{a} = (\tilde{a}_i)_{i \in \mathcal{I}_t^0} \in [0, Q]^{|\mathcal{I}_t^0|} : \sum_{i \in \mathcal{I}_t^0} \tilde{a}_i \leq Q\}$.

Proof: Relations (5) and (17) lead to

$$\forall q, \tilde{q} \in [0, Q], \forall \xi_i, \\ |\theta_i(q, \xi_i) - \bar{\theta}_{i,\xi_i}(q) - (\theta_i(\tilde{q}, \xi_i) - \bar{\theta}_{i,\xi_i}(\tilde{q}))| \leq C_{i,\xi_i},$$

or equivalently

$$\left| \int_{\tilde{q}}^q [\theta'_{i,\xi_i}(x) - \bar{\theta}'_{i,\xi_i}(x)] dx \right| \leq C_{i,\xi_i}. \quad (19)$$

Assume now that there exist $K > 0$, ξ_i and $q \geq 0$ such that

$$\theta'_{i,\xi_i}(q) > \bar{\theta}'_{i,\xi_i}(q) + K. \quad (20)$$

Then from Assumption B, $\theta'_{i,\xi_i}(q + \frac{K}{\kappa}) \geq \theta'_{i,\xi_i}(q) - \kappa \frac{K}{\kappa} > 0$, so that $q + \frac{K}{\kappa} \in [0, Q]$. Therefore (19) gives

$$\begin{aligned} C_{i,\xi_i} &\geq \int_q^{q+\frac{K}{\kappa}} (\theta'_{i,\xi_i}(x) - \bar{\theta}'_{i,\xi_i}(x)) dx \\ &\geq \int_q^{q+\frac{K}{\kappa}} (\theta'_{i,\xi_i}(q) + \kappa(q-x) - \theta'_{i,\xi_i}(q) + K) dx = \frac{K^2}{2\kappa}, \end{aligned}$$

thus from (20) we have

$$\forall \xi_i, \forall q \in [0, Q], \quad \theta'_{i,\xi_i}(q) - \bar{\theta}'_{i,\xi_i}(q) \leq \sqrt{2\kappa C_{i,\xi_i}}. \quad (21)$$

On the other hand, consider $K > 0$, ξ_i and $q > 0$ such that $\theta'_{i,\xi_i}(q) < \bar{\theta}'_{i,\xi_i}(q) - K$. Then from Assumption B, $q - \frac{K}{\kappa} \geq 0$, unless we would have $\theta'_{i,\xi_i}(0) < \bar{\theta}'_{i,\xi_i}(q)$, which would contradict (5). Applying again (19), we get

$$\begin{aligned} -C_{i,\xi_i} &\leq \int_{q-\frac{K}{\kappa}}^q (\theta'_{i,\xi_i}(x) - \bar{\theta}'_{i,\xi_i}(x)) dx \\ &\leq \int_{q-\frac{K}{\kappa}}^q (\theta'_{i,\xi_i}(q) + \kappa(q-x) - \theta'_{i,\xi_i}(q) - K) dx = \frac{-K^2}{2\kappa}, \end{aligned}$$

therefore

$$\forall \xi_i, \forall q \in [0, Q], \quad \theta'_{i,\xi_i}(q) - \bar{\theta}'_{i,\xi_i}(q) \geq -\sqrt{2\kappa C_{i,\xi_i}}. \quad (22)$$

We now establish Proposition 4: from (21), (22) and the immediate relation $\bar{d}_{i,\xi_i}(\bar{u}_t^+) \leq a_{i,t}(s_t) \leq \bar{d}_{i,\xi_i}(\bar{u}_t)$ (that is a consequence of (9)), we easily get, applying (10) and (12),

$$\begin{cases} \theta'_{i,\xi_i}(a_{i,t}(s_t)) \leq \bar{u}_t + \sqrt{2\kappa C_{i,\xi_i}} \\ a_{i,t}(s_t) > 0 \Rightarrow \theta'_{i,\xi_i}(a_{i,t}(s_t)) \geq \bar{u}_t - \sqrt{2\kappa C_{i,\xi_i}}. \end{cases}$$

Let $\mathcal{I}_{t,+} = \{i \in \mathcal{I}_t^0 : \bar{a}_i \geq a_{i,t}(s_t)\}$ and $\mathcal{I}_{t,-} = \{i \in \mathcal{I}_t^0 :$

$\bar{a}_i < a_{i,t}(s_t)\}$. We have

$$\begin{aligned} &\sum_{i \in \mathcal{I}_t^0} \theta_i(a_{i,t}, \xi_i) - \theta_i(\bar{a}_i, \xi_i) \\ &\geq \sum_{i \in \mathcal{I}_{t,-}} \theta'_{i,\xi_i}(a_{i,t}(s_t))(a_{i,t}(s_t) - \bar{a}_i) - \\ &\quad - \sum_{i \in \mathcal{I}_{t,+}} \theta'_{i,\xi_i}(a_{i,t}(s_t))(\bar{a}_i - a_{i,t}(s_t)) \\ &\geq \sum_{i \in \mathcal{I}_{t,-}} (\bar{u}_t - \sqrt{2\kappa C_{i,\xi_i}})(a_{i,t}(s_t) - \bar{a}_i) - \\ &\quad - \sum_{i \in \mathcal{I}_{t,+}} (\bar{u}_t + \sqrt{2\kappa C_{i,\xi_i}})(\bar{a}_i - a_{i,t}(s_t)) \\ &\geq -\sqrt{2\kappa} \sum_{i \in \mathcal{I}_t} \sqrt{C_{i,\xi_i}} |a_{i,t}(s_t) - \bar{a}_i| \\ &\geq -2Q \sqrt{2\kappa \max_i C_{i,\xi_i}}, \end{aligned}$$

which gives the proposition.

5 Player and auctioneer behavior

5.1 Multi-bid choice for a user

As pointed out in the previous section, bidding truthfully at the arrival into the game ensures a user i to obtain at each time slot t a utility that is close to the maximum possible, up to $C_{i,f(X_{i,t-1})}$. If user i has an a priori probability distribution of the allocations she will obtain during her connection period, then she may try to choose her multi-bid so as to minimize the expected value of $C_{i,f(X_{i,t-1})}$. If she does not have such an a priori distribution, then a natural way to choose her bid s_i is to try to be as close as possible to the maximum utility at each time slot *in the worst case*, that is

$$s_i \in \arg \min \{ \max_{\xi_i \in \Xi_i} C_{i,\xi_i} : s_i = (s_i^m)_{1 \leq m \leq M} \text{ truthful} \}. \quad (23)$$

For general valuation functions θ_i , finding an analytical expression of a multi-bid satisfying (23) is beyond the scope of this paper, but could otherwise be obtained by numerical optimization procedures (meta-heuristics for instance).

5.2 Trade-off between economical efficiency and computational complexity

Proposition 4 states that the allocation at a given time slot t is close to the socially optimal one, up to a value that decreases (if users are rational) when the number of bids M_i allowed in a multi-bid increases. Moreover, we proved in [17] that the complexity of computing allocations and prices given the conditional multi-bids $(s_{i,\xi_i})_{i \in \mathcal{I}_t}$ are of the order $O(|\mathcal{I}_t| \times \sum_{i \in \mathcal{I}_t} M_{i,\xi_i})$, where M_{i,ξ_i} is the number

of two-dimensional bids in s_{i,ξ_i} . Notice that at each time slot, selecting the conditional multi-bids s_{i,ξ_i} for all players in $\mathcal{I}_t$ needs around $\sum_{i \in \mathcal{I}_t} M_i$ operations.

Therefore, increasing M_i for all users in $\mathcal{I}_t$ ensures that the conditional social welfare be close to the optimal one, but also yields a higher complexity. Depending on the computational capacities of the auctioneer and the relative importance of efficiency *versus* complexity, the auctioneer may admit at most M bids of the form (q_i^m, ξ_i^m, p_i^m) in a multi-bid, and choose M to balance the trade-off. A study of this trade-off, as well as a comparison with the PSP scheme in terms of both complexity and efficiency, has been realized in [18] in the case of atemporal utility functions, illustrating the advantages of the multi-bids scheme.

6 Numerical illustration

In this section, we present an example of scenario with different types of users entering and leaving the game over time. The valuation functions that are used are those described in Figure 1. The history-relevant criterion considered here for each user is his allocation at the previous slot. We illustrate here the behavior of our scheme for when the following scenario is applied (the corresponding allocations and pseudo-market clearing price are plotted in Figure 3 for $Q = 10$, $M = 200$ and truthful bids taken with equally spaced values of q_i^m and ξ_i^m):

- the game begins at slot $t = 0$ with only one player, of type 2 (valuation increasing in the history-relevant criterion), who leaves the game at time $t = 28$,
- another type-2 player bids for bandwidth between $t = 7$ and $t = 16$,
- a player of type 3 (valuation decreasing in the history-relevant criterion) enters the game at $t = 23$, and another one arrives at $t = 35$.

Between time slots $t = 7$ and $t = 16$, two identical players of type 2 are competing for the resource. However, player 1 was already present upon arrival of player 2, and had obtained some resource at time $t = 6$. Therefore due to his type, he values the resource more than player 2 at time $t = 7$, which explains that he obtains more resource at that time. However he obtains less than in the previous slot, hence his conditional valuation for slot $t = 8$ will decrease, whereas for the same reason the conditional valuation of player 2 increases. For the following slots, the allocations of both players therefore get closer.

The remaining type-2 player then faces a type-3 player in the interval $[23, 28]$: when that player enters the game, he obtains a large amount of resource (due to an important valuation), which will decrease his conditional valuation for

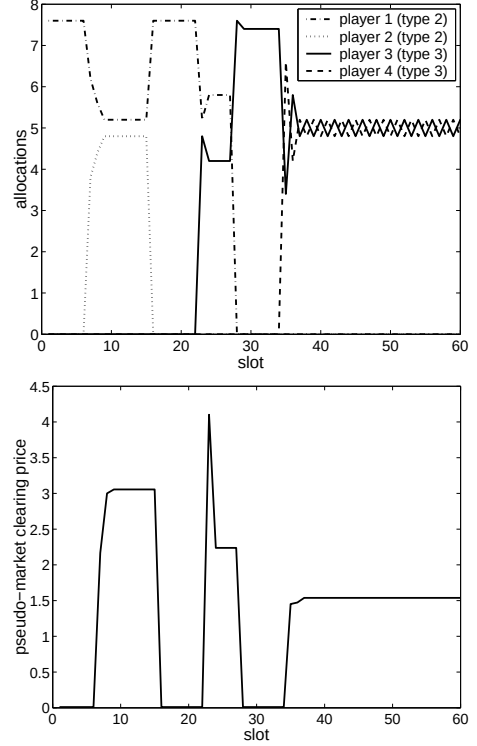


Figure 3. Allocations $a_{i,t}$ (top) and pseudo-market clearing price $\bar{u}_t$ (illustrating the congestion level) (bottom) for the scenario described in Section 6.

the next slot. That is the reason why the corresponding allocation of player 3 is lower at slot 24 than at slot 23. The symmetric reasoning applies to the type-2 player.

After slot 35, we have a situation where two identical players of type 3 compete for bandwidth. Since player 3 had obtained some resource at slot 34, his conditional valuation is lower than that of player 4, consequently at slot 35 player 4 obtains more resource than player 3. At each time slot, the player who obtains more resource is the one who obtained less at the previous slot, which explains the oscillations that occur in Figure 3.

7 Conclusions

In this paper, we have designed an auction scheme to allocate the available bandwidth of a communication link among several users *over time*. Based on the previously defined multi-bid auction mechanism, we adapted the scheme to the case when the valuation of users for the resource depends on their history since their arrival in the game, including also tighter bounds due to a better approximation of the true valuation functions. The signaling overhead is low, since users submit a number M of 3-dimensional bid once only for all their connection duration, and do not need to receive any information, unlike in some other schemes. Moreover, the computational complexity can be controlled, and the scheme has been proved to satisfy some properties in terms of incentives and social welfare maximization.

Some more work can be done to take into account history-dependent valuations: in this paper we assumed the dependency to be monotone in the history-relevant criterion, in order to define properly the conditional multi-bid and the conditional valuation function. Adapting the scheme described here to the case when the dependency is not monotone would deserve some attention.

References

- [1] C. Ambühl, A. E. F. Clementi, P. Penna, G. Rossi, and R. Silvestri. Energy consumption in radio networks: Selfish agents and rewarding mechanisms. In *Proc. of WAOA 2003, LNCS 2909*, pages 248–251, 2003.
- [2] L. Bernstein. Managing the last mile. *IEEE Communications Magazine*, 35(10):72–76, Oct 1997.
- [3] M. Bitsaki, G. Stamoulis, and C. Courcoubetis. An efficient auction mechanism for hierarchically structured bandwidth markets. In *Proc. of ICQT'04, LNCS 3266*, Barcelona, Spain, Oct 2004. Springer.
- [4] E. H. Clarke. Multipart pricing of public goods. *Public Choice*, 11:17–33, 1971.
- [5] C. Courcoubetis, M. P. Dramitinos, and G. D. Stamoulis. An auction mechanism for bandwidth allocation over paths. In *Proc. of the 17th International Teletraffic Congress (ITC)*, Dec 2001.
- [6] J. Crowcroft, R. Gibbens, F. Kelly, and S. Östring. Modelling incentives for collaboration in mobile ad hoc networks. In *Proc. of WiOpt'03*, Mar 2003.
- [7] L. A. DaSilva. Pricing for QoS-enabled networks: A survey. *IEEE Communications Surveys*, 3(2):2–8, 2000.
- [8] M. P. Dramitinos, C. Courcoubetis, and G. D. Stamoulis. Auction-based resource reservation in 2.5/3G networks. In *Proc. of WiOpt'03: Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, INRIA, Sophia-Antipolis, France, Mar 2003.
- [9] M. Falkner, M. Devetsikiotis, and I. Lambadaris. An overview of pricing concepts for broadband IP networks. *IEEE Communications Surveys & Tutorials*, 3(2), 2000.
- [10] D. Fudenberg and J. Tirole. *Game Theory*. MIT Press, Cambridge, Massachusetts, 1991.
- [11] T. Groves. Incentives in teams. *Econometrica*, 41(3):617–631, Jul 1973.
- [12] F. P. Kelly, A. K. Maulloo, and D. K. H. Tan. Rate control in communication networks: Shadow prices, proportional fairness and stability. *Journal of the Operational Research Society*, 49:237–252, 1998.
- [13] A. A. Lazar and N. Semret. Design and analysis of the progressive second price auction for network bandwidth sharing. *Telecommunication Systems - Special issue on Network Economics*, 1999.
- [14] J. K. MacKie-Mason and H. R. Varian. Pricing the internet. In B. Kahin and J. Keller, editors, *Public Access to the Internet*, pages 269–314. MIT Press, 1995.
- [15] P. Maillé and B. Tuffin. The progressive second price mechanism in a stochastic environment. *Netnomics*, 5(2):119–147, Nov 2003.
- [16] P. Maillé and B. Tuffin. A progressive second price mechanism with a sanction bid from excluded players. In *Proc. of ICQT'03, LNCS 2816*, pages 332–341, Munich, Germany, Sept 2003. Springer.
- [17] P. Maillé and B. Tuffin. Multi-bid auctions for bandwidth allocation in communication networks. In *Proc. of IEEE INFOCOM*, Mar 2004.
- [18] P. Maillé and B. Tuffin. Multi-bid versus progressive second price auctions in a stochastic environment. In *Proc. of ICQT'04, LNCS 3266*, pages 318–327, Barcelona, Spain, Oct 2004. Springer.
- [19] P. Marbach. Pricing differentiated services networks: Bursty traffic. In *Proc. of IEEE INFOCOM*, 2001.
- [20] P. Reichl, G. Fankhauser, and B. Stiller. Auction models for multiprovider internet connections. In *Proc. of Messung, Modellierung und Bewertung (MMB'99)*, Trier, Germany, Sept 1999.
- [21] J. Shu and P. Varaiya. Pricing network services. In *Proc. of IEEE INFOCOM*, 2003.
- [22] B. Tuffin. Charging the internet without bandwidth reservation: an overview and bibliography of mathematical approaches. *Journal of Information Science and Engineering*, 19(5):765–786, Sept 2003.
- [23] W. Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *Journal of Finance*, 16(1):8–37, Mar 1961.