Burkhard Stiller
Peter Reichl
Bruno Tuffin (Eds.)

LNCS 4033

Performability Has its Price

5th International Workshop on Internet Charging and QoS Technologies, ICQT 2006
St. Malo, France, June 2006, Proceedings

Lecture Notes in Computer Science 4033

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Burkhard Stiller Peter Reichl
Bruno Tuffin (Eds.)

Performability Has its Price

5th International Workshop on Internet Charging and QoS Technologies, ICQT 2006
St. Malo, France, June 27, 2006
Proceedings

Springer

Volume Editors

Burkhard Stiller
University of Zürich
Department of Informatics
Binzmühlstrasse 14, 8050 Zürich, Switzerland
and
Swiss Federal Institute of Technology, ETH, Zürich
Computer Engineering and Networks Laboratory, TIK
Gloriastrasse 35, 8092 Zürich, Switzerland
E-mail: stiller@tik.ee.ethz.ch

Peter Reichl
Forschungszentrum Wien, FTW
Donau-City-Str. 1, 1220 Wien, Austria
E-mail: reichl@ftw.at

Bruno Tuffin
IRISA-INRIA
Campus universitaire de Beaulieu, 35042 Rennes Cedex, France
E-mail: Bruno.Tuffin@irisa.fr

Library of Congress Control Number: 2006927430

CR Subject Classification (1998): C.2, H.4, H.3, J.1, K.4.4

LNCS Sublibrary: SL 5 – Computer Communication Networks and Telecommunications

ISSN 0302-9743
ISBN-10 3-540-35456-5 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-35456-7 Springer Berlin Heidelberg New York

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 11780502 06/3142 5 4 3 2 1 0

Preface

This volume of the Lecture Notes in Computer Science series publishes the set of papers accepted for the ICQT 2006 workshop, i.e., the 5th International Workshop on Internet Charging and QoS Technology (ICQT), which was collocated with ACM SIGMETRICS. These events took place in St. Malo, France and were hosted by the IUT (Institut Universitaire Technologique) of St. Malo.

QoS-guaranteed services enable a huge variety of prosperous business models, and the need for viable models and pricing schemes is urgent. The resulting combination of technical and economic perspectives drives many relevant research topics for application developers, business architects, network providers, service providers, and customers. Especially the identification of novel service charging solutions, the investigation and evaluation of their technical feasibility, and the consolidation of technical and economic mechanisms for enabling a fast, guaranteed, and efficient charging of services is of fundamental importance for the future evolution of the Internet, and as such the central focus of the international ICQT workshop series.

This year's ICQT constituted the 5th vivid workshop on Internet economics and charging technology, which initially took place in 2001 in Vienna, Austria within the framework of the Annual Meeting of the German Society for Computer Science (GI) and the Austrian Computer Society, and which was collocated in 2002 with the QofIS 2002 workshop in Zürich, Switzerland, in 2003 with the NGC 2003 workshop in Munich, Germany, and in 2004 again with QofIS 2004 in Barcelona, Spain.

Under the specific motto of this collocated workshop "Performability Has Its Price", ICQT brings together researchers from the areas of technology and economics in both industry and academia to discuss key advancements and to support further progress in these fields. The combination of economic models, auctions, peer-topeer, and secure charging addresses a highly interesting facet of networking research and business modelling. While charging for Internet services inter-relates existing networking techniques with economic models, business models drive the need of networking technology to deal with external factors and enrich viable technology with mandatory functionality. Thus, ICQT targets at the identification of those two area overlaps and the range of session topics exactly reflects this situation.

In general, ICQT 2006 provided, as did all of its predecessors, a single-track and one-day program, in order to stimulate interaction and active participation. In summary, the technical sessions of the ICQT workshop contained eight full papers, which were selected after a thorough reviewing process out of a total number of 27 submissions. For the fifth time showing a truly international scope, the final program included four European and four Asian-Pacific full papers – counted on the first author's affiliation – as well as the keynote from Europe.

ICQT's technical and research success is due to the technical program committee, whose members devoted their excellent knowledge as well as many hours of their time to provide the basis of a highly qualified technical program. Furthermore, we would like to express our thanks to the ICQT 2006 web master and the submission system handler, who performed an excellent job. The Fondation Métivier is kindly acknowledged for sponsoring the best student paper award.

Thanks go also to the ACM Sigmetrics/IFIP Performance organizers for allowing us to collocate ICQT with their renowned event. Special thanks go to the local organization handled in an exceptional way by Edith Blin, Elisabeth Lebret, Louis-Marie Le Ny, and Raymond Marie.

Finally, we would like to address our thanks to the Springer Editorial team, for their smooth cooperation on finalizing these proceedings. Last but not least, thanks go to IRISA for hosting the ICQT 2006 workshop in this fascinating environment.

April 2006

Burkhard Stiller
Peter Reichl
Bruno Tuffin

Courtesy of: Ville de Saint-Malo — Service CommunicationPicture: Manuel Clauzier

Organization

General Chair

Bruno Tuffin	*IRISA, France*

Program Co-chairs ICQT 2006

Burkhard Stiller	*University of Zürich and ETH Zürich, Switzerland*
Peter Reichl	*FTW Vienna, Austria*

Program Committee ICQT 2006

Rui Aguiar	*Portugal Telecom and IT Aveiro, Portugal*
Eitan Altmann	*INRIA Sophia-Antipolis, France*
Jörn Altmann	*International University Bruchsal, Germany*
Ragnar Andreassen	*Telenor, Norway*
Nicole Blefari-Melazzi	*University of Rome, Italy*
Torsten Braun	*University of Bern, Switzerland*
Dominique Barth	*University of Versailles, France*
Costas Courcoubetis	*Athens University of Economics and Business, Greece*
Chris Edwards	*Lancaster University, UK*
Errin Fulp	*North Carolina State University, USA*
Richard Gibbens	*University of Cambridge, UK*
Martin Karsten	*University of Waterloo, Canada*
Peter Key	*Microsoft Research Cambridge, UK*
Patrick Maillé	*ENST Rennes, France*
Peter Marbach	*University of Toronto, Canada*
Robin Mason	*University of Southampton, UK*
Lee McKnight	*Syracuse University, New York, USA*
Andrew Odlyzko	*University of Minnesota, USA*
Huw Oliver	*Ericsson, Ireland*
Kihong Park	*Purdue University, USA*
Guido Petit	*Alcatel, Belgium*
David Reeves	*North Carolina State University, USA*
Günter Schäfer	*University of Ilmenau, Germany*
Vasilios Siris	*ICS Forth, Greece*
David Songhurst	*BT Exact Technologies, UK*
George Stamoulis	*Athens University of Economics and Business, Greece*
Richard Steinberg	*University of Cambridge, UK*

Local Organization

Edith Blin	*IRISA, France*
Elisabeth Lebret	*IRISA, France*
Louis-Marie Le Ny	*IRISA, France*
Raymond Marie	*IRISA, France*

Reviewers

A set of detailed reviews for papers submitted to ICQT 2006 was performed by all of our reviewers, which correspond to the full Program Committee members as stated above. Therefore, it is of great pleasure to the Program Co-chairs to thank all those reviewers for their important work.

Table of Contents

Cooperation and QoS in Fast Packet Networks: The View from the Edge

Peter Key

Microsoft Research, Cambridge, U.K
Peter.Key@microsoft.com

1 ICQT'06 Keynote — Extended Abstract

The fundamental architecture of the current generation Internet has changed little over the past two decades. This is a problem when services need to be introduced that require something more than basic point-to-point connectivity, such as needing some form of QOS or multicast capability. Multiple ownership of the Internet is both a strength and a weakness — what incentives do ISPs have to evolve their architec-tures? At the same time, processing power and intelligence is increasing at the edge of the network, which can be harnessed to create new services. For example, adaptive network-aware applications can react to changing network conditions, while P2P overlay networks can bypass many of the underlay's inherent restrictions.

We give an example of a form of differential QoS using edge-based or end-sys-tem control, showing how it possible to construct a "lower than best effort" service, suitable for background transfers. We then discuss how to generalise this to give cer-tain minimal guarantees.

For multicast we describe a P2P filecasting solution that uses network coding and a simple form of cooperation. It is possible to view this scheme as a form of multi-path routing. In general, giving the edge-systems some degree of control over routing has potential performance benefits. For unicast applications, by combining conges-tion control with a flexible routing scheme, it is possible to halve response times and double the load the network can carry compared with existing approaches. If imple-mented at the WAN level, this may also change the incentive structure for ISPs.

Finally we discuss issues of incentives and cooperation, and comment on how far it is possible to progress within the current pricing framework and with edge-based solutions *without* requiring fundamental changes to the core of the network.

B. Stiller, P. Reichl, and B. Tuffin (Eds.): ICQT 2006, LNCS 4033, p. 1, 2006.

How Many Parallel TCP Sessions to Open: A Pricing Perspective

Bruno Tuffin[1] and Patrick Maillé[2]

[1] IRISA-INRIA, Campus universitaire de Beaulieu
35042 Rennes Cedex, France
btuffin@irisa.fr
http://www.irisa.fr/armor/lesmembres/Tuffin/Tuffin_en.htm
[2] GET/ENST Bretagne, 2 rue de la Châtaigneraie CS 17607
35576 Cesson Sévigné Cedex, France
patrick.maille@enst-bretagne.fr

Abstract. TCP is one of the main transmission protocols used in the Internet. It has also been recently observed that opening parallel TCP sessions might be of interest for a user in order to increase his overall average throughput. We suggest in this paper to charge users per TCP session, and we investigate the resulting game in a homogeneous context: how many sessions should each user open? Given the discrete (and even finite) space of strategies, we propose to implement a probabilistic adaptation algorithm, analyze its theoretical properties and provide numerical illustrations.

1 Introduction

Nowadays, the Internet has become a common tool in daily life. Several protocols may be used to transit data, one of the most prominent being the Transmission Control Protocol (TCP), in its early version [1] or in one of its numerous more efficient versions (slow start, Reno, Vegas...). Basically, a TCP session can be modelled by an Additive-Increase Multiplicative-Decrease (AIMD) process [2], where the rate at which packets are sent increases linearly in time, but suffers a multiplicative decrease as soon as a loss is detected. Then the rate increases linearly again until the next loss, and so on.

Recently, there has been a surge of interest in opening several TCP sessions in order to increase a user's overall throughput for bulk data transfers and by then decreasing transfer time. This concept is used by applications such as GridFTP (dev.globus.org/wiki/GridFTP), or the MultTCP proposed by Oechlin and Crowcroft [3]. The question is thus, how many TCP sockets to open simultaneously? Increasing this number increases the overall throughput, but the gain can be topped by some "technological cost" or, as we will introduce, some financial cost. This induces a game between selfish users, where each user looks at the optimal number of sessions he should open, this number of sessions influencing the overall throughput of other users. The natural framework of analysis is thus the one of non-cooperative game theory (see for instance [4] for an introduction).

B. Stiller, P. Reichl, and B. Tuffin (Eds.): ICQT 2006, LNCS 4033, pp. 2–12, 2006.

In this paper, we assume that each user has to pay a fixed price per open session. Of course, a charge based on the connection duration has to be considered too, but since we consider here the system in steady-state, it can be discarded in the present analysis, and is therefore out of the scope of this paper. Our goal is to analyse the game depending of the pre-specified price (and cost). We want to study the convergence to a so-called Nash equilibrium, that is a point where no user has an interest in unilaterally changing his strategy for the number of open parallel sessions. Though, the number of choices (the number of sessions to open) is discrete, which makes the analysis a little more difficult than in a continuous context. We consider here the use of a discrete learning algorithm to solve the problem in a distributed manner [5, 6]. This algorithm adjusts the number of sessions over time using some feedback on the user's average throughput. It presents the advantages of (i) operating in a probability space to search the best number of sessions, so that it is not stuck into a local optimum (ii) being able to discover mixed strategies (iii) handling deterministic and stochastic situations (iv) being computationally simple and efficient.

In a second step, we discuss the prices that the network manager (the Internet Service Provider (ISP)) should settle in order to maximize social welfare. The corresponding *coordination ratio*, representing the loss of social welfare due to non-cooperation with respect to a centralized optimum, is also considered.

Related Work. Non-cooperative game theory has received a lot of attention in the Internet community within the last decade. It has for instance been used to model the selfish behavior of TCP users (see, among others, [7, 8]), each player of the game representing generally a single TCP session, playing with the AIMD parameters. Notice that, *in parallel to our work*, other authors have also just paid attention the game on the number of TCP connections [9]. Though, their work uses another throughput formula (applied mainly with symetric users too), with strong assumptions on the goodput at the bottleneck that we do not have to impose here. Furthermore, it does not have a pricing perspective and is mainly devoted to a continuous game and thus does not have to use a learning algorithm as we do.

Pricing has also been recently regarded as a natural way to control congestion in the Internet and to incentivize users to fairly use the resource [10, 11], but few of them have been especially devoted to the relation with TCP (see for example [12]). Again, none of them were dealing with parallel TCP sessions.

Finally, the learning algorithm that we use has successfully been applied in wireless packet networks for prediction and tracking [13, 14], as well as for power control in CDMA networks [15].

Outline. The paper is organized as follows. In Section 2 we introduce the basic model leading in [16] to the key formula for the average throughput in the case of homogeneous sessions. In the same section, we then introduce the game on the number of sessions that each selfish user should open, and present the pricing scheme. Section 3 is devoted to the theoretical analysis of the game. Section 4 illustrates the convergence of the algorithm for different values of parameters.

It also aims at finding out optimal prices when maximizing network revenue or social welfare. Finally we conclude and present some perspectives of research in Section 5.

2 Model

2.1 TCP Model

The basic model comes from [2, 16, 17]. It represents elastic users competing for bandwidth at a link of capacity C, and controlling their send rates via a additive-increase, multiplicative-decrease (AIMD) process. Those processes are often used to model the behavior of TCP sessions. We consider here a homogeneous population of AIMD users, meaning that all users have identical additive increase parameter η and multiplicative decrease parameter β. When dealing with TCP sessions, η is proportional to the inverse of the square of the round-trip time (RTT) [17], so that we assume here that they all have the same RTT.

Assume that there are N such sessions in competition. Let $X_i(t)$ be the sending rate of session i ($1 \leq i \leq N$) at time t. All sessions increase their sending rate according to the additive-increase parameter η until capacity C is reached. This corresponds to a congestion epoch if we assume that no buffering is used in the model. Then exactly one session is selected to immediately decrease its sending rate according to the multiplicative-decrease parameter β. Let T_n be the time of the n-th congestion epoch, and let $X_{i,n}$ be the send rate of user i just *after* time T_n.

The dynamics of the model is then formalized as follows. Let $Z_{i,n}$ be equal to 1 if the user i undergoes multiplicative decrease at time T_n, and $Z_{i,n} = 0$ otherwise. We have

$$X_i(t) = (1 - (1 - \beta)Z_{i,n})X_{i,n} + \eta(t - T_n), \quad T_n \leq t < T_{n+1},$$

and the send rate at congestion times obeys the recurrence, for $i = 1, 2, \ldots, N$,

$$X_{i,n+1} = (1 - (1 - \beta)Z_{i,n})X_{i,n} + \eta S_n$$

with $S_n = T_{n+1} - T_n$ obtained from $\sum_{i=1}^{N} X_{i,n+1} = C$.

It has been shown in [16] that the average aggregated throughput in steady-state is

$$\bar{x}(N) = C\left(1 - \frac{1}{1 + N\frac{1+\beta}{1-\beta}}\right). \tag{1}$$

Remarkably, this formula is shown to be true in [16] whatever the drop policy used (the choice of the session selected to decrease its rate at each congestion epochs). Several specific policies are also investigated in more details: the proportional one, where the decreased session is chosen with a probability proportional to its sending rate at the congestion epoch; the fixed one, where each one is selected with a fixed (state-independent) probability; or the largest one, where the session with the largest sending rate at a congestion epoch is selected.

One of the conclusions of formula (1) was that opening too many sessions is not worthwhile, because of the management overhead it introduces, given that for N large, the throughput increase is very small when adding a session. The goal of our paper is to investigate this point in more details in the case of users in competition, and assuming that users. Assume that we have I people in competition for the capacity C, and that user i opens N_i sessions. Then, from (1), the total number of sessions is $N = \sum_{i=1}^{I} N_i$ and, in the homogeneous case, the total throughput of user i is

$$x_i = C\frac{N_i}{N}\left(1 - \frac{1}{1 + N\frac{1+\beta}{1-\beta}}\right)$$

(with our previous notations).

2.2 Pricing Scheme and Game-Theoretic Formulation

The question asked in the paper is: what is the best strategy for user i? In other words, how many sessions should user i open? The analysis requires the framework of non-cooperative game theory assuming that he reacts selfishly, since user i's throughput depends on the total number of sessions of other users.

We additionally assume that the network operator wishes to control this number of sessions by incorporating a charge depending on the number of open sessions in order to prevent a too large number. Users' choices are then driven by their utility functions, representing a measure of the happiness or satisfaction gained from the service $U_i = f(x_i) - d(N_i)$ where

- f is the valuation function representing the gain that user i gets from a throughput x_i. We can for instance assume that $f(x_i) = \log(1 + x_i)$ [18].
- d is the charge for opening N_i sessions. It seems here also reasonable to consider it linear in N_i, $d(N_i) = \alpha N_i$. To this charge could be added a (perceived) technological cost of operating several sessions at the same time, in terms of management to "reorder" all data. We neglect it here, but it could easily be incorporated for instance by adding a fixed value α' to α.

User i's utility function is thus considered to be:

$$U_i(N_1, \ldots, N_I) = \log\left[1 + C\frac{N_i}{N}\left(1 - \frac{1}{1 + N\frac{1+\beta}{1-\beta}}\right)\right] - \alpha N_i. \tag{2}$$

The space of strategies $\mathcal{S}_i$ of each user i is then the number of sessions he can open. We assume here that, for some technological reason, this number is upper-bounded by N_{max}. We thus have $\mathcal{S}_i = \{0, \ldots, N_{max}\}$ $\forall 1 \leq i \leq I$. this discrete number of choices complicates the pure theoretical analysis of the game. A learning alogorithm is therefore used in next subsection.

2.3 A Learning Algorithm to Approach Nash Equilibrium

To solve this problem, we propose to use a decentralized discrete stochastic learning algorithm similarly to what was done in [15] for a power allocation game

in CDMA networks. The goal of each user/player is to maximize his utility. The game is played repeatedly and the optimal strategy is learned.

Player i's strategy is defined by a probability vector $p_i = (p_{i0}, \ldots, p_{iN_{max}})$ with p_{ij} the probability that user i chooses to open j sessions.

Let

$$g_i(p_1, \ldots, p_I) = \sum_{N_1,\ldots,N_I} U_i(N_1, \ldots, N_I) \prod_{j=1}^{I} p_{jN_j}$$

be the expected utility for player i given strategy probability vectors. We will say that a I-tuple of strategies $(p_1^*, \ldots, p_I^*)$ is a Nash equilibrium if $\forall 1 \leq i \leq I$ and for all probability vector p defined over $\{0, \ldots, N_{max}\}$,

$$g_i(p_1^*, \ldots, p_i^*, \ldots, p_I^*) \geq g_i(p_1^*, \ldots, p, \ldots, p_I^*).$$

Similarly to [15], we assume that the following discrete learning algorithm is used by each player:

1. Set the initial probability vector $p_i(0)$ for each user i. In this paper we will (arbitrarily) choose uniform initial distributions over $\{0, \ldots, N_{max}\}$.
2. At each time step k, the number $N_{i,k}$ of sessions open by user i is chosen according to probability vector $p_i(k)$.
3. User i then monitors his throughput x_i and computes his utility function $U_{i,k}$ at time step k.
4. User i updates his probability vector according to the rule

$$p_{ij}^{(k+1)} = \begin{cases} p_{ij}(k) - bu_{i,k}p_{ij}(k) & \text{if } j \neq N_{i,k} \\ p_{ij}(k) + bu_{i,k} \sum_{\ell \neq N_{i,k}} p_{i\ell}(k) & \text{otherwise.} \end{cases}$$

   In words, this step consists in adjusting the probability of choosing one's strategy in the next step, considering the utility brought by the current strategy: if that utility is high then the probability of the current strategy is increased, otherwise it is lowered.
5. If the algorithm has not converged goto step 2., otherwise stop.

In the algorithm, parameter b is the step size of the updating rule, and $u_{i,k}$ is a normalized utility

$$u_{i,k} = \frac{U_{i,k} - A_{i,t}}{B_{i,t} - A_{i,t}}$$

with $A_{i,t} = \min_{k \leq t} U_i(k)$ and $B_{i,t} = \max_{i,t} U_i(k)$.

Note that no knowledge of the number of players I is required, nor any specific knowledge of other users' strategies.

2.4 Social Welfare Issues

We furthermore assume that it is computationally costly for the network to support too many simultaneous TCP sessions. This can be taken into account by introducing a (converted to monetary) *network cost* per open session, that

we denote γ. Remark that this cost is not perceived by the users, who are only sensitive to their throughput and to the price they pay. The overall *social welfare* SW when each user i chooses to open N_i sessions is therefore expressed by[1]

$$SW = \sum_i f_i \left(1 + C\frac{N_i}{N}\left(1 - \frac{1}{1 + N\frac{1+\beta}{1-\beta}}\right)\right) - \gamma N, \tag{3}$$

with $N = \sum_i N_i$. All the simulations presented in this paper were run with $C = 1$, $\beta = 1/2$, $N_{max} = 5$ and $\gamma = 0.05$.

3 Game Analysis

The game studied in this paper is quite complex, and difficult to study analytically. In this section, we therefore summarize the general results and properties concerning this type of game, and state what could be expected from the use of the algorithm described in the previous section.

3.1 Game Without Pricing ($\alpha = 0$)

When no pricing is introduced, the game becomes easy to solve, since each user has a dominant strategy which consists in opening the maximum number N_{max} of TCP sessions (indeed, from (1), the larger the number of sessions user i opens, the larger his throughput is, whatever the number of other sessions). Such a Nash equilibrium may not be efficient in terms of social welfare, since the computational cost incurred to the network will be maximal. This motivates the use of pricing as a tool to incentivize users to reduce their number of open connexions.

3.2 General Game: Existence of an Equilibrium

When the number of sessions each user can open is upper-bounded by a finite value N_{max}, the game that is played between users is a finite game [19]. It is a classical result in game theory that there always exists at least a Nash equilibrium in mixed strategies for such a game. Nevertheless, for general utility functions, no results of unicity can be given.

3.3 Nash Equilibrium of the Continuous Game

When the strategy set of each user is continuous (which would mean here that a user could open a non-integer number of sessions), the concavity of utility functions can be used to prove the existence and unicity of the Nash equilibrium [20]. This is the case with the utility functions given in (2). Therefore, for any price per session α, there exists a unique Nash equilibrium. Since all users are

[1] Social welfare is defined as the sum of the utilities of all agents (users+network). Here the prices paid by users do not appear, since they are paid to the network and would appear in his utility.

identical, for that equilibrium each user would open the same number N^* of sessions, with N^* satisfying

$$N^* = \underset{x\in[0,N_{max}]}{\arg\max} \left\{ f_i\left(C\frac{x}{x+(I-1)N^*}\left(1-\frac{1}{1+(x+(I-1)N^*)\frac{1+\beta}{1-\beta}}\right)\right) - \alpha x \right\}. \tag{4}$$

3.4 Expected Outcome of the Learning Algorithm

From Theorem 1 in [15], we know that the algorithm can converge only to a point that is a Nash equilibrium of the game. The convergence of the algorithm is proved in [15] when the game has a unique pure Nash equilibrium, which is not the case here, as we will see in the following. In [6] (theorem 3.3), the convergence is established under some assumptions on the utility functions that we were not able to verify.

We ran the algorithm several times for identical initial conditions, and it turned out that, though the probability vectors p_i seem to converge (and actually to converge to pure strategies), the attained distributions may differ, illustrating the fact that there may be several Nash equilibria for the game. Two examples of those distributions with 3 players are displayed in Table 1. Those distributions correspond to pure strategies (Dirac distributions on a single point), leading to different Nash equilibria.

An interesting remark (from the left-hand side of Table 1) is that two identical players may have different optimal strategies at equilibrium, depending on others' choices (due to the discrete nature of the game).

Table 1. Two different outcomes (strategy probability vectors $p_{i,\cdot}$ after 10000 rounds) of the same algorithm with the same initial conditions ($b = 0.03, \alpha = 0.1$)

# sessions	Player 1	Player 2	Player 3
0	0	0	0
1	1	0	1
2	0	1	0
3	0	0	0
4	0	0	0
5	0	0	0

# sessions	Player 1	Player 2	Player 3
0	0	0	0
1	0	0	0
2	1	1	1
3	0	0	0
4	0	0	0
5	0	0	0

4 Numerical Results

This section aims at illustrating the behavior of the learning algorithm, and at highlighting the interest of pricing in the considered context.

Figure 1 shows a trajectory of the number of sessions chosen by three players in competition up to $k = 300$, with $\alpha = 0.1$ and $b = 0.1$. The curve suggests the convergence to a Dirac distribution for each player, which corresponds to what was observed in Table 1. Again, identical users may have different equilibrium values due to the discrete nature of the game.

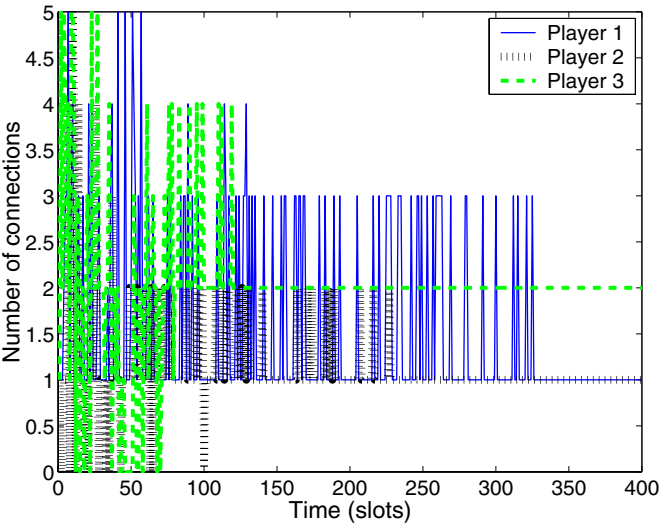


Fig. 1. Number of sessions used by three players in competition: convergence phase ($b = 0.1, \alpha = 0.1$)

Table 2 shows how the number I of players affects the equilibrium probability. We have chosen to show the mean distribution (all Dirac, but at different values) over all players. As expected, the number of open sessions tends to decrease as I increases, since the marginal throughput gain of opening an additional session is smaller, whereas the marginal cost remains fixed to $\alpha = 0.1$.

Table 3 illustrates the modifications on the aggregated strategy distributions for different values of price α. As expected, we observe a decrease in the number of sessions when the price increases. The impact on the revenue is shown in the last line of Table 3. Due to the discontinuity of the equilibrium strategies in the per session price α (those strategies are Diracs as previously observed), the revenue is not a concave function of the per session price, and therefore it is not obvious to a network manager to discover the price that will yield the largest revenue.

The introduction of prices aims at incentivizing users to better use the network. In other terms, we expect pricing to help reduce the *coordination ratio*,

Table 2. Equilibrium probabilities for various numbers of players, with $\alpha = 0.1$

Number of TCP sessions	$I = 2$	$I = 4$	$I = 6$	$I = 8$	$I = 10$
0	0	0	0	0.125	0.2
1	0	0.5	0.833	0.875	0.8
2	1	0.5	0.167	0	0
3	0	0	0	0	0
4	0	0	0	0	0
5	0	0	0	0	0

Table 3. Equilibrium probabilities (aggregated distribution) for various prices α

Number of TCP sessions	$\alpha = 0$	$\alpha = 0.05$	$\alpha = 0.1$	$\alpha = 0.2$	$\alpha = 0.3$
0	0	0	0	0.333	0.333
1	0	0	0.333	0.333	0.667
2	0	0	0.667	0.333	0
3	0	0.333	0	0	0
4	0	0.667	0	0	0
5	1	0	0	0	0
Corresponding revenue	0	0.55	0.5	0.6	0.6

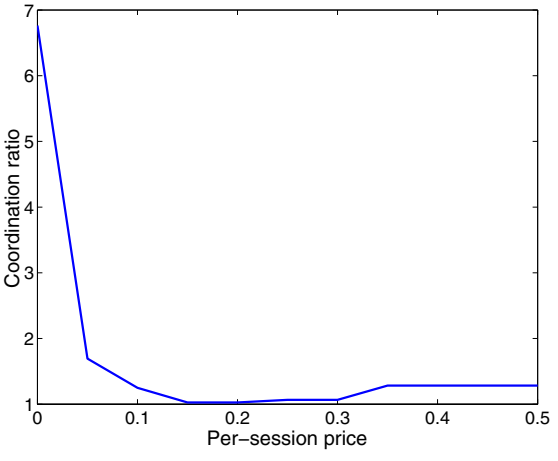


Fig. 2. Coordination ratio for different values of the charge per session (3 players, $b = 0.05$)

defined as the ratio of the maximum attainable social welfare and the actually reached social welfare (at a Nash equilibrium).

Remark: several Nash equilibria are likely to exist in the discrete game we are studying. The coordination ratio gives the loss of efficiency with respect to a centralised decision corresponding to the equilibrium attained by the algorithm. Depending on the algorithm progress, different equilibria can be reached as highlighted in Table 1, giving different values of the coordination ratio. For that reason, Koutsoupias and Papadimitriou [21] suggested to take the Nash equilibrium with the worst social welfare. The coordination ratio associated to the worst Nash equilibrium is called *price of anarchy*. In this paper, we only plot the coordination ratio with the equilibrium given by the learning algorithm, since we cannot derive all Nash equilibria.

Here, computing the maximal social welfare can be hard when considering discrete strategy sets. We therefore computed an upper bound, corresponding to the continuous strategy set case. The effects of the charging factor α on

the coordination ratio can be seen in Figure 2. As expected, the case where the per session charge is null is very bad, since it corresponds to every user opening N_{max} sessions. We also remark that for a judiciously chosen value of α, the social welfare is very close to the optimum value that could be reached if users would collaborate (the coordination ratio is very close to 1). The shape of the coordination ratio curve plotted in Figure 2 is typical of congestion games: while the introduction of prices helps lowering the demand and leads to a better utilisation of resources by reducing the negative externalities (descending phase of the curve), setting too large charges prevents users from entering the game and the resource becomes underused (ascending phase of the curve).

We can therefore conclude that introducing a very simple pricing scheme (fixed per-session price) can lead the initially inefficient Nash equilibrium to an efficient one (where efficiency is in the sense of social welfare), therefore arguing in favor of the use of pricing in such contexts.

5 Conclusions

In this paper, we have considered a game where TCP users compete for bandwidth by opening parallel sessions in order to increase their QoS. The game is controlled by imposing a fixed charged for each open session. We consider that each user implements a (decentralized) discrete learning algorithm to find out his best strategy, and convergence to a Nash equilibrium is discussed. We have also discussed the pricing strategy for the network manager.

As directions for future research, we aim at investigating the more realistic case of heterogeneous sessions, with different round-trip times, meaning different additive-increase parameters. A closed-form expression for the average throughput does not exist in full generality yet, but we could numerically look at the resulting equilibrium, as well as at pricing solutions for incentivizing users to fairly share the ressource (this pricing scheme probably using the RTT values).

Acknowledgement

The authors would like to thank R. Chandramouli and Y. Xing for some discussions on the learning algorithm used in the paper.

References

1. Jacobson, V.: Congestion avoidance and control. In: ACM SIGCOMM. (1988) 314–329
2. Baccelli, F., Hong, D.: AIMD, Fairness and Fractal Scaling of TCP Traffic. In: Proceedings of IEEE INFOCOM 02. (2002)
3. Crowcroft, J., Oechslin, P.: Differentiated End-to-End Internet Services using a Weighted Proportional Fair Sharing TCP. ACM Computer Communications Review **47**(4) (2004) 275–303
4. Osborne, M., Rubenstein, A.: A Course on Game Theory. MIT Press (1994)

5. Narendra, K., Thathachar, M.: Learning Automata: An Introduction. Englewood Cliffs: Prentice Hall (1989)
6. Sastry, P., Phansalkar, V., Thathachar, M.: Decentralized learning of Nash equilibria in multi-person stochastic games with incomplete information. IEEE Trans. Systems, Man, and Cybernetics **24** (1994) 769–777
7. Akella, A., Seshan, S., Karp, R., Shenker, S.: Selfish behavior and stability of the internet: A game-theoretic analysis of tcp. In: Proceedings of ACM SIGCOMM 02. (2002)
8. Altman, E., Jiménez, T., Núñez Queija, R.: Analysis of two competing TCP/IP connections. Performance Evaluation **49**(1) (2002) 43–56
9. Zhang, H., Towsley, D., Gong, W.: TCP connection game: A study on the selfish behavior of TCP users. In: Proc. of 13th IEEE Intl. Conf. on Network Protocols (ICNP'05), Boston, Massachusetts, USA (2005)
10. Courcoubetis, C., Weber, R.: Pricing Communication Networks—Economics, Technology and Modelling. Wiley (2003)
11. Tuffin, B.: Charging the Internet without bandwidth reservation: an overview and bibliography of mathematical approaches. Journal of Information Science and Engineering **19**(5) (2003) 765–786
12. Altman, E., Barman, D., El Azouzi, R., Ros, D., Tuffin, B.: Pricing Differentiated Services: A Game-Theoretic Approach. Computer Networks (2005 (to appear))
13. Chandramouli, R.: A stochastic technique for on-line prediction and tracking of wireless packet networks. In: Proceedings of the Thirty-Fifth Asimolar Conference on Signals, Systems and Computers. (2001) 672–676
14. Kiran, S., Chandramouli, R.: An adaptive energy efficient link layer protocol using stochastic learning control. In: Proceedings of the IEEE International Conference on Communications (ICC). (2003)
15. Xing, Y., Chandramouli, R.: Stochastic learning solution for distributed discrete power control game in wireless data networks. Technical report, Stevens Institute of Technology (2004)
16. Altman, E., Barman, D., Tuffin, B., Vojnović, M.: Parallel TCP Sockets: Simple Model, Throughput and Validation. In: IEEE INFOCOM 2006, Barcelona, Spain (2006)
17. Altman, E., El Azouzi, R., Ros, D., Tuffin, B.: Loss strategies for competing TCP/IP connections. Computer Networks (2005 (to appear))
18. Kelly, F.: Charging and rate control for elastic traffic. European transactions on Telecommunications **8** (1997) 33–37
19. Fudenberg, D., Tirole, J.: Game Theory. MIT Press, Cambridge, Massachusetts (1991)
20. Rosen, J.: Existence and uniqueness of equilibrium points for concave n-person games. Econometrica **33**(3) (1965) 520–534
21. Koutsoupias, E., Papadimitriou, C.: Worst-case equilibria. In: Proc of 16th Annual Symposium on Theoretical Aspects of Computer Science (STACS 1999). Volume 1563 of *Lecture Notes in Computer Science*. (1999) 404–413

Performance Modeling on the Interaction of ISPs

Sam C.M. Lee, Joe W.J. Jiang, and John C.S. Lui

Department of Computer Science & Engineering,
The Chinese University of Hong Kong
{cmlee, wjjiang, cslui}@cse.cuhk.edu.hk

Abstract. In this paper, we consider interactions of Internet Service Providers (ISPs) and how these interactions can affect the overall traffic and resource allocation between ISPs. In particular, we consider a simplified two-level hierarchical model wherein there are a single tier-1 ISP and $N > 1$ tier-2 ISPs. Each tier-2 ISP needs to pay the tier-1 ISP for the connectivity service. At the same time, a tier-2 ISP can also arrange to have "private peering" links with other tier-2 ISPs. Therefore, each tier-2 ISP can optimize its utility by deciding on the proper traffic routing of transmitting traffic, either via the tier-1 ISP link, or via the private peering link with other tier-2 ISPs. The tier-1 ISP, on the other hand, needs to decide on the proper resource allocation for all its tier-2 peers so as to avoid monopolization of its resource by a single peer (i.e., to achieve customer diversity). We investigate a distributed framework wherein a tier-1 ISP can achieve customer diversity while tier-2 peers can perform their utility maximization in terms of traffic routing. We also explore other important issues such as sensitivity and convergency. Extensive simulations are carried out to quantify the merits of the proposed distributed framework.

Keywords: Interaction, Utility Maximization, Sensitivity and Convergency.

1 Introduction

Current Internet is basically hierarchical in nature: there are many tier-1 Internet Service Providers (ISPs) providing backbone connectivity service. Regional ISPs, which are usually called the tier-2 ISPs, are *customers* to these tier-1 ISPs and they pay for the connectivity service. There are some smaller ISPs, which are called the tier-3 ISPs, are then connected to tier-2 ISPs for connectivity and/or local access. Often times, ISPs of the same tier (or level) may negotiate with each other so as to provide mutual connectivity. Whenever two peers of the same tier are connected, we say that these two peers have established a *private peering* relationship. In essence, a private peering relationship allows two peers to transfer traffic for their local customers without going through their providers at the higher tier. Figure 1 illustrates the hierarchy and various connectivity relationships.

There are two common ways for a local ISP (which we call peer from now on) to gain the Internet access. The first way is to transmit traffic via the connectivity service provided by its providers (or higher tier ISPs). For this form of traffic transmission, cost will be incurred since the local ISP needs to pay for the amount of traffic transmitted via the *provider-customer* link. The other way to gain the Internet access is to transmit the

B. Stiller, P. Reichl, and B. Tuffin (Eds.): ICQT 2006, LNCS 4033, pp. 13–24, 2006.

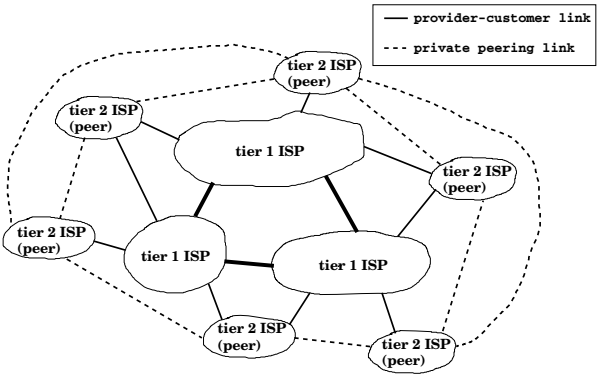


Fig. 1. Internet hierarchical relationship and various connectivity relationships

traffic via the private peering link, and that the traffic is destined to that particular local ISP. Since there are two ways to transmit traffic, a local ISP often needs to decide on the proper traffic routing: the amount of traffic transmission via the provider-customer link and private peering links, so as to minimize its operating cost and at the same time, satisfy some bandwidth or quality-of-service constraints. It is important to note that for two peers that are geographically apart, it may not be economical or even possible to establish a private peering link, therefore, connectivity between these two peers will be provided by the higher tier ISPs.

The tier-1 ISP (which we call ISP from now on), needs to set a proper price for each of its customers so as to attract peers to transmit traffic via the provider-customer link. If the price is set too high, peers may opt to transmit via their peering links. On the other hand, if the price is set too low, then the ISP may not be able to maximize its profit. Another important issue that an ISP needs to consider is proper resource provisioning so that no peer can monopolize the connectivity resource. It is obvious that there exists certain level of *interaction* between peers and this interaction can affect the decision of the ISP. In particular, the level of pricing and resource allocation can affect the routing decisions of peers, and the routing decisions by peers also affect the level of resource allocation set by an ISP.

The contribution of our paper is to provide an *understanding* of this form of interaction, in particular, how it can affect the routing strategy of individual peers and resource allocation of an ISP. We propose a distributed algorithm wherein peers and ISP can communicate so that a peer can maximize its utility while the ISP can provide fair resource allocation. We also show that the distributed algorithm is stable and can converge to an equilibrium point quickly.

Let us summarize some related work in this topic. There is a rich literature on Internet pricing [6, 8, 4, 5], but they mainly concern about the pricing strategy of individual customers, i.e., access charging. Our work focuses on the interaction of a tier-1 ISP and its customers, namely, tier-2 ISPs. Authors in [1] present the revenue maximization and scalability of a service provider. It shows that using the suggested pricing scheme, there are sufficient incentives for an ISP to upgrade its network. However, they do not consider the interaction between peers, namely, peers can exchange traffic via private peering links.

The organization of the paper is as follows. In Section 2, we present our mathematical model of representing the interaction of the ISP and its peers. We formulate the maximization function for a peer and show how each peer can perform routing so as to maximize its utility. Conditions of maximization are also presented. We also present the resource allocation algorithm of the ISP to achieve customer diversity. In Section 3, we investigate the convergency issue of the distributed algorithm. In Section 4, we investigate the sensitivity of the system behavior on various system parameters. Finally, conclusion is given in Section 5.

2 The Mathematical Model, Distributed Maximization and Resource Allocation

Let us consider a simplified two-tiers hierarchical model in which there is one tier-1 ISP and N tier-2 ISPs (or peers).

Peers can communicate with each other either by sending traffic through the ISP, or by the private peering links between two peers. To provide connectivity, the ISP has a communication network which has a total capacity of $\mathcal{R}$ (in units of Mbps). For a peer $i \in \{1, 2, \ldots, N\}$, it has a link to the ISP, and *possibly* $N-1$ private peering links connecting to the other $N-1$ peers. Let l_{ij} denote the private peering link between peer i and peer j and this link has a capacity of r_{ij} (in unit of Mbps). Note that when $r_{ij} = 0$, it implies that there is no private peering link between peer i and peer j. The link connecting peer i and the ISP is denoted as l_{ii}, and the ISP allocates $\mathcal{R}_i$ amount of bandwidth (in units of Mbps) for this connection. It is important to point out that our model can be viewed as a generalization of the network model presented in [1], in which private peering links are not considered and so there is no interaction between peers. Note that we only have one ISP in our model. The issues of multiple ISPs and multihoming are much more complicated and will appear in our future work. Lastly, Table 1 contains all notations used in our mathematical model.

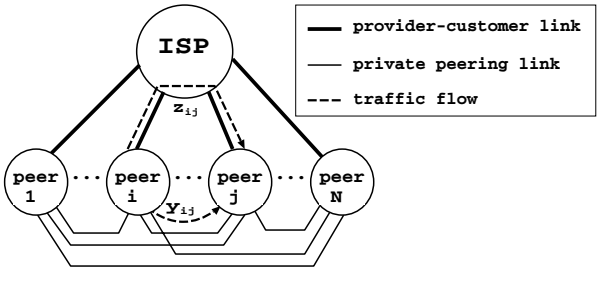


Fig. 2. A simplified two-tiers hierarchical model with one tier-1 ISP and N peers

Let x_{ij} denote the traffic demand (or transmission rate in unit of Mbps) *from* peer i *to* peer j. In essence, it is the traffic originated from peer i and destined to peer j. If the peer i can sustain the transmission rate of x_{ij}, peer i receives a utility of $F_{ij}(x_{ij})$, where F_{ij} is a strictly concave function in x_{ij}. The utility $F_{ij}(x_{ij})$ represents the degree

Table 1. Mathematical notations

N :	Number of peers (the local ISPs) in the communication network.
l_{ii} :	An abstraction of the communication link between peer i and the ISP.
l_{ij} :	The private communication link connecting peer i to peer j.
$\mathcal{R}$:	Total capacity of the ISP link.
$\mathcal{R}_i$:	Allocation of ISP's link bandwidth to peer i.
r_{ij} :	Capacity of the private link l_{ij} connecting peer i to peer j.
x_{ij} :	Traffic demand or transmission rate from peer i to peer j, such that $x_{ij} = y_{ij} + z_{ij}$.
y_{ij} :	Traffic transmission rate from peer i to peer j going through the private link l_{ij}.
z_{ij} :	Traffic transmission rate from peer i to peer j going through the ISP link l_{ii}.
z_i :	Aggregate traffic rate that peer i sends through the ISP link.
$\bar{z}$:	Aggregate traffic rate through the ISP link from all peers.
$\mathcal{P}_i$:	Price per unit bandwidth on the ISP link for peer i. In this work we assume $\mathcal{P}_i = \mathcal{P}$ for all i.
p_{ij} :	Price per unit bandwidth of the private peering link l_{ij}.
$\boldsymbol{y}_i$:	$\boldsymbol{y}_i = (y_{i1}, y_{i2}, \ldots, y_{in})$ denotes the traffic rate vector for peer i through its private links.
$\boldsymbol{z}_i$:	$\boldsymbol{z}_i = (z_{i1}, z_{i2}, \ldots, z_{in})$ denotes the traffic rate vector for peer i through the ISP link.

of happiness of peer i by sending data to peer j at the rate of x_{ij}. Noted that concave function is commonly used to represent elastic traffic[2], which is the dominant traffic in the current Internet.

This constant traffic demand x_{ij} can either go through the ISP link l_{ii}, or the private link l_{ij}. We denote y_{ij} as the traffic rate that peer i decides to transmit through the private link l_{ij}, and z_{ij} as the traffic rate through the ISP link l_{ii}. In other words, we have: $x_{ij} = y_{ij} + z_{ij}$, with $y_{ij}, z_{ij} \geq 0$, for $i, j \in \{1, \ldots, N\}$.

Let x_{ii} be the traffic demand from peer i to destinations other than the $N-1$ peers. This represents traffic to other part of the Internet wherein peer i has to send the data through the ISP. In this case, peer i can only use the provider-customer link l_{ii} for the traffic transmission. Therefore, we have the following relationship: $y_{ii} = 0$ and $x_{ii} = z_{ii}$ for $i \in \{1, \ldots, N\}$. For the ease of presentation, let $z_i = \sum_{j=1}^{N} z_{ij}$ denote the aggregate traffic demand that peer i sends through the ISP link, and let $\bar{z} = \sum_{j=1}^{N} z_j$ denote the aggregate traffic on the ISP network from all N peers.

To transmit data across the ISP, peers need to pay the network operators for the transmission service. The price per unit bandwidth through the ISP link l_{ii} is $\mathcal{P}_i$, which is determined by the ISP. Peer i can also send the traffic y_{ij} through the private link l_{ij}, and the price per unit bandwidth is p_{ij}, which can be mutually agreed upon between peers i and j. In this work, we do not consider issues on the cost of setting up peering links, since it is not part of the operating cost. We assume peers can utilize existing peering links with fixed capacities r_{ij}. For convenience, we denote $\boldsymbol{y}_i = (y_{i1}, y_{i2}, \ldots, y_{iN})$ as the traffic rate vector for peer i, representing the traffic going through its private peering links and $\boldsymbol{z}_i = (z_{i1}, z_{i2}, \ldots, z_{iN})$ as the traffic rate vector for peer i through the ISP link. We denote $\boldsymbol{\mathcal{P}} = (\mathcal{P}_1, \mathcal{P}_2, \ldots, \mathcal{P}_N)$ as the price vector set by the ISP for different peers.

In making the routing decision, each peer not only needs to consider the cost of transmitting the traffic, but also on the quality of service. In other words, each peer needs to take into consideration of the delay or congestion cost on the links. In this

work, we assume each link is represented by an M/M/1 model as in [1], and one can take the average delay on the link as its congestion indicator.[1] Rather than informing all peers about the current transmission demands $\bar{z}$ on the ISP link (this is considered as a confidential information by a peer), the ISP will compute and announce its bandwidth allocation to peer i as $\mathcal{R}_i$. Under this form of setting, the congestion cost $\mathcal{D}_{ij}$ of a link l_{ij} can be represented as:

$$\mathcal{D}_{ij} = \begin{cases} \frac{1}{r_{ij}-y_{ij}} \text{ if } i \neq j, \\ \\ \frac{1}{\mathcal{R}_i - z_i} \text{ if } i = j. \end{cases}$$

The traffic demand x_{ij} can be viewed as the long-term average aggregate request from the customers of peer i destining to peer j. For example, average of a monthly traffic from a peer. So the traffic demand is considered as a constant within a certain period. Consider the case when peer i can always obtain a *sufficient* bandwidth capacity to transmit all the aggregate requests, i.e. $\sum_j x_{ij} \leq \sum_j r_{ij} + \mathcal{R}_i$, then the peer will transmit all the requests, while maximizing its utility at the same time. With fixed traffic demands x_{ij}'s, the aggregate happiness $\sum_j F_{ij}(x_{ij})$ is therefore a constant.

Let us now formulate the objective function of a peer, say i. Peer i wants to maximize the following function:

$$\text{Max } U_i = \sum_j F_{ij}(x_{ij}) - \mathbf{1}_{\{y_{ij}\neq 0\}}\Big[\frac{1}{r_{ij}-y_{ij}}\Big] - \sum_{j\neq i} p_{ij}y_{ij} - \mathbf{1}_{\{z_i\neq 0\}}\Big[\frac{1}{\mathcal{R}_i - z_i}\Big] - \mathcal{P}_i z_i$$

$$\text{Max } U_i = K - \sum_{j\neq i}\mathbf{1}_{\{z_{ij}\neq x_{ij}\}}\Big[\frac{1}{r_{ij}-x_{ij}+z_{ij}}\Big] + \sum_{j\neq i} p_{ij}z_{ij} - \mathbf{1}_{\{z_i\neq 0\}}\Big[\frac{1}{\mathcal{R}_i - z_i}\Big] - \mathcal{P}_i z_i$$

$$\text{Min } \mathcal{C}_i = \sum_{j\neq i}\mathbf{1}_{\{z_{ij}\neq x_{ij}\}}\Big[\frac{1}{r_{ij}-x_{ij}+z_{ij}}\Big] - \sum_{j\neq i} p_{ij}z_{ij} + \mathbf{1}_{\{z_i\neq 0\}}\Big[\frac{1}{\mathcal{R}_i - z_i}\Big] + \mathcal{P}_i z_i \qquad (1)$$

$$\text{s. t.} \quad \max\{0, x_{ij} - r_{ij}\} \leq z_{ij} \leq x_{ij} \quad \text{for all } j \neq i, \quad z_{ii} = x_{ii}, \quad \sum_j z_{ij} \leq \mathcal{R}_i \qquad (2)$$

where $K = \sum_j F_{ij}(x_{ij}) - \sum_{j\neq i} p_{ij}x_{ij}$ is a constant.

In here, $\mathbf{1}_{\{p\}}$ is an indicator function. The objective of the optimization problem (1) is to minimize the aggregate congestion costs and payments under constant traffic demands. The variable transmission rate vector $\boldsymbol{y}_i$ is absorbed and the remaining variable in the new optimization problem is $\boldsymbol{z}_i$. The constraints in Equation (2) represent the feasible region of the ISP link transmission rates. The first constraints give the lower and upper bounds for z_{ij}'s. When $r_{ij} \geq x_{ij}$, the bandwidth in the private peering link l_{ij} is larger than the demand x_{ij}, i.e. private peering link capacity is sufficient for the demand and so the minimum transmission rate in ISP link z_{ij} is zero. When $r_{ij} < x_{ij}$, the bandwidth in the private peering link is insufficient for the demand and so part of the traffic must go through the ISP link. It makes the minimum value of $z_{ij} = x_{ij} - r_{ij}$. The second constraint again is due to the absence of private peering link to the "outsiders". The third constraint is the ISP link capacity constraint.

It is important to point out that the optimization process is indeed a *coupled* optimization process. For each peer i, given the bandwidth allocation $\mathcal{R}_i$ of the ISP link,

[1] We have also investigated in using expected waiting time in the peers' utility functions. Please refer to our technical report[3].

it performs an optimization and determines its optimal rates $\boldsymbol{z}_i$ and will send a bid for this bandwidth allocated by the ISP. After collecting the bidding information from all peers, the ISP calculates the bandwidth allocation according to the biddings. Peers will offer their new biddings based on the allocated bandwidth by the ISP. We model this interaction process as a *non-cooperative game* wherein each peer offers a bid to the ISP so as to minimize its own cost.

For a given ISP price vector $\boldsymbol{\mathcal{P}} = (\mathcal{P}_1, \mathcal{P}_2, \ldots, \mathcal{P}_N)$, this defines a non-cooperative game between these N peers [7]. These peers interact with each other and determine their optimal transmission rates periodically and asynchronously. For each price vector $\boldsymbol{\mathcal{P}} > 0$, a Nash equilibrium point for this N-peers game is defined as N-tuple $\boldsymbol{z}^* = (\boldsymbol{z}_1^*, \boldsymbol{z}_2^*, \ldots, \boldsymbol{z}_N^*)$, such that for *all* peers $i \in \{1, 2, \ldots, N\}$:

$$\mathcal{C}_i(\boldsymbol{z}^*, \boldsymbol{\mathcal{P}}) \leq \mathcal{C}_i(\boldsymbol{z}, \boldsymbol{\mathcal{P}}) \tag{3}$$

for any other feasible traffic vector $\boldsymbol{z} = (\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_N)$ that satisfies the constraints defined in Equation (2).

2.1 Distributed Solution of the Minimization Problem

In the following, we illustrate how a peer, say i, can determine its transmission rates, that is $\boldsymbol{z}_i$, rates to other peers via the ISP's link, as well as $\boldsymbol{y}_i$, rates to other peers via private peering links, so as to minimize its cost when the bandwidth supply is *sufficient*. Assuming that the peer knows the price $\mathcal{P}_i$ specified by the ISP and the associated bandwidth allocation $\mathcal{R}_i$, one can model an individual peer's behavior as a convex optimization problem as defined in Equation (1). Let us first study the necessary and boundary conditions for a peer to minimize the cost.

Necessary Conditions with Positive Transmission Rate
Since the cost $\mathcal{C}_i$ is discontinuous at $z_{ij} = x_{ij}$ (i.e., transmission rate through the private peering link l_{ij} is zero) and $z_i = 0$ (i.e., transmission rate through the ISP link is zero), we first investigate the necessary conditions when $z_{ij} \neq x_{ij}$ and $z_i \neq 0$. The optimization problem of Equation (1) has $N - 1$ variables (with $z_{ii} = x_{ii}$). The first and second order partial derivatives with respect to z_{ij} and z_{ik} for $k \neq j \neq i$ are:

$$\begin{aligned}
\frac{\partial \mathcal{C}_i}{\partial z_{ij}} &= \frac{-1}{(r_{ij} - x_{ij} + z_{ij})^2} - p_{ij} + \frac{1}{(\mathcal{R}_i - z_i)^2} + \mathcal{P}_i, \\
\frac{\partial^2 \mathcal{C}_i}{\partial z_{ij}^2} &= \frac{2}{(r_{ij} - x_{ij} + z_{ij})^3} + \frac{2}{(\mathcal{R}_i - z_i)^3} > 0, \\
\frac{\partial^2 \mathcal{C}_i}{\partial z_{ij} \partial z_{ik}} &= \frac{2}{(\mathcal{R}_i - z_i)^3} > 0.
\end{aligned}$$

This shows that the Hessian matrix of the objective function in Equation (1) is positive definite on the non-negative space bounded by the capacity constraints $x_{ij} - r_{ij} \leq z_{ij} \leq x_{ij}$ and $z_i \leq \mathcal{R}_i$. So the cost $\mathcal{C}_i$ is strictly convex in z_{ij} for all $j \neq i$. The minimum cost and optimizer to this problem is unique and can be found by the Lagrangian method. The necessary conditions of z_{ij} for the minimization of $\mathcal{C}_i$ are:

$$\frac{\partial \mathcal{C}_i}{\partial z_{ij}} \begin{cases} > 0 & \text{if } z_{ij} = 0, \\ = 0 & \text{if } z_{ij} > 0. \end{cases} \tag{4}$$

Boundary Cases to Minimization Problem

Due to the discontinuity of the objective function, the necessary conditions given above may not achieve the global minimum. In here we explore the boundary cases when the transmission rates are zero, i.e., $z_{ij} = x_{ij}$ or $z_i = 0$. Figures 3 and 4 show these cases. Figure 3 corresponds to the case when $z_{ij}^* = \arg\{\frac{\partial \mathcal{C}_i}{\partial z_{ij}} = 0\}$ is in the feasible range. The vertical axis shows the aggregate cost $\mathcal{C}_i$ and the horizontal axis shows the transmission rate z_{ij}. Figure 3(a) considers if $x_{ij} \leq r_{ij}$, which implies the private peering link capacity is adequate for the transmission demand; and Figure 3(b) considers if $x_{ij} > r_{ij}$, which implies the private peering link capacity is inadequate for the transmission demand. In Figure 3, the minimum point of the curve is at $P1$ when $z_{ij} = z_{ij}^*$. We first consider the upper bound. When $z_{ij} = x_{ij}$, the transmission rate goes through the ISP link only. The congestion cost in peering link l_{ij} is not considered and is subtracted from $\mathcal{C}_i$, so $P3$ rather than $P2$ is the point of $\mathcal{C}_i$ when $z_{ij} = x_{ij}$. We then consider the lower bound under two cases: case i) when $x_{ij} \leq r_{ij}$ (as in Figure 3(a)), the minimum value of $z_{ij} = 0$. If there is a $z_{ik} > 0$ for some $k \neq j$, $P4$ is the point when $z_{ij} = 0$. But if the aggregate traffic through the ISP link is zero ($z_i = 0$), $P5$ is the point when $z_{ij} = 0$. Note that the congestion cost in the ISP link is subtracted from $\mathcal{C}_i$ in this case; case ii) when $x_{ij} > r_{ij}$ (as in Figure 3(b)), the minimum value of $z_{ij} = x_{ij} - r_{ij}$. This is because the maximum amount of traffic through the private peering link is r_{ij}, the remaining rate $x_{ij} - r_{ij}$ has to go through the ISP link and the congestion cost in the ISP link must be considered. In general, when z_{ij}^* is in the feasible range, the optimal transmission rate is either $z_{ij} = 0$, $z_{ij} = x_{ij}$ or $z_{ij} = z_{ij}^*$.

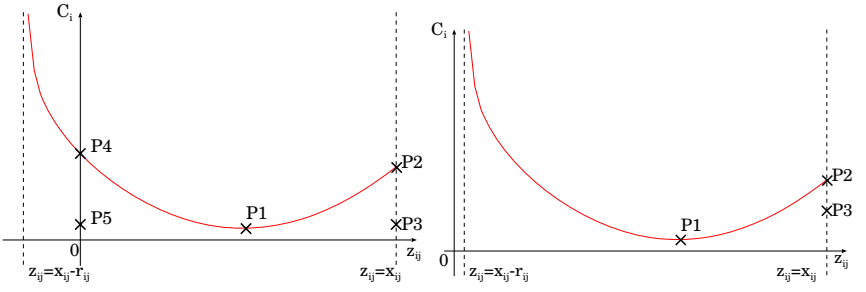


Fig. 3. Peer i's cost against transmission rate z_{ij} when $z_{ij}^* = \arg\{\frac{\partial \mathcal{C}_i}{\partial z_{ij}} = 0\}$ is in the feasible range: (a) when $x_{ij} \leq r_{ij}$, (b) when $x_{ij} > r_{ij}$

Figure 4 illustrates when $z_{ij}^* = \arg\{\frac{\partial \mathcal{C}_i}{\partial z_{ij}} = 0\}$ is not in the feasible range. Figure 4(a) considers when $z_{ij}^* \leq \min\{0, (x_{ij} - r_{ij})\}$. As $\mathcal{C}_i$ is strictly convex in z_{ij}, the minimum feasible $z_{ij} = \min\{0, (x_{ij} - r_{ij})\}$, is either at $P4$ (when $z_i > 0$) or at $P5$ (when $z_i = 0$). For the upper bound of z_{ij}, when $z_{ij} = x_{ij}$, the congestion cost in the private peering link is subtracted from the cost. This concludes that the minimum point of $\mathcal{C}_i$ is either when $z_{ij} = \min\{0, (x_{ij} - r_{ij})\}$ (optimizer is either $P4$ or $P5$) or when $z_{ij} = x_{ij}$ (optimizer is $P3$). Figure 4(b) shows the case when $z_{ij}^* > x_{ij}$. The maximum feasible $z_{ij} = x_{ij}$ due to the convexity of $\mathcal{C}_i$. For the lower bound of z_{ij},

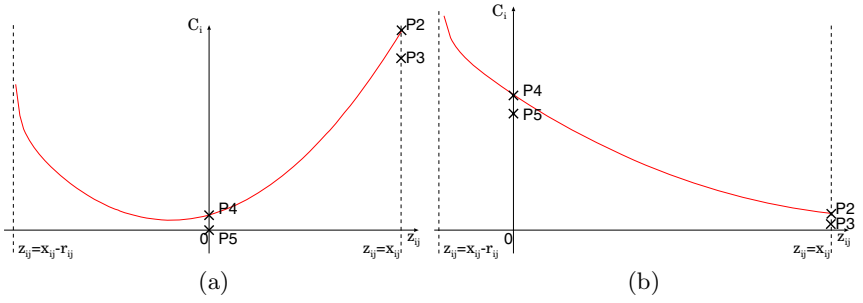


Fig. 4. Peer i's cost against transmission rate z_{ij} when $z_{ij}^* = \arg\{\frac{\partial \mathcal{C}_i}{\partial z_{ij}} = 0\}$ is not in the feasible range: (a) when $z_{ij}^* \leq \min\{0, (x_{ij} - r_{ij})\}$, (b) when $z_{ij}^* \geq x_{ij}$

when $z_i = 0$ (which implies $z_{ij} = 0$), the value of $\mathcal{C}_i$ at $P5$ may be smaller than that at $P3$. This concludes that the minimum point of $\mathcal{C}_i$ is either $P3$ when $z_{ij} = x_{ij}$ or $P5$ when $z_{ij} = 0$. Lastly, after the ISP link transmission rates z_{ij}'s are computed, the private peering link transmission rates y_{ij}'s can be found by $y_{ij} = x_{ij} - z_{ij}$.

2.2 Distributed Resource Allocation by ISP

Let us present the algorithm in which the ISP can determine the appropriate capacity $\mathcal{R}_i$ for every peer i, for $i \in \{1, \ldots, N\}$. At the beginning, the ISP distributes its capacity equally among all peers, so $\mathcal{R}_i = \frac{\mathcal{R}}{N}$ for all i, and sends the distribution $\mathcal{R}_i$ to every peer i. Upon receiving the information, each peer i applies the procedure described in the previous sub-section to compute its own optimal transmission rates ($\boldsymbol{y}_i$ and $\boldsymbol{z}_i$) and sends the information ($z_i = \sum_j z_{ij}$) to the ISP as its bidding for the ISP capacity. The ISP gathers the biddings from the peers. Then it allocates the resource represented by the following formula:

$$\mathcal{R}_i = z_i + \frac{(\mathcal{R} - \bar{z})}{N}.$$

3 Convergency of Traffic Rates

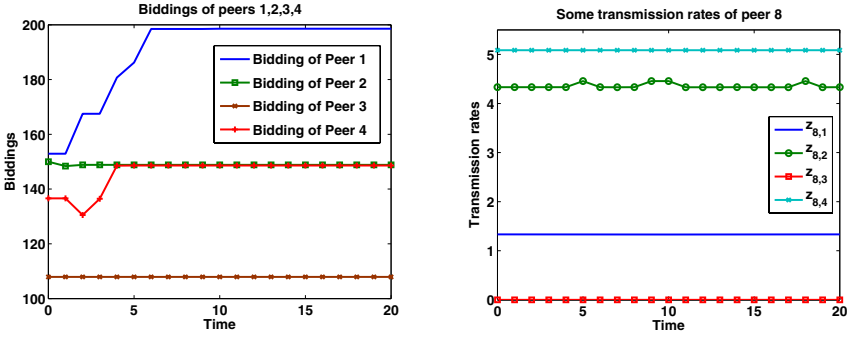
With the traffic distribution algorithm by peers and resource allocation by the ISP we described in the previous section, one important issue that we need to address is whether these traffic rates and biddings will converge. In this section, we investigate the convergency of rates and biddings of peers when the number of peers is large.

The experiment considers the case when *all* peers are of similar sizes and thus have similar traffic demands. We show that the traffic rates distributions and biddings of all peers converge rapidly. We have also investigated in the case when *some* of the peers are of *larger and smaller* sizes. For more details, please refer to our technical report [3].

The environment of the experiment is constructed as follows. There is a network of $N = 50$ peers and one ISP. Each peer has an aggregate link connecting to the ISP and peering links connecting to the other peers. A peering link l_{ij} exists with a probability of 0.5. If the link l_{ij} exists, its capacity r_{ij} is chosen from a random variable

uniformly distributed in $[0, 10]$, and its unit price p_{ij} is another random variable uniformly distributed in $[1, 2]$. The traffic demand x_{ij} is a random variable equal to 0 with probability 0.3 and uniformly distributed in $[0, 10]$ with probability 0.7. The ISP provides a bandwidth of $\mathcal{R} = 7500$ units with unit price $\mathcal{P} = 2.5$. After each period of one second, the ISP applies the resource allocation algorithm we described in the previous section, and sends the distribution $\mathcal{R}_i$ to every peer i. Upon receiving the signal from the ISP, peer i performs the traffic rate distribution algorithm with the parameters $\mathcal{R}_i$ and $\mathcal{P}_i = \mathcal{P}$. To introduce noise in the information exchange, a bidding is successfully transmitted to the ISP with a probability of 0.8.

Experiment I: Homogeneous Peers with Similar Traffic Demands. In this experiment, we have a network of 50 peers and one ISP and the parameters are generated as described above. The peers are of similar sizes and have similar traffic demands. Figure 5(a) shows the biddings of peers 1, 2, 3 and 4 throughout the experiment. The vertical axis shows the biddings of the peers and the horizontal axis shows the time. Note that the biddings converge rapidly. Figure 5(b) shows the transmission rates of peer 8. The vertical axis shows the transmission rates and the horizontal axis shows the time. Again, we observe that the traffic rates converge to the equilibrium quickly.



(a) (b)

Fig. 5. Exp. I: (a) biddings of peers 1, 2, 3 and 4, (b) samples of transmission rates of peer 8

4 Sensitivity to System Parameters

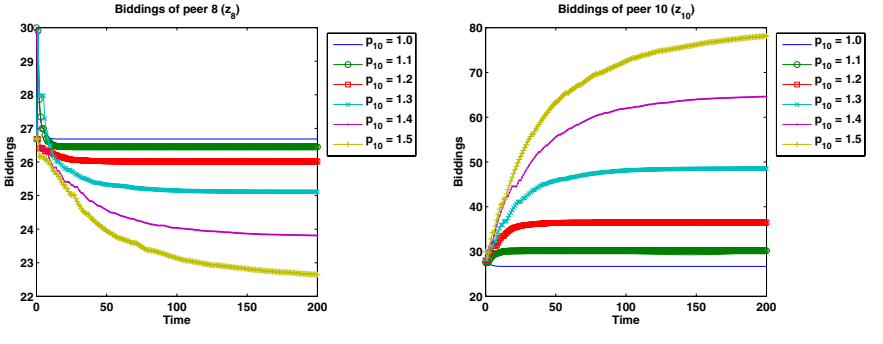
In this section, we investigate in the sensitivity of the equilibrium point as we vary some of the system parameters. The observation is made to the variation of the transmission rates and the biddings from peers. We have three experiments, each corresponds to only one variation in the system parameters.

- Experiment A: variation in the unit price of the private link (p_{ij})
- Experiment B: variation in the traffic demand (x_{ij})
- Experiment C: variation in the unit price of the ISP link $(\mathcal{P})$

The environment of the experiments is constructed as follows. We have a network of 10 peers and one ISP. Each peer has an aggregate link connecting to the ISP and 9 private links connecting to the other peers with capacity $r_{ij} = 10$ and unit price $p_{ij} = 1.0$. All peers have the same traffic demands (i.e., $x_{ij} = 10$ for all i, j). The ISP provides a bandwidth of $\mathcal{R} = 300$ units with unit price $\mathcal{P} = 1.2$. After every period of one second, the ISP applies the resource allocation algorithm, and sends the distribution $\mathcal{R}_i$ to every peer i. Upon receiving the signal from the ISP, peer i performs the rate distribution algorithm with the parameters $\mathcal{R}_i$ and $\mathcal{P}_i = \mathcal{P}$. To introduce noise in the information exchange, a bidding is successfully transmitted to the ISP with a rate of 80%.

In the following experiments, we only change the parameters of peer 10 while keeping the parameters of the other peers unchanged. Our observation is made to the changes of traffic rate distributions and biddings of peer 10 and other peers (eg. peer 8).

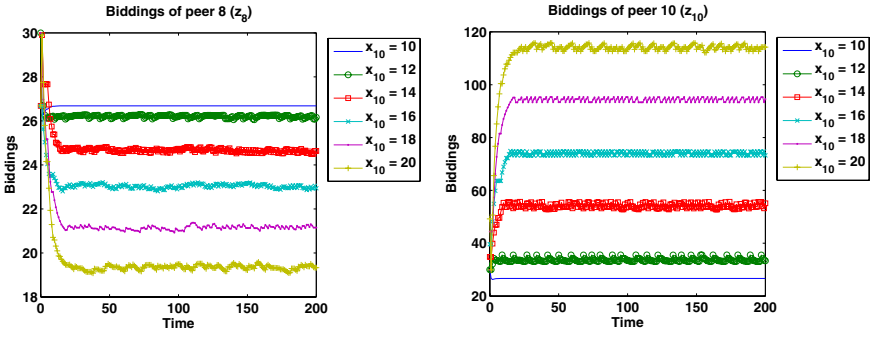
Experiment A: Change of Unit Price in Private Link (p_{ij}). In here, we show how the change in the unit price of private link affects the biddings of all peers. The parameters are constructed as described above. We vary the unit prices of the private links of peer 10 from 1.0 to 1.5, and investigate in the effects in the biddings of peers. Figure 6 illustrates the biddings in the ISP link bandwidth of peers 8 and 10. The vertical axis shows the biddings and the horizontal axis shows the time. When the unit price of the private links of peer 10 increases, peer 10 shifts its traffic from the private links to the ISP link in order to reduce the payment. So peer 10 offers a larger bid. This increases the congestion cost in the ISP link. Other peers (e.g., peer 8) detect this increase in congestion cost and so they shift their traffics from the ISP link to their private links, and give smaller bids.



(a) (b)

Fig. 6. Experiment A: (a) peer 8's bidding (z_8), (b) peer 10's bidding (z_{10})

Experiment B: Change of Traffic Demand of a Peer (x_{ij}). In here, we show how the change in the traffic demand affects the biddings of all peers. The parameters are constructed as described above. We increase all the traffic demands of peer 10 from 10 to 20, and investigate the effects in the biddings of peers. Figure 7 illustrates the biddings in the ISP link bandwidth of peers 8 and 10. When peer 10 increases the traffic demands, it has to increase the transmission rates both in the private links and the ISP

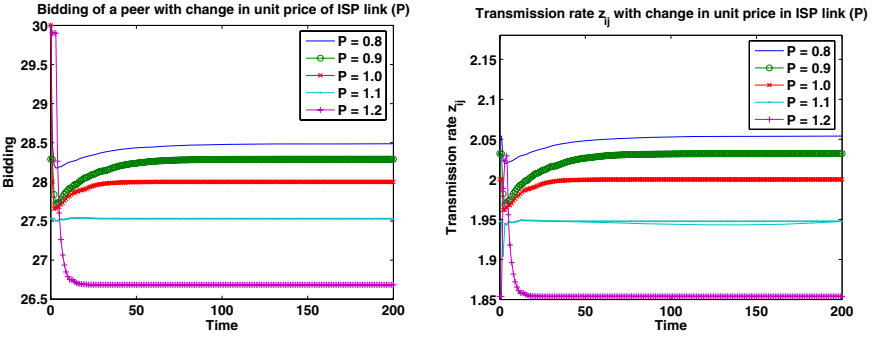


(a) (b)

Fig. 7. Experiment B: (a) peer 8's bidding (z_8), (b) peer 10's bidding (z_{10})

link. So peer 10 gives a larger bid to ISP asking for more bandwidth. This increases the congestion cost in the ISP link. Other peers (eg. peer 8) detect this increase and shift their traffics from the ISP link to their private links, and give smaller bids.

Experiment C: Change of Unit Price in ISP Link ($\mathcal{P}$). In here, all peers have the same traffic demands, and the unit prices and capacities of all private links are identical. The only changing parameter is the unit price of the ISP link. The unit price decreases from 1.2 to 0.8. Figure 8(a) shows the biddings in the ISP link of a peer throughout the experiment. We observe that a peer increases the bidding in the ISP link with decreasing price in the link. Figure 8(b) shows the transmission rate z_{ij} for $i \neq j$ throughout the experiment. We see that a peer increases the transmission rate in the ISP link with decreasing price in ISP link. The increases in both the biddings and transmission rates in ISP link are due to the decrease in payment to the ISP link. As a result, peers shift some of their traffics from the private link to the ISP link. We have two extra experiments in the sensitivity test of (i) private link capacity (r_{ij}) and (ii) ISP link capacity ($\mathcal{R}$). For more details, please refer to our technical report [3].



(a) (b)

Fig. 8. Experiment C: (a) biddings in ISP link (z_i), (b) transmission rates (z_{ij})

5 Conclusion

In this paper, we investigate the interplay between a tier-1 ISP and N tier-2 ISPs (peers). A peer has a connection to the ISP, and possibly connected to other peers with some private links. Each peer needs to determine the appropriate amount of traffic via the ISP's link and the private links so as to minimize its cost. The ISP, on the other hand, needs to perform proper resource allocation to distribute its resource properly. We show the necessary and boundary conditions for the transmission rate vectors of a peer to obtain the minimum cost. We present an algorithm for the ISP to do the resource allocation. We then show the optimal rates and biddings of peers converge with the resource allocation algorithm of ISP even when the number of peers is large. Finally, we show and explain how the change in a single parameter can affect the optimal rates and biddings of all peers and that peers can adapt to these changes and quickly converge to an equilibrium solution. The complicated issues of multiple ISPs and multihoming will appear in our future work.

Acknowledgement. This work is supported in part by the RGC grant.

References

1. T. Basar and R. Srikant. Revenue-maximizing pricing and capacity expansion in a many-user regime. In *Proceedings of the IEEE Infocom*, New York, June 2002.
2. F. Kelly. Charging and rate control for elastic traffic. In *European Transactions on Telecommunications*, volume 8, 1997.
3. S. C. M. Lee, J. W. J. Jiang, and J. C. S. Lui. Performance modeling on the interaction of isps technical report.
4. P. Marbach. Pricing differentiated services networks: Bursty traffic. In *Proceedings of the IEEE Infocom*, Alaska, March 2001.
5. P. Marbach. Priority service and max-min fairness. In *Proceedings of the IEEE Infocom*, New York, March 2002.
6. A. M. Odlyzko. The economics of the internet: Utility, utilization, pricing, and quality of service. In *AT&T Research Lab, Tech Report*, 1998.
7. M. J. Osborne and A. Rubinstein. *A Course in Game Theory*. The MIT Press, 1957.
8. S. Shenker, D. Clark, D. Estrin, and S. Herzog. Pricing in computer networks: Reshaping the research agenda. In *ACM Computer Communication Review, Vol 26*, 1996.

A Random Walk Model for Studying Allocation Patterns in Auction-Based Resource Allocation*

Manos Dramitinos, George D. Stamoulis, and Costas Courcoubetis

Network Economics and Services Group (N.E.S.),
Department of Informatics, Athens University of Economics and Business
76 Patision Str. Athens, GR 10434, Greece
{mdramit, gstamoul, courcou}@aueb.gr

Abstract. We consider users bidding in a series of multi-unit sealed-bid auctions, aiming at reserving the same amount of units of the resource auctioned, e.g. transmission slots in a wireless network. Each user attains from each successful allocation of resource units an instant marginal utility that depends on his history of resource allocation. The user's bid at each auction equals this marginal utility. We introduce a random walk model for transient analysis of this series of auctions, we study the properties of the resulting user resource allocation patterns and we provide a numerical and experimental evaluation of this model.

Keywords: Auctions, random walk, resource allocation.

1 Introduction

In this paper, we consider users who are participating in a series of consecutive sealed-bid multi-unit auctions, aiming at reserving the same amount of units of the resource (or, in general, of the good) auctioned. Each user attains from each successful allocation of resource units an instant marginal (i.e. additional) utility that depends on the resource allocation pattern. We assume that the user's bid at each auction equals this marginal utility. Hence, the auction's price fluctuations and the users' utility functions greatly affect both the bids submitted and the resulting resource allocation patterns. This is a problem of practical importance in communication networks. A prominent example is bandwidth allocation in UMTS (and other, e.g. GPRS) networks. Indeed, in [1], we study the problem of resource reservation in UMTS networks in which users request services other than telephony that last for long time intervals. Each of these sessions has a fixed target QoS level, which for simplicity we assume that corresponds to a certain bit-rate. The duration of network time-slots over which resource units are allocated is much shorter. We define in [1] an auction-based mechanism achieving nearly consistent reservation of the resources of a UMTS network by the users that

* This research was partly supported by EU-IST-2003-507607 B-BONE and IST-NoE EuroNGI. The authors wish to thank B-BONE and EuroNGI partners for useful discussions on the subject of this paper.

B. Stiller, P. Reichl, and B. Tuffin (Eds.): ICQT 2006, LNCS 4033, pp. 25–36, 2006.

value them the most, in order to satisfy the longer time scale QoS requirements of their service sessions. Thus, due to our mechanism, these users receive service of very good quality at a charge determined by the market. The non-competitive users receive service of very inferior quality for a very short time period at a very low charge. Therefore, our resource reservation mechanism serves a soft CAC. The mechanism is based on a series of Generalized Vickrey Auctions and a set of predefined user utility functions that we introduce [1]. Bidding is performed automatically on behalf of the users on the basis of each user's selection of one of these utility functions and his declaration of a total willingness to pay, and is dependent of the user's history of resource allocation. This approach has also been adopted by other researchers in the field [2] as well as also in the EU-finded IST project B-BONE [4]. In this paper we present a random walk model for transient analysis of this mechanism. Based on this model, we study the impact of the parameters of one of these utility functions in the resource allocation patterns that users attain. We also provide a numerical and experimental evaluation of the model, setting its parameters acccording to the input from project B-BONE. Finally, we explain how it can help users select parameters of their utility/bidding functions. The analysis of the properties of auctions in dynamic environments by means of mathematical tools has also received attention in the literature [3].

2 The Auction Mechanism

The problem of UMTS resource allocation to sessions with QoS requirements is very complicated. Indeed, users demand sessions spanning partly overlapping intervals with different durations, which in general are much larger than the time scale t_a of the network frames in which resources are allocated. The approach that we introduced in [1] is to conduct a sequence of auctions, each concerning reservation of bits within one UTRAN frame. Each auction is a sealed-bid Generalized Vickrey Auction (GVA), with bids of the type (p, q), where q is the quantity of units (bits) sought in the present frame and p is the price proposed for each such unit. These bids are essentially *atomic*, i.e. a winning bid results in the allocation of all the resource units demanded, except for the *cut-off* bidder.

In a realistic case of a UMTS network, it would not be feasible for users to participate in all these auctions. Thus, since the user cannot place his bids on a per auction basis, we define utility functions pertaining to the various services. These functions are provided by the network operator as bidding functions for the user to choose from; they are scaled by the user's i total willingness to pay $U_{s,i}$ for the service s, which is given by the user himself as part of his service request. Then, the network runs the various auctions by bidding on *behalf* of each user. We assume that the user's i utility $u_{s,i}$ from obtaining the service s is the sum of the marginal utilities attained due to each successful allocation; thus, $u_{s,i}(x_i^{(1)}, ..., x_i^{(K_{s,i})}) = \sum_{t=1}^{K_{s,i}} v_{s,i}^{(t)}(x_i^{(1)}, ..., x_i^{(t)})$, where $K_{s,i}$ is the number of auctions where user i participates during his service session. Also, for every user the *bid placed by the network at each auction equals the marginal utility* to be attained if his bid is a winning one in the auction. This choice is motivated by the

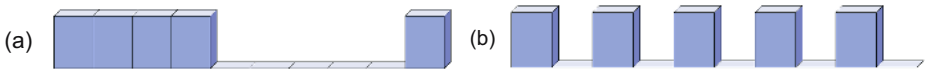


Fig. 1. Inconsistent resource allocation patterns

incentive compatibility property of the Generalized Vickrey Auction, whereby sincere bidding is a dominant strategy; see [1].

In this paper, we restrict attention to one of these predefined utility functions, namely that suitable for users sensitive to service continuity, such as audio and video streaming. Thus, these users prefer the allocation pattern of Fig. 1(a) to that of Fig. 1(b). In order to express this preference, we define the sub-utility function to be $v_{s,i}^{(t)}(x_i^{(1)}, ..., x_i^{(t)}) = \mathbf{1}(x_i^{(t)} = m_i)\frac{U_{s,i}}{K_{s,i}} \cdot \alpha^{d_i}$, where: $\alpha \in (0, 1]$ is a discount factor; d_i is the distance between the current and the previous slots during which user i achieved reservations; $\mathbf{1}(\cdot)$ denotes the indicator function, which is justified by the fact that bids are atomic. Therefore, *history* of previous allocations influences $v_{s,i}^{(t)}$ through the value of d_i. Hence, when the user fails to reserve resources at some auction, his marginal utility decreases exponentially, in order to express the user's displeasure from the existence of gaps in his resource allocation pattern. On the contrary, as long as a user is allocated resources, his marginal utility equals $\frac{U_{s,i}}{K_{s,i}}$, which is henceforth denoted as u_0. The utility function considered is suitable for the UMTS Streaming Class which is destined to serve streaming audio and video sessions; m_i pertains to the Maximum Bit-rate parameter of this class.

3 The Random Walk Model

We consider a user who is participating in a series of consecutive sealed auctions (see Sect. 2), aiming at reserving the same amount of units of the resource auctioned (e.g. bits in a UTRAN frame). Recalling how bidding is performed, it follows that the auction's price fluctuations and the parameters u_0, α of the users' utility function greatly affect both the bid submitted and the resulting resource allocation pattern. In this section, we present a random walk model descriptive of the price fluctuation. Employing this model, a user can examine the impact of the value of parameter α of his bidding function on his average gap length, while "ignoring" the actual auction competition (of which he is unaware of), which is now simulated by means of the model. In particular, the fluctuation of the auctions' *cutoff price* over time is modeled as a random walk [5]. That is,

$$p_t = \begin{cases} p_{t-1} + \delta, \text{ with probability } q \\ p_{t-1} - \delta, \text{ with probability } 1 - q \end{cases} \quad \text{for } t = 1, 2, ... \tag{1}$$

Henceforth, p_t is simply referred to as the auction price. Furthermore, according to the definition of the utility function considered (see Sect. 2) the fluctuation of the user's marginal utility is modeled as follows:

$$u_t = \begin{cases} u_0, & \text{if } u_{t-1} \geq p_{t-1} \\ \alpha \cdot u_{t-1}, & \text{if } u_{t-1} < p_{t-1} \end{cases} \quad \text{for } t = 1, 2, ..., \tag{2}$$

where $\alpha \in (0, 1]$ is the discount factor. At time 0 the auction price raises for the first time above user's bid (i.e. $p_0 > u_0$) and fluctuates according to the random walk model thereafter.

Next, we discuss the appropriateness of our model. First, note that unlike with the actual auction, under our model, the price is not affected by the bids of the user considered. This is a good approximation of the actual auction because in an actual network many bidders will be participating. Furthermore, it could be argued that a random walk is not descriptive of the actual auction, since it is expected that when the mechanism is employed in an actual network the price will fluctuate around a the long term average value. However, we do not perform stationary analysis of the auction price in this paper. Instead, the model introduced is appropriate for *transient* analysis of the price dynamics of the auction. Consider, for instance, a case where a user is competitive in general, because his marginal utility is higher than the average auction price; however, his bid is instantly topped by the auction price due to some abrupt increase in demand (e.g. due to new users' arrival). Price evolution in this case can be emulated as a random walk process with positive drift. This actually leads to a conservative analysis, since in the actual auction the price would have the tendency to return more rapidly to its long-term average when it has deviated considerably from it. It should also be noted that transient analysis is very important, because due to the exponentially decreasing marginal utility, the user in the case described above becomes gradually less competitive, his service is interrupted and he may as well decide to drop out.

4 Analysis of the General Model

In this section, we study the probability of user's "re-entrance", i.e. the probability that the user will succeed in being awarded again resources in future auctions, as well as the resulting resource allocation patterns. First, we compute the number of continuous price decrements k that are required until the user's marginal utility exceeds again the (constantly decreasing) auction price. We define the distance of the auction's price minus the user's exponentially decreasing valuation as $dist_t = p_t - u_t$. The parameter k sought is the minimum integer t that satisfies the inequality $dist_t \geq 0$, which implies that

$$k = \min_t \{p_0 - t \cdot \delta - \alpha^t \cdot u_0 \geq 0\}. \tag{3}$$

Thus, solution of this is the minimum possible "time" to elapse until the user's re-entrance. Assuming that after $k + j + i$ auctions, with j price increases and $k + i$ price decreases, the user has still not succeeded in winning in the auction, the distance $dist_{k+i+j}$ is then:

$$dist_{k+i+j} = p_{k+i+j} - \alpha^{k+i+j} \cdot u_0 = p_0 + (j - i - k) \cdot \delta - \alpha^{k+i+j} \cdot u_0.$$

We proceed to compute the probability of user's re-entrance. We define:

$$i_j = \min_i \{dist_{k+i+j} \leq 0\} \tag{4}$$

Since we have already computed k, using (4) we can compute for each number of price increments j, the minimum number of price decrements $i_j + k$ that are required in order for the user to have a winning bid in the auction. Thus, we can compute the feasible values of the "gap size" of the user's resource allocation pattern, which equals $j+i_j+k$, for some j. For example, assume that for certain u_0, δ, and α, we have: $k = 1$ and $i_1 = 1$, $i_2 = 2$ and $i_3 = 3$. Since $k = 1$, if the price drops at $t = 1$, then the user's bid will be a winning one again in that auction. However, if the price increases at $t = 1$ and then keeps decreasing (i.e. $j = 1$), then $2 = k + i_1$ price drops suffice for the user's marginal utility to top the auction's price. If this does not happen, because $j > 1$, then the soonest possible time for user's is $t = 5$ if $j = 2$ etc. Figure 2(b) depicts some possible auction's price fluctuations for which the user re-enters the network after 7 time units.

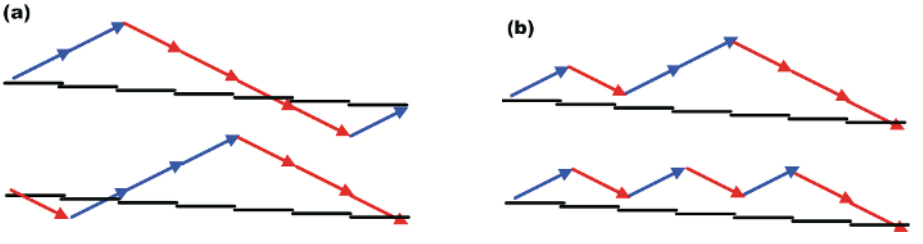


Fig. 2. The price fluctuates and user's marginal utility and bid are reduced

Next, we compute the probability that user's exponentially decreasing marginal utility exceeds the auction's price for the *first time* at time $t = k+j+i_j$ for some j. In order to compute $\Pr_{\text{hit}}(t)$, we must first compute the number $A(j,k)$ of "price fluctuation patterns" that lead to the user's re-entrance *for the first time* at time $t = j + i_j + k$. This number is obviously less than the total number of patterns of length $j+i_j+k$, namely $\binom{j+i_j+k}{j}$. Indeed, Fig. 2(a) depicts two possible ways of price fluctuation, with $j = 3$, that also lead to the user's re-entrance. However, for these patterns this does not happen for the *first time* after *exactly* 7 steps, thus these patterns should not be computed in $A(3,1)$. (Also, for these patterns, the user's marginal utility would not continue to decrease exponentially until $t = 7$, as depicted in Fig. 2(a); it would have become u_0 once the user achieves to re-enter.) Hence, in order to compute $A(j,k)$, we must exclude all those patterns that result in the user's marginal utility to exceed the auction price at some time t prior to $j + i_j + k$. In particular, we must exclude all the sub-patterns having size $j' + i_{j'} + k$, for all $j' < j$. Each such sub-pattern must be excluded $\binom{j+i_j-j'-i_{j'}}{j-j'}$ times; this is the number of possible

allocations from time $j' + i_{j'} + k + 1$ to $j + i_j + k$ that are combined with the original sub-pattern. Hence, the number of acceptable price fluctuation patterns, $A(j,k)$ is:

$$A(0,k) = 1$$
$$A(j,k) = \binom{j+i_j+k}{j} - \sum_{j'=0}^{j-1} A(j',k) \cdot \binom{j+i_j-j'-i_{j'}}{j-j'}, \text{ for } j = 1,2,.... \quad (5)$$

As already explained, it is feasible for the user's marginal utility to exceed the auction's price only at certain times, namely at times $j + i_j + k$, where i_j is derived from (4). Hence the probability $\Pr_{\text{hit}}(t)$ that the user's marginal utility (and bid in the auction) exceeds the auction price at some time t *for the first time*, is:

$$\Pr_{\text{hit}}(t) = \begin{cases} 0, \text{if } \nexists\ j \text{ s.t. } j + i_j + k = t \\ A(j,k) \cdot (1-q)^{k+i_j} \cdot q^j, \text{ otherwise.} \end{cases} \quad (6)$$

The cumulative probability that the user's marginal utility will have exceeded the auction price up to some time t, denoted as $\Pr_{\text{cHit}}(t)$ is $\Pr_{\text{cHit}}(t) = \sum_{t'=1}^{t} \Pr_{\text{hit}}(t')$. We also denote as $\Pr^*_{\text{cHit}}$ the asymptotic value of $\Pr_{\text{cHit}}(t)$, as $t \to \infty$, which equals the probability that the user will ever achieve re-entrance.

5 A Special Case

Assume that $\alpha = 1$, or equivalently that $u_t = u_0$ for $t = 1, 2, ...$ This implies that $i_j = j$. This assumption results in a model that is much easier to analyze. This model will also provide a bound for $\Pr^*_{\text{cHit}}$ of the general model, which is revisited in Sect. 6.

Proposition 1. *There holds*

$$\lim_{t\to\infty} \Pr_{\text{cHit}}(t) = \Pr^*_{\text{cHit}} = \begin{cases} 1, & \text{if } 0 < q \le \frac{1}{2} \\ \left(\frac{1-q}{q}\right)^k, & \text{if } \frac{1}{2} < q < 1\ . \end{cases} \quad (7)$$

Proof. The cumulative probability is $\Pr^*_{\text{cHit}} = \sum_{t'=0}^{\infty} \Pr_{\text{hit}}(t')$. Using (6) we obtain:

$$\Pr^*_{\text{cHit}} = \sum_{j=0}^{\infty} A(j,k) \cdot (1-q)^{(k+j)} \cdot q^j \quad (8)$$

First we consider the case where $0 < q < \frac{1}{2}$. In this case, the price has a negative drift. Hence, it is certain that at some point it will drop below the user's marginal utility $u_t = u_0$; see [5]. Thus, the respective cumulative probability is 1. Next, we consider the case where $\frac{1}{2} < q < 1$. We have:

$$\Pr^*_{\text{cHit}} = \sum_{j=0}^{\infty} A(j,k) \cdot (1-q)^{(k+j)} \cdot q^j = \left(\frac{1-q}{q}\right)^k \cdot \sum_{j=0}^{\infty} A(j,k) \cdot q^{(j+k)} \cdot (1-q)^j \quad (9)$$

The sum in the right hand side of (9) is identical to that in (8) except that q and $1-q$ are swapped. Note that in (8), if $0 \leq q < \frac{1}{2}$ the sum converges to 1. On the contrary, in (9) we have that $0 \leq 1-q < \frac{1}{2}$. Therefore, the sum in (9) converges to 1, which implies that $\Pr^*_{\text{cHit}} = \left(\frac{1-q}{q}\right)^k$, if $\frac{1}{2} < q < 1$. Finally, by continuity for $q \to \frac{1}{2}$, $\Pr^*_{\text{cHit}} = 1$ too. □

The same result for $k = 1$ and $q \in (\frac{1}{2}, 1)$ is also established differently in [5]. However, the above proposition was proved for any value of k, because it will be used in the next section. The aforementioned limits are useful in order to understand the user resource allocation patterns that result when users participate in a series of auctions. When the probability of price increase is lower than $\frac{1}{2}$, it is certain that the user will eventually succeed in topping the auction's price and reserve network resources again. This is obviously the case for the actual network (where the cut-off price is determined by competing users) at periods of low to medium demand. Of course, the lower the probability of price increase, the sooner the user will be re-allocated network resources in the auction. On the contrary, when the probability that the price decreases is higher than the probability that the auction's price will increase - that is, whenever the competition is high - it is uncertain if the user will eventually manage to receive service again. The probability that this happens is $\left(\frac{1-q}{q}\right)^k$, which is a decreasing function of q. The higher the probability of price increase, that is the more intense the competition in the auction, the less probable it is for the user to receive service again. This is justified since if the user's bid is topped at a very competitive auction, it becomes very hard for this user's subsequent bids to become winning again in the series of auctions that follow, where the cut-off prices tend to increase. Exponential reduction of the user's bids (i.e. $\alpha < 1$) may only make matters worse, i.e. result in a smaller value of $\Pr^*_{\text{cHit}}$. Notice also, that for $\alpha = 1$, $\Pr^*_{\text{cHit}}$ depends on q as well as k, which due to (4) equals $k = \lceil (p_t - u_0)/\delta \rceil$. It should also be noted that although the result of Proposition 1 is asymptotic for $t \to \infty$, the time horizon is not actually infinite in the sense that the probability of reentrance can be well approximated by taking the first few terms of the series in (8). Thus, the result is still applicable for the purpose of analyzing transient phenomena. The same comment also applies for the results of Sect. 6 and 7.

6 Revisiting the General Model

Having studied the special case of Sect. 5, we return to the general model presented in Sect. 3 with $\alpha < 1$, the analysis of which is more complicated, as already mentioned. The difficulty of providing a closed-form equation for $\Pr^*_{\text{cHit}}$ (i.e. the asymptotic value of $\Pr_{\text{cHit}}(\text{t})$ for $t \to \infty$) is due to the fact that neither i_j nor $A(j,k)$ are known in closed form. However, it is possible to provide some bounds for these probabilities. Under certain assumptions these bounds are tight.

Clearly, whenever the probability q that the price goes up in the random walk is less than $\frac{1}{2}$, then the cumulative probability converges to 1, i.e. $\Pr^*_{\text{cHit}} = 1$, $0 \leq$

$q \leq \frac{1}{2}$. This can be proven by applying the same arguments with these provided in the proof of Proposition 1. For $q > \frac{1}{2}$, the computation of $\Pr^*_{\text{cHit}}$ is very complicated. Below, we provide an upper and lower bound for this probability.

Proposition 2. *We have*

$$(1-q)^k + k \cdot \frac{(1-q)^{k+i_1}}{1-q\cdot(1-q)^{i_1}} \leq \Pr^*_{\text{cHit}} < \left(\frac{1-q}{q}\right)^k, \quad \textit{if} \quad \frac{1}{2} < q < 1. \tag{10}$$

Proof. First, we prove that $\Pr^*_{\text{cHit}} < \left(\frac{1-q}{q}\right)^k$, if $\frac{1}{2} < q < 1$. This is easily proven since $\left(\frac{1-q}{q}\right)^k$ is the value of $\Pr^*_{\text{cHit}}$ if $\alpha = 1$, i.e. if $u_t = u_0$ for $t = 1, 2, ...$ In the general model, we have $u_t \leq u_0$. Clearly, by monotonicity, under the exponentially decreasing marginal utility (and bid) the user can never be better off compared to the case where $u_t = u_0$ for $t = 1, 2, ...$ Thus, $\Pr^*_{\text{cHit}} < \left(\frac{1-q}{q}\right)^k$, if $\frac{1}{2} < q < 1$.

Next, we prove that $(1-q)^k + k \cdot \frac{(1-q)^{k+i_1}}{1-q\cdot(1-q)^{i_1}} \leq \Pr^*_{\text{cHit}}$. We have

$$\begin{aligned}\Pr^*_{\text{cHit}} = (1-q)^k &+ A(1,k)\cdot q\cdot(1-q)^{k+i_1} + A(2,k)\cdot q^2\cdot(1-q)^{k+i_2} \\ &+ A(3,k)\cdot q^3\cdot(1-q)^{k+i_3} + ...\end{aligned} \tag{11}$$

Due to the exponential decrease of the marginal utility, we have $u_j - u_{j-1} = \alpha\cdot(u_{j-1} - u_{j-2})$, which easily implies that the difference $\Delta i_j = i_j - i_{j-1}$ is non-increasing in j. That is, the number Δi_j of extra price decrements required to cope with one additional price increment is non-increasing in j, although i_j itself is increasing in j. Thus, $\max\{\Delta i_j\} = \Delta i_1 = i_1$. Furthermore, $\Delta i_2 = i_2 - i_1$. Therefore, we obtain that $i_2 \leq 2\cdot i_1$. Similarly, $\Delta i_3 = i_3 - i_2 \leq i_2 - i_1 \leq i_1$, which implies that $i_3 \leq i_2 + i_1 \leq 3\cdot i_1$. Continuing this argument, it follows easily that in general:

$$i_j \leq j\cdot i_1 \qquad \text{for } j = 2, 3, ... \tag{12}$$

Applying (5), we have $A(1,k) = k$. It is also easily seen that $A(j,k) \geq A(j-1,k)$ for $j = 2, 3, ...$ The proof is omitted for brevity reasons. Therefore, $A(j,k) \geq k$ for $j = 1, 2, ...$ Combining this bound for $A(j,k)$ and that for i_j as given by (12), we obtain from (11):

$$\Pr^*_{\text{cHit}} \geq (1-q)^k + k\cdot(1-q)^{k+i_1}\cdot[1 + q\cdot(1-q)^{i_1} + q^2\cdot(1-q)^{2\cdot i_1} + ...] \Rightarrow$$

$$\Pr^*_{\text{cHit}} \geq (1-q)^k + k\cdot\frac{(1-q)^{k+i_1}}{1-q\cdot(1-q)^{i_1}} \quad \{\text{since } q\cdot(1-q)^{i_1} < 1\}$$

□

We have so far provided an upper and a lower bound for $\Pr^*_{\text{cHit}}$ for $\frac{1}{2} < q < 1$. We proceed to investigate the distance between these bounds, which expresses their accuracy as an approximation of $\Pr^*_{\text{cHit}}$. Notice that the value of the distance

depends also on i_1, which can be computed from (4) employing the values of p_0, u_0 and δ. Recall that we have implicitly assumed in our analysis that the user considered is competitive, which implies that p_0 is not much larger than u_0. Users generally select large values of α (i.e. close to 1), in order to avoid very fast decaying of their marginal utility. Therefore, in order to simplify our study on the bounds on $\Pr^*_{\text{cHit}}$, we restrict attention to small values of k only, which are the ones of practical importance, and we approximate i_1 with k. (Notice that, for $\alpha = 1$ and $p_0 = u_0$, we have $i_1 = k$.) As depicted in Table 1, the difference of the two bounds is very small if $q \geq 0.6$. This distance increases significantly for $q \in (0.5, 0.6)$ and is unacceptable for $k = 2$ and $k = 3$. Additional numerical results show that the accuracy improves again for large k.

Table 1. The distance of the bounds of $\Pr^*_{\text{cHit}}$ as a function of k and q

	Difference of bounds of $\Pr^*_{\text{cHit}}$								
q:	0.51	0.53	0.55	0.57	0.60	0.65	0.70	0.75	0.80
$k=1$	0.15	0.12	0.09	0.08	0.06	0.03	0.01	0.01	0.002
$k=2$	0.55	0.45	0.37	0.31	0.23	0.13	0.08	0.04	0.02
$k=3$	0.72	0.56	0.43	0.33	0.22	0.11	0.05	0.02	0.01

7 Studying the Resulting User Resource Allocation Patterns

Recall that for the user considered, his marginal utility is topped by the auction price at $t = 0$. If the marginal utility exceeds again the auction price *for the first time* at time t, then the gap length is defined as t. The computation of the average length can be done numerically as follows: First, we compute k, and i_j for $j = 1, 2, \ldots$. Then, we compute numerically $\Pr_{\text{hit}}(t)$ from (6). In order to compute the average gap length, we must distinguish two cases:

- $0 < q \leq \frac{1}{2}$: In this case, since the probabilities $\Pr_{\text{hit}}(t)$ sum to 1, it suffices to add the products of all possible gap lengths $j + i_j + k$ with the respective hitting probability $\Pr_{\text{hit}}(j + i_j + k)$.
- $\frac{1}{2} < q < 1$: In this case, since the sum of probabilities $\Pr_{\text{hit}}(t)$ do not sum to 1, we add the products of all possible gap lengths $j + i_j + k$ with the respective hitting probability $\Pr_{\text{hit}}(j + i_j + k)$; then we normalize, and by dividing this sum with the sum of aforementioned probabilities, that is with the limit $\Pr^*_{\text{cHit}}$. Therefore, in this case we compute the conditional average gap length, given that the user does re-enter.

Therefore, the average gap length AGL is given by:

$$AGL = \begin{cases} \sum_{j=1}^{\infty} \Pr_{\text{hit}}(j + i_j + k) \cdot (j + i_j + k), & \text{if } 0 < q \leq \frac{1}{2} \\ \frac{1}{\Pr^*_{\text{cHit}}} \cdot \sum_{j=1}^{\infty} \Pr_{\text{hit}}(j + i_j + k) \cdot (j + i_j + k), & \text{if } \frac{1}{2} < q < 1 \ . \end{cases}$$

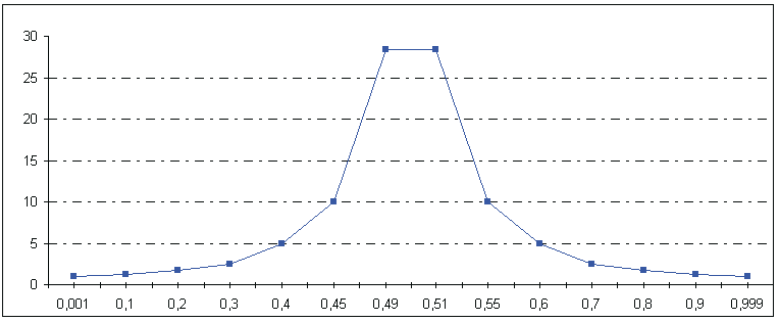


Fig. 3. The average gap length as a function of q for $u_0 = 100$, $\delta = 10$, $\alpha = 1$

Numerical results are depicted in Fig. 3 for $\alpha = 1$. The values of the average gap length are symmetric with respect to the axis $q = \frac{1}{2}$. Note that for values of q that are very small the user is most likely to top the auction's price immediately (after 1 price decrease at the next auction), hence resulting in a value of the average gap length that is very close to 1. Also, for values of q close to 1 the same values of AVG apply, since in this case the user's bid will either become winning again immediately or this will never happen.

The simulation results reported in [1], indicate that the vast majority of user resource allocation patterns are either perfectly consistent (high-value users) or comprise very few resource allocations (non-competitive users). For users whose value is often close to the auctions' cutoff prices, it has been observed that their respective resource allocation patterns are typically in accordance with that depicted in Fig. 1 (a), having few, large gaps; the less preferable patterns with frequent small-sized gaps (like those depicted in Fig. 1 (b)) are rare as required. These phenomena are captured by our model. Indeed, users whose bids are often close to p_t can be viewed as participating in an auction with $q \simeq \frac{1}{2}$ and thus the gaps of their resource allocation patterns are expected to be large.

8 Validation and Usefulness of the Model

The analysis of the paper was based on the assumption that it suffices to model the auction's cut-off price by means of a random walk model. In this section, we provide an experimental evaluation of the proximity of the estimate of the model of Sect. 3 regarding the average gap length to that obtained by auction simulations.

The methodology we adopt is the following: We run various auction simulations where a large population of competing users bid for resources, (see [1]). After each auction, we store the users who have gaps in their resource allocation patterns, their respective u_0 and the gap length. Then, we define an equivalent random walk model: We estimate the probability q of price increase in the equivalent random walk by dividing the number of price increments of the

actual auction with the total number of price fluctuations. We also estimate the step δ by averaging the actual price differences $|p_t - p_{t-1}|$ throughout the actual auction. After running a large number of simulations of the auction, we group together those users that have gaps[1] with similar u_0 and competed in auctions that can be described with similar δ, q. For each such group, we compute the mean actual gap lengths obtained in the experiments; this is then compared to the average gap length derived by means of the equivalent random walk model using Mathematica. Such comparisons reveal that in most cases our model provides a very good approximation of the actual gap length arising in the simulated auction. A typical example is provided in Table 2: in three simulation runs a total of 10 gaps for users with $u_0 \simeq 272$, $q \simeq 0.61$, $\delta \simeq 16$ were recorded. Their mean gap length was 3.3636 while our random walk model gives for the same u_0, q and δ an average gap length of 3.4883.

Table 2. Typical experimental results demonstrating the accuracy of our model

Auction	Auction simulation results ($\alpha = 0.99$) (u_0, Gap size)	Estimated R. Walk q	δ
1	(274, 2), (274, 1), (269, 2)	0.62	15.9
2	(262, 5), (262, 3)	0.59	15.9
3	(276, 7), (276, 3), (276, 2), (278, 7), (278, 2), (278, 3)	0.62	16.1
Equivalent random walk model: $u_0 = 272$, $q = 0.61$, $\delta = 16$			

Next, we explain how the model can be employed in order to support users. Recall that users select a utility function on the basis of which the network bids on behalf of the users; see Sect. 3 and [1]. Note that the user's selection of α is of particular importance for the QoS attained. The higher α the higher the probability of user's re-entry and the lower the average gap length experienced, but also the higher the charge. Therefore, budget-constrained users whose u_0 is often close to the auction cut-off price end up with a small net benefit. Therefore, it might be more profitable for the bids of such users to be losing in some auctions, as long as the resulting gaps are of acceptable sizes. Hence, by selecting an appropriate value of α, a user can affect both the QoS experienced and the expected net benefit to be attained from the auction. The best choice of course depends on the user's sensitivity with respect to the QoS versus the respective charge. Clearly, the user should employ in this selection process a proxy of the expected QoS level. The average gap length computed by means of the random walk model can definitely be used, and as already explained it constitutes an accurate estimate. To better illustrate these ideas we present certain numerical results in Table 3, which shows the sensitivity of the average gap length with respect to $\alpha \in [0.960, 0.999]$ for the equivalent random walk model of Table 2. We assume that we are in the context of a UMTS network where a new auction is run every 10 msec [1]. Clearly, QoS-sensitive users should opt for a value of α in $[0.99, 0.999]$, while those more interested in attaining a somewhat lower

[1] Note that a user may experience multiple gaps in his resource allocation pattern.

charge than receiving almost perfect QoS should select α in $[0.965, 0.990]$. On the contrary, any selection of $\alpha < 0.96$ results in an average gap length of more that 500 msec, and thus it should be avoided.

Table 3. Sensitivity analysis depicting the relation of the value of α involved in the user's utility/bidding function and the respective expected average gap length

Average Gap Length sensitivity analysis (model parameters: $u_0 = 278$, $q = 0.355$, $\delta = 10.9$)										
α:	0.955	0.960	0.965	0.970	0.975	0.980	0.985	0.990	0.995	0.999
AGL:	53.06	51.85	30.89	27.78	24.08	18.37	12.57	7.08	4.34	3.47

9 Conclusions

In this paper, we have analyzed the mechanism of [1], where users participate in a series of consecutive sealed-bid auctions, aiming at reserving the same amount of units of the resource auctioned. We have introduced a random walk to model the fluctuations of the auctions' cut-off price. Using this model, we have studied the resource allocation patterns of a user whose utility/bidding function remains constant as long as the user's bids are winning while it is exponentially decreasing in case the cut-off price exceeds the user's bid. Finally, we have provided experimental results validating the model and demonstrated how it can be used in order to support the user in the selection of one of the parameters for his bidding function.

References

1. M. Dramitinos, G. D. Stamoulis, C. Courcoubetis.: Auction-based Resource Reservation in 2.5/3G Networks. Kluwer/ACM Mobile Networks and Applications Special Issue: Mobile and Pervasive Commerce, 9:6, pp. 557-566, December 2004.
2. P. Maillé and B. Tuffin.: An Auction-Based Pricing Scheme for Bandwidth Sharing with History-Dependent Utility Functions. In Proc. of the First International Workshop on Incentive Based Computing, France, September 2005.
3. P. Maillé and B. Tuffin.: The Progressive Second Price Mechanism in a Stochastic Environment. Netnomics: 5:2, pp. 119-147, 2003.
4. FP6-IST-507607 Project B-BONE, http://b-bone.ptinovacao.pt/.
5. S. Ross.: Stochastic Processes. Wiley Series in Probability and Mathematical Statistics, ISBN 0-471-12062-6, 1996.

A Simulation-Based Approach to Bidding Strategies for Network Resources in Competitive Wireless Networks

Fernando Beltrán and Matthias Roggendorf

University of Auckland Business School
7 Symonds Street, Auckland, New Zealand
{f.beltran, m.roggendorf}@auckland.ac.nz

Abstract. We introduce a simulation-based approach to the problem that mobile users may face in a multi-provider environment when seeking to satisfy their demand for bandwidth; if they are allowed to satisfy their individual demands by aggregating shares from two or more providers the problem becomes one of resource allocation in a competitive market. We use the Progressive Second-Price auction at each provider, exploring the properties of three bidding strategies. Simulations aim at learning whether the auction converges at each seller when bidders, either make coordinated or non-coordinated decisions among auctions, or complement already secured shares by bidding at other auctions. Aggregate measures of welfare and sellers' revenue are obtained for each simulation run.

1 Introduction

The introduction of IP for packeting, routing and transportation of digital information in data communication networks has opened up a tremendously broad range of possibilities for the creation of innovative services. Wireless networks are no exception to this trend; traditional cellular telephony providers as well as new entrants are already operating IP-based services in networks of the second (2G) and third (3G) generation. The next generation of wireless networks (NGWN), which currently emerge from cellular network standards and wireless data communications networks, promises to be an all-IP ubiquitous networkçapable of providing multiple service types with guaranteed quality of service [1].

IP-based wireless networks introduce new network management paradigms, especially with reference to resource allocation. When resource allocation is considered, it is convenient to break the problem up in accordance to its particular definition and design on each layer of the Internet protocol. If a whole approach to resource allocation is to be attempted, two functions need to be considered: *subscription* and *access*. Both describe stages in the transaction between provider and users when purchasing services from a network. A network provider hands in a contract to a consumer by which a commercial relation is begun; consumers count as subscribers to the network. When subscribers need to activate their connection, they must get access to network resources - for instance, bandwidth.

B. Stiller, P. Reichl, and B. Tuffin (Eds.): ICQT 2006, LNCS 4033, pp. 37–48, 2006.

One of the most exciting implications of such technological progress is the possible erosion of the subscription paradigm. As new providers step into the market for individual consumers, the flexibility provided by more efficient and adaptive networks will make it possible for consumers to demand access from a network where no previous subscription contract had been signed. Therefore, networks will have to compete for consumers "on-the-spot". The central issue of this paper is the modelling of a new resource allocation scenario implied by NGWN. In such a scenario, two or more wireless operators serving a common service area will see mobile users demand connection to their networks. We assume that a competitive wireless multi-provider setting may well be endowed with a competitive access bidding mechanism. Therefore, any wireless provider herein considered is assumed to solve its resource allocation problem at the access level using an auction.

Pricing schemes consisting of a flat fee provide wrong incentives for resource utilization. Such schemes risk rendering the network inefficient as users, unaware of their impact on the efficient utilization of resources, tend to behave as the exploiters of a common resource with the known consequences of over consumption known as 'the tragedy of the commons' [2]. When a limited resource, such as bandwidth, in an access link is consumed on a flat-fee payment basis, the main concern for the operator is congestion. If the network keeps admitting new connections above a certain level, the consequent degradation of the quality of service will make users turn away. This is especially true in wireless access networks as, despite the development of new technologies, capacity is still of concern. Therefore, a mechanism is needed that will charge an amount that aims to compensate for the effect that any user has on others and, at the same time, provide disincentives for over-utilization.

This paper is organised as follows: in Section 2 the PSP auction is revisited; in Section 3 we formulate the main problem studied here; in Section 4 we introduce three bidding strategies and in Section 5 we present the results of extensive simulation trials. Conclusions and future research are discussed in Section 6.

2 The Progressive Second-Price Auction

The literature on the design of pricing mechanisms for congestion control and charging mechanisms presents an interesting application of the Vickrey auction [3]. When considered as a divisible amount, bandwidth becomes a 'divisible' object to be allocated among agents searching for network resource through a competitive bidding process. Semret [4] introduces the Progressive Second-Price (PSP) auction, an application of the Generalised Vickrey (GV) auction, to allocate divisible objects, in which a bidder submits a quantity and a price to an auctioneer who, in return, will tell the bidder how much of the requested quantity he will get and the overall cost per time unit to be charged.

The Vickrey-Clark-Groves (VCG) mechanism is an incentive-compatible mechanism with additional properties: VCG is efficient (i.e., it maximises social welfare) and individually rational (i.e., it guarantees that any agent joining

the mechanism derives a non-negative utility) [5]. The PSP auction inherits all these properties.

Let us suppose the seller's network has a capacity of Q units. In a PSP auction any user submits information consisting of two values: the desired share of the total resource q_i and the price p_i he is willing to pay for it. The auctioneer allocates a share a_i of the resource to player i at the cost c_i. The allocation rule assigns player i bandwidth a_i equal to the minimum value between his capacity bid, q_i, and the remaining capacity after all those capacity bids, q_k, whose prices beat $i's$ bid ($p_k \geq p_i$) are subtracted from the total capacity Q to be allocated. In other words, the allocation rule is:

$$a_i(s) = q_i \wedge \left[Q - \sum_{p_k \geq p_i, k \neq i} q_k \right]$$

and s represents the set of bids by i, denoted as s_i and by the rest of the players, denoted as s_{-i}. The payment by any agent i is a weighted average of the (unit) prices offered by the other agents; each weight is the incremental capacity from including j in the auction. The pricing rule can be written as:

$$c_i(s) = \sum_{j \neq i} p_j \left[a_j(s_{-i}) - a_j(s_i; s_{-i}) \right]$$

Events such as a new user attempting to join the network or another user leaving trigger the search for a new equilibrium and prompt users to start the submission of new bids. In order to guarantee the convergence of the algorithm a bidding fee ϵ has been introduced to let bidders change their bids only when the gain in net benefit is large enough. This is expressed in [4] as a modified concept of equilibrium known as ϵ-Nash equilibrium. From a technical perspective, the algorithm produces a minimum of signalling overhead since only two values have to be submitted.

Several extensions and modifications of PSP have been proposed. The most prominent one is the multi-bid auction by Maillé and Tuffin [6]. Instead of sending single bids in each auction round a player submits his demand function, which is actually a stepwise, descending price schedule, to the auctioneer. This avoids the convergence phase to reach equilibrium.

3 Procuring Resources in a Multi-provider Setting

We are concerned with the following problem: bidders may participate in two or more auctions occurring simultaneously. At each auction, network capacity or bandwidth is being offered. Each bidder is seeking to win an amount, which he can procure from multiple sources in any combination. We would like to explore what constitutes an optimal bidding strategy for a bidder. Being incentive-compatible, PSP will be considered as the mechanism implemented at each seller.

In the problem herein considered, consumers seek bandwidth to fulfil their needs for communication services. Two or more providers may provide access to the users' mobile terminals through the auctioning of bandwidth. Auctions occur simultaneously. Each user is seeking to win an amount of bandwidth, which he could procure from one or more sources in any combination.

In the case of several multi-unit auctions with users requesting one or more objects, the literature does not provide a solution in which direct mechanisms at each seller elicit truthful (incentive-compatible) bids from the bidders when bidders are allowed to satisfy their demands by adding shares from several sellers. Some progress has been done when studying the problem faced by the bidder when each of two auctioneers has a single unit to auction and both use either a first-price or a second-price auction [7].

Let us assume that a bidder needs a given amount (share) of a divisible good and there are two sellers which can provide the good. The objective of each bidder participating in the market is to maximise the individual utility derived from all auctions. We cannot assume beforehand that each bidder will be motivated to report truthfully to each auctioneer in the marketplace. When a user needs to procure a resource from a divisible resource being auctioned, he faces the problem of finding an adequate bidding strategy. In one line of analysis we must consider the bidder who seeks to source from one provider as opposed to sourcing from several providers. The former situation might, for example, apply to mobile users which are restricted in terms of hand-over or handset capabilities. In the latter, users may be able to bundle resources from several wireless providers. Bundling of resources can, for example, be used by stationary users with adaptive services to increase bandwidth for data transfers or video streaming.

4 Bidding Strategies

To explore possible bidding strategies for both, single-source and multiple source bidding agents, different policies have been defined and implemented in the simulation environment. We restrict our attention to sequential bidding strategies in which agents submit only one non-zero bid to one auction. A exception is the *BidAll* strategy, in which agents submit bids to all auctions simultaneously.

BidAll: With this bidding strategy bidders behave as if they were independently bidding on both auctions. No coordination of submitted bids takes place and agents bid on both auctions. Since bids are not coordinated this strategy is not truthfully revealing an agent's preferences to the system. If a bidder receives resources from several providers in equilibrium, he risks paying more than its marginal value when adding up resources from all auctions. In this case, a bidder would pay a negative rent for the resources obtained and would be better off by not bidding at all. The simplified algorithm for *BidAll* is given as Algorithm 1.

UtilityBased: The *UtilityBased* bidding strategy coordinates bidding on several auctions by comparing the utility expected to be received and selecting the

Algorithm 1. *BidAll* Bidding Strategy

```
loop
  Receive results from all active auctions
  for all active auctions do
    Generate a truthful reply
    if truthful reply can be generated then
      send new bid to auction
  sleep for 1 second
```

auction with the highest utility in each period.[1] Only one new bid is submitted in each period. Bids from previous periods stay active but might be overbid by other bidders in the following rounds. With this bidding strategy a bidder reduces his risk of overbidding since he only sends one truthful reply in each period. However, in equilibrium, bidders can potentially end up with resources allocated from more than one auction as bids from previous periods might be still winning bids. An algorithmic description of *UtilityBased* is presented as Algorithm 2.

Algorithm 2. *UtilityBased* Bidding Strategy

```
VARIABLES: highest_auction
loop
  Receive results from all active auctions
  for all active auctions do
    if truthful reply can be generated then
      if expected utility from truthful reply > utility[highest_auction] then
        Save current auction index in highest_auction
      else if received utility > utility[highest_auction] then
        Save current auction index in highest_auction
  if for highest_auction a truthful reply can be generated then
    send new bid to highest_auction
  sleep for 1 second
```

ComplementaryUtility: This bidding strategy implements the idea of "dividing up" the demand for bandwidth between auctions. In each step the auction with the highest utility is determined and a new bid is sent out. Other auctions with lower utility are seen as additional sources to 'complement' the resource allocation from the highest auction. But instead of risking to overbid at other auctions, a bidder adapts his demand function by subtracting the quantity expected on the leading auction. This lowers the chances of winning on other auctions but prevents overbidding situations since the bidder is truthfully revealing his value under the assumption that results on the first auction can be achieved. Figure 1

[1] We define the expected utility as calculated from the bid to be submitted and the expected utility as the consumers welfare obtained from the share of resources won in the last round.

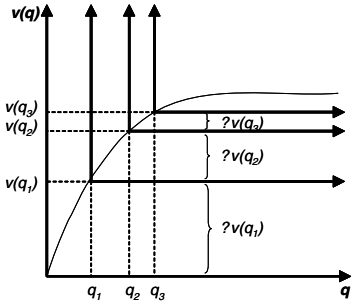


Fig. 1. Definition of valuation functions for subsequent auctions

shows how the demand functions for subsequent auctions are implemented. If an agent has obtained q_1 from the auction with the highest utility it can form a new valuation function beginning at $q_1, v(q_1)$, which can be used for other auctions, complementing the already obtained resources. Algorithm 4 depicts the simplified algorithm for *ComplementaryUtility*.

Algorithm 3. *ComplementaryUtility* Bidding Strategy

```
VARIABLES: sorted_auction_list[ ], i
loop
  Receive results from all active auctions
  for all i = active auctions do
    if truthful reply can be generated then
      Sort result into sorted_auction_list[i]
  for all i = auctions in sorted_auction_list[ ] start with the highest do
    if for sorted_auction_list[i] a truthful reply can be generated then
      send new bid to sorted_auction_list[i]
      form a new valuation function with remaining utility
  sleep for 1 second
```

5 Simulation Approach

We employ simulation as the main research methodology. Simulation allows us to translate the defined bidding strategies into software code and to directly observe equilibrium results with several settings and with different input parameters. Since bidding within the PSP context happens in multiple rounds we are also able to observe the bidding behaviour over time as well as the progression of aggregated values such as provider revenue or overall social welfare.

While in principle it is possible to use mathematical modelling to obtain exact results in terms of convergence and equilibrium results, we believe that because of the complexity introduced by the competitive setting and the ability of bidders to obtain results from multiple sources, closed solutions can only be expected

in a very few and specialised cases. Therefore, we see simulation as a tool for discovering the emergent properties of the developed bidding strategies and to apply a more rigorous analytical analysis in a second step.

Additionally, simulation allows us to gain a richer picture of the proposed bidding strategies, which are impossible to analyse with alternative research methods. For example, we can introduce an additional bidder when a market has already come to equilibrium an observe the consequences in terms of convergence time and allocation of resources. For the development of the simulation platform we have made use of the standard development techniques described in the literature (for an overview see e.g., [8]). This especially applies to the model verification after the basic implementation and the design of the simulation experiments.

The general simulation platform, which has been developed with the objective of reusability and openness toward alternative market mechanisms, has been developed in Java using the *Java Agent DEvelopment Framework* (JADE)[2]. JADE provides a middleware concept to set up multiple, independently acting software agents. Each market participant can be modelled as a separate agent entity with a specific behaviour profile. This also allows for the setup of a mixed agent population in which each agent employs a different bidding strategy. The JADE communication protocol provides a simple implementation of agent interaction in form of messages. Additionally, JADE provides a generic discovery service to dynamically identify other agents with certain properties.

A detailed discussion of the simulation architecture and the developed agent ontology can be found in [9].

6 Experimental Results

We have conducted two types of experiments. In the first type of experiments input, parameters are deterministic but due to the timing of events (for example, in which order bids are submitted to the auctioneer), different results can emerge. The second type aims at understanding the dynamic behaviour of the proposed bidding strategies. Users profiles are randomly generated.

We assume that agents use a second-order (parabolic) valuation model of the form

$$\theta_i(q) = \begin{cases} -\frac{\overline{p}_i}{2\overline{q}_i}q^2 + \overline{p}_i q & \text{for all } q \leq \overline{q}_i \\ \frac{\overline{p}_i \overline{q}_i}{2} & \text{for all } q > \overline{q}_i \end{cases}$$

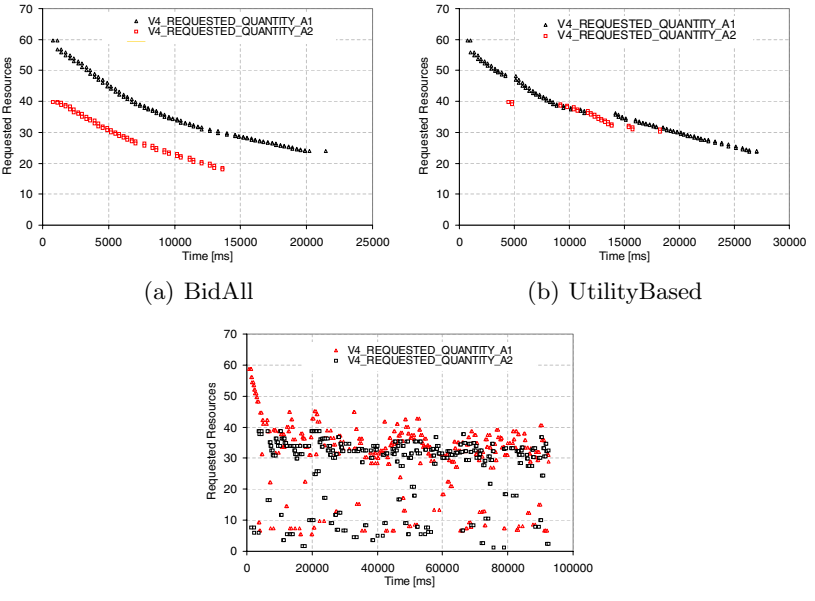
The parameters $\overline{p}_i$ and $\overline{q}_i$ can be interpreted as follows: $\overline{p}_i$ defines the marginal unit price of player i at quantity 0, and $\overline{q}_i$ defines the maximum quantity share a player wishes to win. This type of valuation function has been proposed and substantiated by Semret [4] and simplifies the implementation of the corresponding calculations by the simulator.

[2] A more detailed description of the JADE environment can be found at http://jade.tilab.com

6.1 Scenario 1: A Five-Bidder Case with Two Providers

A very simple setup, in which five agents have access to two providers, is used to demonstrate the basic properties of the three bidding strategies.

In Scenario 1 two wireless networks and five customers are represented by software agents. All customers have access to both wireless network providers, which offer resources of $Q^{(1)} = 60$ and $Q^{(2)} = 40$, respectively. The factor ϵ has been set to 10 and bidders update their bids every $300msec$. The values of $\overline{q}$ and $\overline{p}$ for each bidder are (90,10), (85,12), (80,15), (70,20) and (65,22).



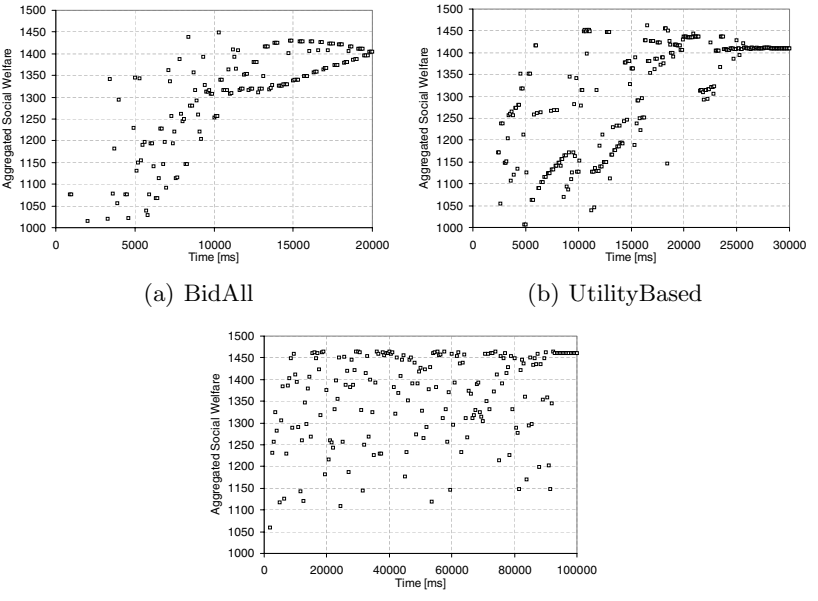
(a) BidAll (b) UtilityBased

(c) ComplementaryUtility

Fig. 2. Requested quantities of *BidderAgent4* over the simulation period with each of the proposed bidding strategies

For each bidding strategy we run an experiment and record the results over time, showing the results obtained by *BidderAgent4* in Figure 2. As expected, with the *BidAll* bidding strategy, bidders reduce their demand on all auctions until equilibrium is reached. The same behaviour can be observed for the *Utility-Based* strategy. However, several steps are undertaken when bidders stay inactive on one auction while bidding on the other auction. This process delays the final convergence to equilibrium.

The behaviour of *ComplementaryUtility* differs from both other strategies because no smooth convergence to equilibrium can be observed. Instead, bidders change bids on both auctions erratically depending on their opponents' profiles received from the last round. While a stepwise convergence (bidders start to decrease their bids continuously) can be observed for short time intervals,



(a) BidAll (b) UtilityBased

(c) ComplementaryUtility

Fig. 3. Aggregated social welfare over the simulation period with each of the proposed bidding strategies

the strategy is non-converging in general. However, due to the simple setup of the simulation experiment we can observe that the market achieves a ϵ-Nash equilibrium. Since the experiments are conducted in an agent-based simulation environment without central synchronisation, the equilibrium and the convergence process depend on the order of bids submitted. Therefore, results differ in each simulation run.

In a second experiment we have tested the relation between the factor ϵ and the convergence time to equilibrium. While for the two converging bidding strategies a clear relation between an increasing ϵ and a decreasing convergence time can be observed, the relation for the *ComplementaryUtility* is not obvious.

Besides convergence, we are also interested on the performance of the system measured through the aggregated welfare in equilibrium. Aggregated (social) welfare is defined as the sum of the revenue and the consumer surplus for each simulation run and is measured in a fictitious monetary unit. For the given example we can analytically derive the optimal allocation and resulting maximal welfare to be 1465.85. Figure 3 shows the progression of aggregated welfare over time until equilibrium has been reached. It can be observed that with the *BidAll* strategy and the *UtilityBased* strategy the equilibrium reached is not welfare maximising. With an aggregated social welfare of 1459.1 the *ComplementingUtility* strategy reaches an outcome in equilibrium that is within the ϵ-Nash boundaries. For this special case we can therefore conclude that this strategy allocates efficiently.

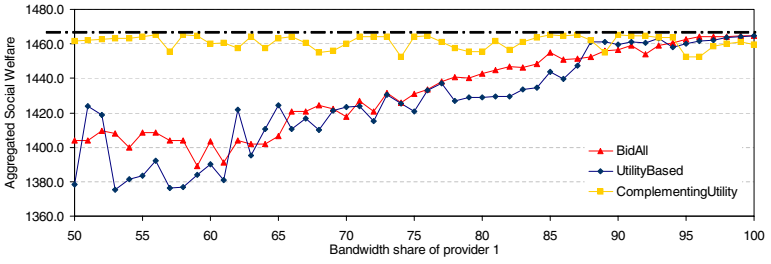


Fig. 4. Aggregated social welfare when shifting resources between auctioneers from [50,50] to [100,0]

So far, we have kept the allocation of resources between the two auctioneers fixed. In the next experiment we aim at understanding the change in aggregated social welfare when the relative share of resources between providers is gradually changed. For each bidding strategy, 50 experiments were conducted. In each run the distribution of resources between the two providers was changed from $Q_1 = 50, Q_2 = 50$ to $Q_1 = 100, Q_2 = 0$. When the equilibrium at both auctions was reached, the values for revenue and consumer surplus were recorded for all bidders. Figure 4 shows the aggregated social welfare for each possible allocation of resources between the two providers.

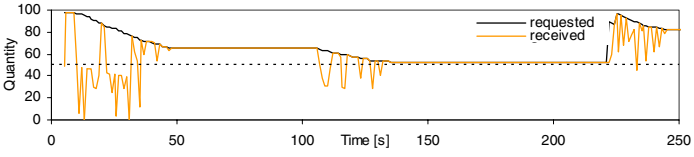
A prominent result is that for the two bidding strategies, *BidAll* and *UtilityBased*, the total welfare generated by different combinations of proportions in which providers supply the access market approaches the maximum as one seller's share becomes larger than the other's. There is some loss in efficiency when the market is equally supplied in comparison to the one-provider situation. The *ComplementaryUtility* strategy produces equilibrium results which are in the defined bounds of the 2ϵ interval. Since agents seem to bid more carefully when using the *Utility Based* strategy, because they only submit a new bid when it provides a higher utility than the current bid does, we would expect such strategy to improve consumers' surplus over the *BidAll* strategy. However, when providers equally supply the access market, *UtilityBased* yields more revenue to them than *BidAll* does.

6.2 Scenario 2: Bidding Behaviour in a Complex Scenario

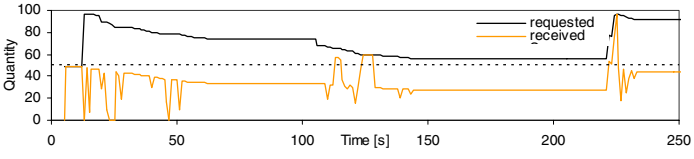
For the second simulation scenario we define a more complex setting and randomly create agent profiles and locations. We can summarise the setup as follows:

- Two network providers are running running four access points (AP) each to cover an area of 500 by 500 units. Access points are represented by agents offering network resources. The entire area is covered by both providers. Each access points offers a capacity of $Q = 300$.
- 100 user agents are randomly distributed over the service area. All users have a constant maximum demand of $\bar{q} = 50$ and a maximum marginal unit price $\bar{p}$ generated from a uniform distribution on the interval $[10, 20]$. All

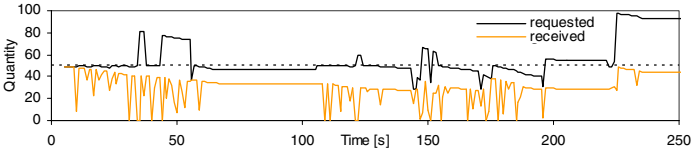
agents have access to only one provider, which has been randomly selected. If a user can access more than one AP it selects the AP closest to him.
- 70 agents initially request service. 30 agents join the market place at $t = 100sec$. 50 randomly selected agents leave at $t = 220sec$.
- One agent with $\bar{q} = 50$ and $\bar{p} = 15$, which has access to both providers, is located at position $(200, 200)$. In three different experiments he uses the *BidAll*, *UtilityBased*, and *ComplementaryUtility* strategy, respectively.



(a) BidAll



(b) UtilityBased



(c) ComplementaryUtility

Fig. 5. Summed requested and received quantities for the bidder with access to two providers under the three bidding strategies

In all experiments we record the requested and received resources for this agent. Figure 5 shows the results. With the *BidAll* strategy the agent is able to acquire the the highest amount of resources. However, since bids to the different auctions are not coordinated, he also receives more than his actual demand for some time periods. While the total price may not be above his willingness-to-pay he captures units of the resource, which bring no additional value to him.

With using the second bidding strategy we observe that while overall requested quantity is comparable to the BidAll strategy. However, since the player is coordinating his bids, the received quantity is not larger than his maximum demand for longer periods of time. This is because a bidder may still have a valid bid in an auction but is not updating it any more because resources on other auction places have become more attractive.

With the *ComplementaryUtility* strategy the player bids much more cautiously. The received quantity always stays below the maximum demand. Compared to the other two strategies the identification of equilibrium is erratic and the process to get to a stable allocation takes much longer. This is especially true for the second time period, when a total of 100 players are present in the market.

7 Conclusion

In this paper we have presented a simulation approach to three bidding strategies if players are allowed to satisfy their individual demands by aggregating shares from bidding for resources at two or more auctions. We have endowed each seller with the Progressive Second-Price auction, which provides a rich framework as the auction implemented at a single seller is efficient and incentive-compatible. Simulations aim at learning whether the convergence properties of PSP hold at each seller when bidders either make coordinated (*UtilityBased* strategy) or no coordinated decisions among auctions (*BidAll* strategy), or complement their already won share at a given auction by bidding at other auctions if they need to (*ComplementaryUtility* strategy). Results provide an idea on how social welfare is affected by the aggregated behaviour of the bidders. Also, we can observe how the different bidding strategies influence the bidding behaviour of a single bidder when given the option of having access to multiple service providers.

References

1. Berezdivin, R., Breinig, R., Topp, R.: Next-generation wireless communications concepts and technologies. IEEE Communications Magazine **40**(3) (2002) 108–16
2. Hardin, G.: The tragedy of the commons. Science **162** (1968) 1243–48
3. Vickrey, W.: Counterspeculation, auctions and competitive sealed tenders. Journal of Finance **16** (1961) 8–37
4. Semret, N.: Market mechanisms for network resource sharing. PhD thesis, Columbia University, Center for Telecommunications Research (1999)
5. Krishna, V.: Auction theory. Academic Press (2004)
6. Maillé, P., Tuffin, B.: Multi-bid auctions for bandwidth allocation in communication networks. In: the proceedings of the 23rd IEEE Infocom Conference, Hong-Kong, China. (2004)
7. Zeng, D., Cox, J., Dror, M.: Coordination of purchasing and bidding activities across markets. In: the proceedings of the 37th Hawaii International Conference on System Sciences. (2004)
8. Woolridge, M.J.: An introduction to multiagent systems. New York: J. Wiley (2002)
9. Roggendorf, M., Beltran, B., Gutierrez, J.: Architecture and implementation of an agent-based simulation tool for market-based pricing in next-generation wireless networks. In: the proceedings of TridentCOM 2006, 2nd International IEEE/Create-Net Conference on Testbeds, Barcelona, Spain, March 1-3. (2006)

Charging in Peer-to-Peer Systems Based on a Token Accounting System

Nicolas Liebau[1], Oliver Heckmann[1], Aleksandra Kovacevic[1], Andreas Mauthe[2], and Ralf Steinmetz[1]

[1] Technische Universität Darmstadt
[2] Lancaster University

Abstract. Today, Peer-to-Peer applications are predominant on the internet when considered in terms of its traffic consumption. However apart from Skype, their commercial success is still very limited. This is due to the difficulties faced when trying to implement crucial functionality such as accounting and charging without violating the Peer-to-Peer paradigm. A fully decentralized accounting scheme based on tokens was presented by the authors last year. In this paper we analyse the interactions between token-based accounting and charging in order to enable peers to charge for their services. We present three different charging schemes using tokens as (1) pure receipts, as (2) Micropayment, and as (3) bill of exchange. These schemes are evaluated based on the provided security and the overhead traffic introduced into a Peer-to-Peer system.

1 Motivation

Apart from Skype, the commercial success of Peer-to-Peer (P2P) applications is negligible. Internet Service Providers believe that the future of P2P is very promising in the combination with Triple Play [1], due to the strong interest of customers today in private content, which can be delivered efficiently using P2P. Besides this, other P2P applications have been envisioned whereby peers have to pay for services which they receive. However, it is still an open question for service providers how to charge for the services within a P2P system. A basic requirement for P2P business applications is a P2P architecture which supports commercial services. Often such an architecture is provided by the manufacturer [2, 3]. In this paper, we do not consider payment models used by the manufacturer to charge the peers using his P2P platform. Instead, we focus on the P2P business applications whereby peers charge for their services delivered.

The requirements for an architecture suitable in supporting such business applications and related work about charging systems are summarized in Sect. 2. A core requirement is a reliable, trustworthy accounting mechanism that complies with the P2P paradigm [4]. We have developed a token-based accounting mechanism which fulfils these requirements (see [5]). A short overview is given in Sect. 3. In this paper, we present and analyse three charging alternatives which can be added to our token-based accounting scheme. These alternatives are

B. Stiller, P. Reichl, and B. Tuffin (Eds.): ICQT 2006, LNCS 4033, pp. 49–60, 2006.

presented in Sect. 4. In Sect. 5 we compare the different alternatives in terms of the transaction costs born by the peers. In Sect. 6, we draw the conclusions.

2 Requirements for P2P Business Applications

Peer-to-Peer business applications that offer service providers the possibility to charge money for their services have to fulfil several requirements. The fundamental mechanism needs to be able to determine supply and demand, both of which can be determined using the search functionality in P2P. Further requirements include pricing, metering and accounting, charging, billing and payment [6, 7, 8, 9, 10]. We will now present the process from pricing to the final payment together with the related work.ta

Pricing. Before peer A requests a service from peer B, first, both peer A and B must agree on the service and its price. This price will be expressed in the form of a tariff. There are several options for determining a price, e.g. negotiations or auctions [11, 12, 13]. For a fair market the availability of price information is important. Price or tariff distribution is discussed e.g. in [14].

When A and B agreed on both service and price, the service will be delivered (e.g. the file will be uploaded by B to A). This period is called service session. During the service session, other functions mentioned above are also required. Several integrated frameworks in different fields of the Internet have been presented [15, 16].

Metering. Metering is the process of objectively observing events happening within the P2P system and communicating them to the accounting system. In P2P systems metering is limited to local observation

Accounting. By using information from metering, accounting creates receipts and may distribute these within the system for storage. Thus, receipts contain information about the events which the peers claim to have happened. It is the most objective information about service sessions available. Examples of accounting mechanisms for P2P systems are [17, 18, 5], see also next section.

Charging. Charging combines the accounting information provided, with the tariff which the transaction partners agreed upon and calculates the charge, the actual amount of money the service requestor has to pay to the service provider. Charging can be an ongoing process during the service session, an once only process at the end of the service session, or even an aggregating process over several service sessions. Examples for P2P based systems are [19, 18]. The charging information is fed into a billing and payment system.

Billing and Payment. The billing functionality creates a bill which states, among with other information, the amount that the requestor has to pay to the provider. As money is something external to the P2P application, we also assume that the

P2P application will use a billing and payment system which is external for the P2P application. The different options for payment are e.g. direct money transfer between bank accounts, online payment systems like PayPal [20], Micropayment systems like eCash [21].

Obviously, there are many alternatives for how a P2P platform for business applications may be built. Examples include the projects MMAPPS [22] and P2P Yardsale [23]. Examples from other domains include [15, 16].

3 The Token-Based Accounting System

The basic concept of our Token-based Accounting Scheme (TbAS) involves a service requestor paying tokens in return for a service provided. Tokens serve as receipts for services provided. Every token has an associated owner, i.e. only the owner may spend his tokens. Accordingly, service providers will collect foreign tokens from various service requestors. A service provider cannot respend foreign tokens he collected but only exchange them in the so-called token aggregation process against new own tokens. This process of issuing new own tokens is fully decentralized and therefore follows the P2P paradigm. The exchange of tokens using a flexible exchange function enables the limitation of the number of tokens which a peer may possess. This allows the introduction of incentives for service provision within the P2P system. Further, behavior rules can be enforced by relating observed peer behavior with the number of new own tokens a peer receives in a token aggregation process. Next, the building blocks of the TbAS are explained in more detail. For further details please refer to [5].

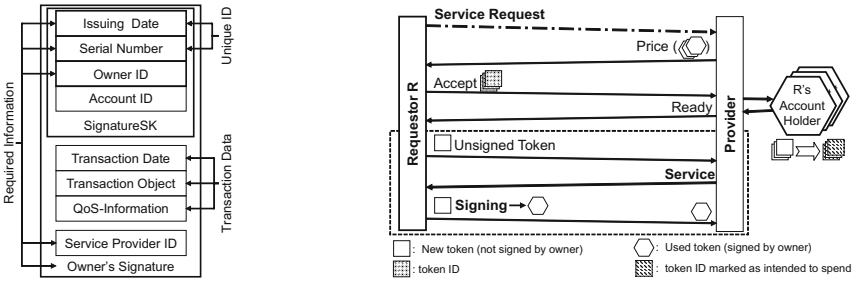
3.1 Token Structure

New tokens contain the owner's identification, e.g. the owner's public key, and a unique ID. To ensure integrity of this information and to prevent forgery of tokens, they are signed with the system's private key (SignatureSK) (see Sect. 3.3 and Fig. 1 (a)). The unique ID allows the detection of double spending. When the owner spends a token, he has to add the required accounting data, which includes the service provider, and then sign the token with his private key in order to achieve information integrity. The token structure is shown in Fig. 1 (a). A token is not anonymous because its main purpose is to provide accountability in a P2P system. However, using the cryptographic scheme presented in [24], anonymity could also be added if desired.

3.2 Payment Process

The payment process of the TbAS is depicted in Fig. 1 (b). In order to prevent double spending, for each peer in the P2P system there exists a set of third peers (the so called account holder set) which keep track of the tokens issued to a peer and tokens spent by the peer. Before a service session begins, the requestor discloses to the provider the IDs of the tokens the requestor intends

to spend for receiving the service (see Fig. 2 (b)). Now, the provider can check if these tokens are valid. To avoid that the requestor double spends the tokens in a parallel transaction, account holders will mark these tokens as intended to be spent. Thus, double spending is avoided.



(a) Token Structure (b) Payment Protocol

Fig. 1. Token structure and payment

3.3 Token Aggregation Process

After a peer has collected foreign tokens, it will have to exchange these foreign tokens against new own tokens in order to receive further services. The token aggregation process will determine the amount of new tokens the exchanging peer should receive and create a signature with the system's private key on the new created tokens to provide validity. The process is depicted in Fig. 2 (a). In order to create the system signature in a fully decentralized way, a subset of peers of the P2P system is selected as so-called trusted peers based on their reputation (the TbAS assumes that a reputation system is present within the P2P system). The exchanging peer (EP) sends its foreign tokens to a trusted peer (TP_1). TP_1 calculates the amount of new tokens to be created using the global aggregation function. It creates the new tokens (without system signature) and sends their IDs to EP's account holder set (see Fig. 2 (b)). The account holders update the list of tokens available to EP. Now TP_1 further chooses k trusted peers who create the system signature using the threshold cryptography scheme presented in [25]. The system's private key is split into parts and each trusted peer owns one of these parts. k key parts are required to create a signature with the system's private key. Each trusted peer involved sends the tokens signed with the partial key back to EP, who reconstructs the final system signature. In this way, the system's private key is not compromised.

3.4 Token-Based Accounting Scheme as Incentive Scheme

We have studied the use of TbAS in a file sharing scenario, whereby users pay one token per 1 MB of file size in order to receive the file. Whenever a peer does not have enough tokens to download another file, the peer exchanges foreign

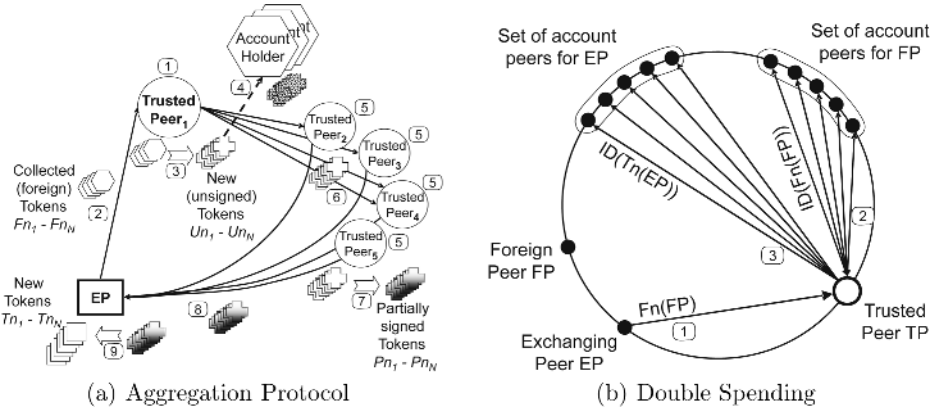


Fig. 2. Aggregation and double spending protection

tokens collected against new own tokens. Each peer receives a specific amount of tokens upon entering the P2P system.

In [26], we examined the file sharing scenario for different aggregation functions. We have shown that the idea of using tokens as virtual currency (aggregation function set to $N = F$ (N =amount new tokens, F =amount foreign tokens)) will lead to market failure in the presence of altruistic peer. Altruistic peers provide much more services than they consume and therefore accumulate the majority of tokens in the P2P system. Accordingly, other peers do not have the possibility to redeem enough own tokens to be able to request further services. This problem can be solved by using weak or asymmetric incentives.

Now we consider the use of the accounting mechanism within a P2P market, where users pay actual money for receiving services. For these scenarios a charging system has to be added.

4 Charging Based on Token-Based Accounting Scheme

This section covers three possible alternatives for charging within a P2P application where users pay actual money for receiving services. Thus, we assume a P2P application providing the functionality as described in Sect. 2. Further, we assume that each peer owns a private/public key pair which enables it to provide legally valid signatures. This means that before a service session starts, the peers agree on the service to be provided and a tariff for calculating the charge of the service.

4.1 Tokens as Receipts

Concept. The service requestor (A) will send one or several tokens to the service provider (B) as receipt(s) for delivered service. B can use these tokens to demand payment from A via a prior agreed billing and payment system. Each peer can request any amount of tokens using the token aggregation protocol. Tokens are not exchanged, only new ones are created.

Discussion. Here, tokens serve the same purpose as receipts created by transaction partners without having to be issued before. Receipts not issued must remain non-forgeable and double spending has to be detectable. This however does not have to be system wide but only between the transaction partners. Both are easy to achieve through the use of signatures and unique receipt ID. Thus with the TbAS, it seems unnecessary to issue receipts.

However, such issuing of receipts offers the possibility of decentralized control within P2P systems. E.g., it can be controlled who is allowed to participate in the P2P system. This can be used to exclude peers with a bad reputation. Further, the number of tokens available to a peer can be limited. Thus, a peer can do only a limited number of transactions between two token aggregations. This limits the danger of misuse of the reputation system, as seen at eBay; A person could be well behaved until he has a high reputation value; then suddenly he starts to defraud his customers by not sending the purchased good. The person could continue this for some time until it becomes clear that he is a fraud. The limitation of the number of tokens available to a peer is possible, because peers aggregate tokens only after a transaction is completed to mutual satisfaction. To further limit possibilities of fraud, for higher valued services peers could agree on a higher amount of tokens. The enhanced functionality described is especially wise for P2P business applications, as there is no central instance which users could contact in case of fraud (as there is in eBay).

In order to make fraud limitation effective for both the service requestor and provider, TbAS has to be adapted so that both transaction partners must spend tokens for a transaction. Both, the requestor when the service was received and the provider when he received the payment, must spend tokens. Otherwise, only service requestors could be excluded from the system.

It is apparent that this charging scheme also requires a fast payment scheme. Should the payment require e.g. several days to arrive at the service provider (as in eBay), the P2P business application is a lot less attractive.

4.2 Tokens as Micropayment

Concept: When using tokens as Micropayments such as eCash [21], each token symbolizes a specific amount of money. Users use tokens to pay for receiving services.

Discussion: In comparison to existing Micropayment schemes tokens are not anonymous but can be modified to be (see Sect. 3.1 and Fig. 1 (a)). When using tokens as Micropayment protection against forgery and double spending is highly important. In the TbAS the forging of tokens is still possible under certain circumstances. However, it is highly unlikely (see [5]). Further, without a central bank it is not trivial problem to solve whom users should pay in order to receive the tokens necessary for requesting services. A central bank to host the user's accounts and provide the token aggregation functionality would solve this issue. However, this compromises the P2P paradigm.

A solution without a central bank would require the cooperation of several banks with the (manufacturer of the) P2P system. A user would pay money

to a participating bank which would in return create a certificate that entitles the user to the receive an amount of tokens. The peer (user) would present this certificate to a trusted peer for token aggregation in order to receive the tokens. It is important that the TbAS ensures that certificates are redeemed only once. The peer's account holders can save this information or a callback function with the banks is possible.

An advantage of using tokens as a micropayment scheme is that peers could exchange foreign tokens received against new own tokens by using token aggregation instead of exchanging them at the bank. This reduces transaction cost.

4.3 Tokens as Bills of Exchange

Concept: Tokens can also be used as a bill of exchange. A bill of exchange is a written order in which one person pays another person a specific sum on a specific date. It can be enforced easily without being subjected to defenses. In the past, the bill of exchange was a very important instrument for trading. Today, it used primarily in international trade. A token is worth the amount of money stated in it. Further information required for a bill of exchange (date of issue, drawee, recipient, due date) must be contained in the token.

Discussion: This concept is similar to the first alternative (Sect. 4.1), however the legal consequences here are much more strict. Therefore, this concept has higher requirements on the peers' signatures, because they have the potential of being accepted internationally.

As an extended concept, a token used as a bill of exchange could be transferred by endorsement to another peer. The old recipient would add the new recipient under the token and sign it. However, now double usage of the token must be avoided (the old recipient could still claim the money from the drawee if he keeps a copy of the token). Therefore, the drawee must be informed about a transfer. If he is not available, the drawee's account holder set must store the information.

Tokens as bill of exchange also offer the opportunity for peers to charge up the mutually "drawn" tokens.

Tokens as bill of exchange could be handled similarly as tokens used as receipts, because each peer needs to get as many tokens as he requires for the services he requests (Sect. 4.1).

When applying this alternative there is the danger of fraud by the transaction partner who has to deliver second (as explained in Sect. 4.1). Therefore, it is more secure to use several tokens in transactions if the service can be delivered in parts. To simplify the status quo of mutual debts, tokens with fixed amounts of money are preferable. Further, the control mechanisms presented in Sect. 4.1 should be applied also here.

5 Assessment

In order to evaluate if a charging scheme can be applied in practice, the two most crucial criteria are scalability and security. We will assess and compare the three alternatives based on these two criteria.

5.1 Security

The security of a charging scheme is measured by its ability to prevent double spending and token forgery. The main mechanism used to avoid double spending is the account holder set. As this increases in size, the system becomes more reliable. The reliability depends on the probability distribution of the life time of the peers. We modeled the reliability using Marov Chains with a life time distribution derived in [27]. For an availability of the account holder set of 99% a set size of $k = 6$ is required.

In order to avoid forgery we require a signature with the system's private key that can only be created by a quorum using threshold cryptography. We assume the employed threshold cryptography scheme [25] to be secure. Further, we require a least one honest peer in the quorum in order to avoid forgery. In [5] it has been shown that the probability for a quorum consisting of only bad peers ($p(T, t, p_g)$) can be calculated using:

$$p(T,t,p_g) = \frac{\binom{T(1-p_g)}{t}}{\binom{T}{t}}$$

T =Total number of trusted peers

t =quorum size

p_g =percentage of good trusted peers

Tokens as Receipts. This alternative has the least security requirements compared to the other two alternatives. It is sufficient, if a defrauding peer must assume that double spending will be detected. Therefore, the account holder set can be kept small. An account holder set size $k = 6$ was selected for the traffic analysis. In order to calculate the quorum size, we assumed $p_g = 67\%$ and require $p(T, t, p_g) = 0.1\%$ which results in $t = 7$ for $T > 100$. If the total number of trusted peers is below 100 the required quorum size is below 7.

Tokens as Micropayment. Here, strong peer IDs to enable enforceability of sale agreements is required. Further, this scheme requires very tight security against forgery and double spending as this is equipollent to creating money. Assuming that 50% of the peers are bad and a probability of 0.01% for at least one good peer in the quorum, a quorum size $t = 14$ is required. To prevent double spending, the account holder set size also needs to be increased to $k = 16$.

In order to make forging the initial tokens created from a bank certificate impossible, these can contain information of this certificate, which can also be held by the account holders. For any tokens created hereafter, the other security mechanisms must be sufficient.

Tokens as Bills of Exchange. In this alternative, the transfer by endorsement is the most critical part because different scenarios for cheating exist here. First, receiver B transfers a token to receiver C. Then B agrees with the drawee A to be paid 50% of the amount of the token. A would save 50% and B gains another 50% and C would not be able to collect the money from A. As time stamps can

be easily forged, it is hard to decide which happened first, the token transfer or the payment of A to B. In order to prevent such fraud, the account holder set must always note the actual holder of a token and each clearing of a token to remove it from its list. Accordingly, it is important that the account holder set is available and therefore its size needs to be increased to 16 as calculated above.

As token aggregation is primarily used for the limitation of fraud as in the "tokens as receipts" alternative, the quorum size is similarly configured. In order for an effective limitation of fraud in this scheme, it is required that the service provider sends an own token to the service requestor. These tokens are not allowed to be transferred as they are not bills of exchange.

5.2 Scalability

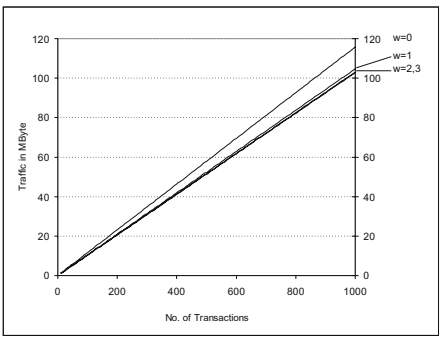
In [5] the scalability of the TbAS was evaluated using measurements of our prototype based on JXTA [28], simulations and a worst case scenario analysis. It has been shown that the traffic overhead which TbAS introduces into a P2P file sharing system, where one token is exchanged for 1 MByte file size, is approximately one percent. The overhead traffic for the three charging alternatives is analysed by using the worst case analysis, considering the different required configurations of the TbAS. In order to compare the overhead of a charging scheme based on simple receipts without tokens, this alternative was also evaluated as shown in Fig. 3 (b).

Tokens as Receipts. We have evaluated the scalability of charging based on receipts with extended mechanisms in order to limit the possibilities of fraud (see Sect.4.1). We have assumed that peers exchange received foreign tokens in batches. The number of messages generated per transaction can be determined using $M(k) = 2s + 4k$ where k is the size of the account holder set and s is the amount of tokens used for the transaction. The number of messages generated by a token aggregation process can be calculated using $M(k, t, b) = \frac{ns}{b}(1 + 2k\frac{b}{s} + 2k + 2t)$, where t is the quorum size, n is the number of transactions that are considered by the aggregation, b is the batch size of aggregated tokens. In comparison to a file sharing scenario, double the amount of token aggregation processes will have to be executed, because both, the requestor and receiver use tokens in these transactions and have to aggregate them. We have assumed a quorum size of 7 and an account holder set size of 8. (see last section). The resulting traffic for a batch size of $b = 20$ is depicted in Fig. 3 (b).

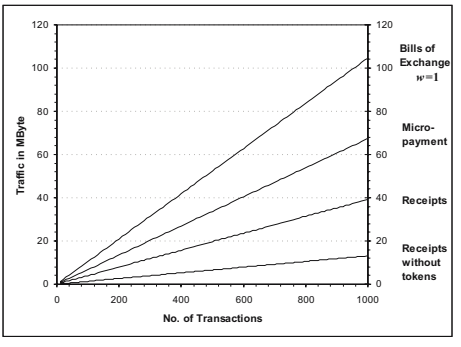
Tokens as Micropayment. When using tokens as Micropayment, the traffic created by the TbAS is comparable to the traffic generated when tokens are used as an incentive in a file sharing scenario (see [5]). However, the system parameters have to be adjusted according to the security requirements (see last section). Accordingly, for the results presented in Fig. 3 (b) a quorum size of 14 and an account holder set of 16 was assumed.

Tokens as Bills of Exchange. The traffic overhead of this alternative is similar to the file sharing scenario of [5], however, the possibility of token transfer by

endorsement has to be considered. By paying for a service with a token which the requestor received as bill of exchange, means that message sizes are larger but with fewer token aggregations. The effect of this coherence is shown in Fig. 3 (a), where w is the average number of transfers by endorsement.



(a) Charging using Bills of Exchange



(b) Traffic Comparison of the Charging Alternatives

Fig. 3. Generated overhead traffic

6 Conclusions

In most P2P systems today, some mechanisms which are required for business applications are still missing. These are crucial for effective accounting and charging functionality. In this paper, three alternatives for charging based on our Token-based Accounting Scheme [5] were presented. In general, the use of tokens has its advantages compared to using simple receipts. Especially, the number of tokens available to a peer can be limited. This can be used either as mechanism to resolve market failure or as a mechanism for limiting fraud possibilities, as the number of transactions which a peer may execute can be limited. In a P2P environment, this is especially important as the transaction partners are widely anonymous and therefore mechanisms which build trust are required. Therefore, identity management plays an important role as it is required in order to be able to identify defrauding peers clearly.

The additional functionality and control results in the generation of higher traffic overhead. As Fig. 3 (b) shows, in comparison to simple receipts without tokens, the overhead traffic generated increases by a factor 3 for token-based receipts, by a factor 5 for token-based Micropayments, and by a factor 8 for tokens as bills of exchange. In comparison to simple receipts without tokens, the additional traffic of approximately 26 kbyte for token-based receipts, 54 for token-based Micropayment, and 92 kbyte for tokens as bills of exchange was created. This traffic includes all transaction related traffic, but without key management. This is a worst case analysis based on measurements carried out with the JXTA based implementation of the TbAS.

The advantage of charging based on Micropayments or bills of exchange is the possibility for peers to charge up mutual debts and by doing so to save on banking fees. Further, especially when using tokens as Micropayment, peers receive their payment immediately. This means that customers can retrieve the requested service immediately and do not have to wait for a bank confirmation.

Security aspects become very important in P2P systems as soon as it involves real money. Therefore, it is questionable if users of banks would accept a Micropayment scheme which relies on a decentralized mechanism without a trusted third party. The presented charging scheme using tokens as micropayments can be considered secure, apart from the aggregation of foreign tokens for new own tokens, because after aggregation a new token cannot be traced back to the certificate signed by a bank.

References

1. Wikipedia: Triple play (telecommunications). http://en.wikipedia.org/wiki/Triple_play_%28telecommunications%29 (2006)
2. MetaMachine: eDonkey2000. http://www.edonkey2000.com (2004)
3. BBC: BBC integrated Media Player. http://www.bbc.co.uk/imp/ (2006)
4. Steinmetz, R., Wehrle, K.: Peer-to-Peer-Networking and -Computing. Informatik Spektrum **27** (2004) 51–54
5. Liebau, N., Darlagiannis, V., Mauthe, A., Steinmetz, R.: Token-based Accounting for P2P-Systems. In: Proceeding of Kommunikation in Verteilten Systemen KiVS 2005. (2005) 16–28 (Received Best Paper Award).
6. Androutsellis-Theotokis, S., Spinellis, D., Karakoidas, V.: Performing peer-to-peer e-business transactions: A requirements analysis and preliminary design proposal. In: IADIS International e-Commerce 2004 Conference Proceedings. (2004) 399–404
7. Schoder, D.: Suitability of p2p for business transactions. In: Proceedings of the Peer-to-Peer Systems and Applications Daghstuhl Seminar, March 2004. (2004)
8. Gerke, J., Hausheer, D.: Peer-to-Peer Market Management. In: Peer-to-Peer Systems and Applications. Volume 3485 of LNCS. Springer-Verlag (2005) 491–507
9. Gerke, J., Stiller, B.: A Service-Oriented Peer-to-Peer Middleware. In: Preceeding of 14. Fachtagung Kommunikation in Verteilten Systemen 2005 (KiVS 05). (2005)
10. Hummel, T., Muhle, S., Schoder, D.: Business Models and Revenue Models. In: Peer-to-Peer Systems and Applications. Volume 3485 of LNCS. Springer-Verlag (2005) 473–489
11. Hausheer, D., Stiller, B.: Decentralized auction-based pricing with peermart. In: Proceedings of 9th IFIP/IEEE International Symposium on Integrated Network Management (IM 2005). (2005)
12. Lang, K.R., Vragov, R.: A pricing mechanism for digital content distribution over peer-to-peer networks. In: HICSS '05: Proceedings of the Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS'05) - Track 8, Washington, DC, USA, IEEE Computer Society (2005) 211.1
13. Reichl, P., Hausheer, D., Stiller, B.: The Cumulus Pricing model as an adaptive framework for feasible, efficient, and user-friendly tariffing of Internet services. Computer Networks, Elsevier **43** (2003) 3–24
14. Heckmann, O., Darlagiannis, V., Karsten, M., Steinmetz, R.: A Price Communication Protocol for a Multi-Service Internet. In: Informatik 2001 - Wirtschaft und Wissenschaft in der Network Economy - Visionen und Wirklichkeit. (2001)

15. Stiller, B., Fankhauser, G., Plattner, B., Weiler, N.: Charging and accounting for integrated internet services - state of the art, problems, and trends. In: The Internet Summit (INET 98). (1998)
16. Briscoe, B., Darlagiannis, V., Heckmann, O., Huw, O., Siris, V., Stiller, B., Songhurst, D.: A Market Managed Multi-Service Internet. Computer Communications **26** (2003) 405–415
17. Hwang, J., Aravamudham, P., Liddy, E., Stanton, J., MacInnes, I.: Charging Control and Transaction Accounting Mechanisms using IRTL (Information Resource Transaction Layer) Middleware for P2P Services. In: International Workshops for Quality of Future Internet Services and Internet Charging and QoS Technologies. (2002)
18. Hausheer, D., Gerke, J., Stiller, B.: A generic and modular accounting and charging system for peer-to-peer applications. In: 14. Fachtagung Kommunikation in Verteilten Systemen 2005 (KiVS 05). (2005)
19. Roscoe, T., Hand, S.: Transaction-based charging in mnemosyne: A peer-to-peer steganographic storage system. In: Revised Papers from the NETWORKING 2002 Workshops on Web Engineering and Peer-to-Peer Computing, London, UK, Springer-Verlag (2002) 335–350
20. : PayPal. (http://www.paypal.com/)
21. Schoenmakers, B.: Basic Security of the ecashTM Payment System. In Preneel, B., Rijmen, V., eds.: Course on Computer Security and Industrial Cryptography. Volume 1528 of LNCS. Springer (1998)
22. MMAPPS: Project "Market Management of Peer-to-Peer Services". http://www.mmapps.info (2004)
23. Gao, R.: P2P Yardsale Engine (Project Venezia) & P2P Yardsale Application (Project Gondola). http://venezia-gondola.jxta.org/ (3)
24. Chaum, D., Fiat, A., Naor, M.: Untraceable electronic cash. In: CRYPTO '88. Volume 403 of LNCS., Springer Verlag (1990) 319–327
25. Luo, H., Kong, J., Zerfos, P., Lu, S., Zhang, L.: URSA: ubiquitous and robust access control for mobile ad hoc networks. IEEE/ACM Trans. Netw. **12** (2004) 1049–1063
26. Liebau, N., Darlagiannis, V., Heckmann, O., Steinmetz, R.: Asymmetric Incentives in Peer-to-Peer Systems. In: Proceddings of AMCIS 2005. (2005)
27. Darlagiannis, V., Liebau, N., Heckmann, O., Mauthe, A., Steinmetz, R.: Caching Indices for Efficient Lookup in Structured Overlay Networks. In: Proceedings of AP2PC 2005, Springer (2005)
28. Sun Microsystems: Project JXTA. http://www.jxta.org (2004)

A Market-Managed Topology Formation Algorithm for Peer-to-Peer File Sharing Networks

Tarik Idris[1] and Jörn Altmann[1,2]

[1] School of Information Technology, International University in Germany
76646 Bruchsal, Germany
[2] Techno-Economics & Policy Program, College of Engineering,Seoul National University
Seoul 151-744, South Korea
jorn.altmann@acm.org, tarik.idris@i-u.de

Abstract. Currently, peer-to-peer (P2P) networks suffer from users that do not contribute any kind of resources to the P2P community. Those users, which are called freeriders, benefit largely from contributions of other users but reduce the system performance for contributing users. This paper proposes an incentive scheme for P2P networks that motivates users to collaborate within the system. The solution that we propose has an impact on the topology formation of a P2P network. Using our market-managed topology formation algorithm (IUTopForm) for P2P networks, contributing users will be clustered within clubs that are different to clubs of freeriders. The differentiation is possible because of a reputation system, which considers users' past contributions. The effect of this approach is that service requests of freeriders will take longer to be answered (if at all) than service requests of resource-contributing users. We illustrate this effect through measurements with our P2P network simulator. We also show that clubs are only interconnected if the difference in their reputation values is not large. The comparison with Bagla and Kapalia's approach, which inspired our work, shows that the IUTopForm approach improves the overall utility of the system. The utility function and the topology formation algorithm are described in detail within this paper.

Keywords: Incentive Scheme, Pricing, Peer-to-Peer Networks, Simulation, Market-Management, Trust, Reputation, Economics, Utility Function.

1 Introduction

Although there have always been applications that used the peer-to-peer (P2P) paradigm (e.g. *USENET* [22] and *FidoNet* [12]) from the beginning of the network era, this technology has received much more attention during the last 5 years. On the one hand, this was caused by widespread availability of network computers and decreasing bandwidth costs. On the other hand, "killer applications" like *Napster*, *Kazaa*, and *Gnutella* were developed, which allow file sharing in a very simple way. However, P2P networks suffer from two issues. First, in order to stop illegal file sharing, the music industry injects faked files into the network. Second, and even more severe, many P2P network users do not provide any resources to the P2P network. These so-called freerides benefit highly from contributions of other users. Consequently,

B. Stiller, P. Reichl, and B. Tuffin (Eds.): ICQT 2006, LNCS 4033, pp. 61 – 77, 2006.

resource-contributing/cooperating users suffer in such a network from reduced performance.

In order to solve this problem, an incentive scheme has to be integrated into P2P networks. An incentive scheme provides benefits to those users who contribute resources (may it be content or hardware) to the network and reduces the performance of users who do not contribute. A few existing approaches on incentive schemes have been proposed during the last years. The one that we consider in this paper are from Asvanund, Bagla, Kapadia, Krishnan, Smith, &Telang [5], Walsh & Sirer [23], Yang, Chen, Zhao, Dai, & Zhang [26], Cohen [8], and Feldman, Papadimitriou, Chuang, & Stoica [10]. While the first four describe implementations of incentive schemes, the last analyzes the importance of incentive schemes for the success of P2P networks.

Our approach, the market-managed topology formation algorithm (IUTopForm) is based on a new utility function, which considers the amount of shared content of a user, the distance to another user, the similarity in taste with another user, and the other user's reputation. Since we especially focus on P2P file sharing networks, the content that the utility function considers are files. Note, any future reference to a P2P network will be in the context of a file sharing application. The topology formation algorithm, which we present in this paper, achieves significant performance improvements for P2P networks through a combination of different approaches. These approaches are:

- The definition of datasets to find similar interests
- A use of a reputation system
- The definition of a utility function for users
- The definition of a utility function for clusters (clubs)

The remainder of the paper is organized as follows. Section 2 gives an overview about the problems that P2P file sharing networks faced in the past. The architecture of version 0.6 of the Gnutella network, the version that our simulation is based on, is described in Section 3. An overview about previous research on incentive systems for P2P systems is given in Section 4. Our topology formation algorithm is described in Section 5 and Section 6. Finally, we conclude by presenting our measurement results in Section 7.

2 File Sharing of Music Files and Countermeasures

Napster was first released in the fall of 1999 and became increasingly popular during 2000, introducing millions of users to P2P file sharing, more specifically, to sharing of digital music files in MP3 format. Napster's architecture was not purely distributed, since the search for files was performed on a central server. Therefore, in July 2001, the service could easily be shut down by a judge's order after the music industry successfully sued Napster for copyright infringement [16].

Given the success of Napster, it came as no surprise that many purely decentralized file sharing applications were created to provide a similar service to about 13 million users that Napster had during its peak. One of the earliest attempts was Gnutella, in March 2000. The first client was developed by *Nullsoft*, a small developer studio owned by *AOL*. Its development continued in several distinct projects. Nowadays,

there are numerous different clients interoperating by the standards defined by the Gnutella Developer Forum (GDF) [23]. The original Gnutella protocol is nowadays referred to as Gnutella version 0.4 and is rather outdated, although it was a technical revolution at its time. Changes and extensions, which are incorporated in all major clients, will soon be standardized as Gnutella version 0.6 by the GDF [13]. However, scalability issues with the 0.4 protocol hindered a success similar to Napster's.

The *FastTrack* network and its most popular client Kazaa did not have these technical problems. The FastTrack protocol is fully decentralized and allows for swarming, i.e. downloading different parts of the same file from several hosts to increase throughput through parallelization.

Recently there has been a significant decline in users on the FastTrack network [17]. This is attributed to several factors. Firstly, the lack of innovations in the P2P clients, the protocol, and the bundled *adware* and *spyware* that has plagued users of the official clients, has driven the more tech-savvy users to alternative networks like Gnutella and *eDonkey*. Secondly, the lawsuits by the music industry have specifically targeted Kazaa hosts sharing over 1000 files, causing users to reduce the amount of shared content or to completely desert the network. Last, but not least, the music industry's tactic to introduce bogus files into the network, sometimes described as "polluting the pool", has had success in spoiling the file sharing experience. These files are offered by hosts under the control of firms like *Overpeer*, which specialize in anti-piracy solutions. The files contain either looped parts of the advertised song/movie or plain noise. Because of a design flaw in the FastTrack protocol, only parts of a file are hashed, allowing malicious nodes to advertise bogus files with the same hash as the respective original file. Because the clients download segments of a file from multiple sources, only one source has to be malicious to corrupt the downloaded file. Another problem is the increased efficiency of these firms in mimicking the appearance and behavior of "normal" file sharers [1].

It can be expected that the arms race between the P2P developers and the content distributors will continue in the near future. The latest releases of Gnutella and eDonkey clients block certain IP-ranges containing malicious nodes. As a next step, copyright holders might release modified versions of open-source clients that interfere with normal network operation. A possible answer for the Gnutella vendors would be to only accept connections to trusted clients, i.e. clients with a high reputation value.

3 The Gnutella Protocol

For this work, the Gnutella protocol has been chosen, since it is widely used and researched. It has been developed in an open-source process, making information about it easily available. Nevertheless, the algorithm that has been developed within this

[1] In the beginning, it was rather simple to identify malicious users and bogus files. User names were generated by a simple scheme (commonly used words concatenated by a two digit number) and the file names contained phrases like "no loops" or "real version". The host that offered the bogus file always had broadband connections. Now, all these give-away-clues have been eliminated.

work could be applied to any other P2P network using the *Ultrapeer* paradigm and relying on flooding for resource discovery (e.g. the FastTrack protocol).

The stable version of the Gnutella protocol is version 0.4, but this version is no longer in use in its pure, original form, since it has been shown that it does not scale up to bigger network sizes [19]. Due to the open nature of the protocol, with several independent clients using it, there is no strict standard adhered to by everybody. However, the basic features defined in the specifications for version 0.6 are widely adopted and will be followed in this work.

3.1 Basic Architecture of the Gnutella Network

The Gnutella network, often referred to as the *GNet*, is fully decentralized, i.e. there are no central servers. There are two types of nodes in the GNet, Ultrapeers and *Leafs* (see section 2.3 of the protocol definition in [18]). A Leaf is connected to several Ultrapeers (usually three) and does not have connections to other Leafs (Fig. 1). An Ultrapeer is a reliable, powerful node that handles most of the routing, so that the majority of nodes (Leafs) is not overwhelmed by the overhead of network organization. The number of Leafs that an Ultrapeer is connected to depends on the user's choice and the specific client used, and usually lies between 30 and 300 Leafs. Ultrapeers also maintain connections to several (usually five) other Ultrapeers (Fig. 1).

Ultrapeers use the *Query Routing Protocol (QRP)* to route requests for files. A file request is only forwarded to a Leaf that has been determined to be able to answer the file request. The QRP is based on hashing filenames of shared files, accumulating the hashes in a table (QRP table) and storing the tables at the Ultrapeer [20]. Therefore, an Ultrapeer can be seen as an indexing server for the connected leafs.

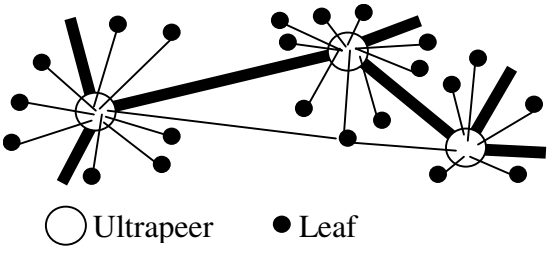


Fig. 1. The two types of nodes in the GNet are Ultrapeers and Leafs

The election of Ultrapeers is self-organized (see section 3.7 of the specifications [18]). There are several requirements that a node has to fulfill to be considered capable of becoming an Ultrapeer. The requirements state that it must not be located behind a firewall, have sufficient resources (CPU, RAM, and bandwidth), and a high uptime. An Ultrapeer-capable node can become an Ultrapeer, if there is a need for more Ultrapeers, i.e. there are only a few Ultrapeers with open connection slots.

3.2 Finding an Entry Point into the Network

To connect to the GNet, a node must know the IP address of at least one other connected node. This is done by accessing node addresses stored during a previous

successful run or by referring to a GWebCache. This is a script, which is located on a Web server and stores node addresses and URLs of other GWebCaches. If there are no node IP addresses and no working GWebCaches known to a node, it will not be able to make a connection to the Gnet (see sections 2.1 (Bootstrapping) and 3.2 (The Web Caching System) [18]).

3.3 Querying the Network

As described in section 2.7 [18], a node sends Query packets to connected nodes to find out who offers files with filenames containing certain search terms in the GNet. A Query packet contains a 16-byte long *Global Unique Identifier (GUID)*, a *Time-To-Live (TTL)*, and a *hops* value. Before a node forwards a Query packet, the TTL is decremented and the hops are incremented. If the TTL becomes zero, the Query will be dropped. Otherwise, it will be sent to the connected Ultrapeers and to the connected Leafs if the QRP demands it. A Query is never forwarded to the node that sent it. If a Query with the same search term and the same GUID has already been received, the duplicate is dropped. Normal values for the TTL are three or four, because this seems to be a good compromise between coverage and generation of network load.

If a Query is received by a node and the search terms match one or more files, a *QueryHit* is generated. The QueryHit contains the issuing node's IP address, matching filenames, the same GUID as the Query, a TTL that is one higher than the number of hops the Query took to reach the answering node and a hops value of initially zero. A QueryHit is treated similar to a Query, with the difference that it is only forwarded along the way the associated Query came. For this purpose, each node keeps a routing table, containing information about received Queries. Once a querying node receives a valid answer, it will negotiate the file transfer directly with the responding node determined by the IP address in the QueryHit packet.

4 Existing Incentive Schemes for P2P Networks

The goal of incentive schemes is two-fold: Firstly, the sharing of files should be encouraged; secondly, traffic costs are supposed to be reduced. We describe three of these schemes in this section. However, only the scheme that is the basis for our scheme is explained in more detail.

4.1 Credence P2P Network

Walsh & Sirer developed an incentive scheme for their P2P network in order to exclude malicious users [23]. The incentive scheme is based on a voting system. If a node needs to make a decision whether it can trust another node, it requests votes about the other node from the network. Based on the returned votes, the weighted average of votes is calculated. The weights are set according to experiences from earlier requests. This algorithm is based on a reputation system. It does not consider any kind of topology.

4.2 Maze P2P Network

Yang, Chen, Zhao, Dai, & Zhang developed a P2P file sharing application (called Maze) with a centralized search engine [26]. One of Maze' main characteristics is its

evolving incentive scheme. It is based on a set of incentive policies that are driven by user feedback from forums. Users get points awarded for uploading files and get points subtracted for downloads.

In detail, a new node gets 4096 points on its account P initially. Each Mbyte uploaded results in 1.5 points. The cost of a download is tiered, depending on the byte volume: 1 point per Mbyte within the first 100 Mbyte; 0.7 points per Mbyte between 100 and 400 Mbyte; 0.4 points per Mbyte between 400 and 800 Mbyte; 0.1 points for each Mbyte thereafter. Users with an account of less than 512 points get at most a bandwidth capacity of 300Kb/s. Otherwise, each request is ordered according to the value T = requestTime – 3 log (P). Requests with a top ranking get more resources. However, this has not been specified in the paper.

Besides, it has to be noted that the success of this scheme is influenced by the fact that users with a high value on their account get positive recognition (i.e. prestige, respect) within the online forum. The shortcoming of this incentive scheme is that it does not evaluate node/user behavior (e.g. offering mislabeled or corrupt files), it simply measures all file transfers.

4.3 The Club Approach

As introduced by Asvanund et al. [5], the club approach uses economic measures to create a more efficient network overlay topology for the Gnutella network. To reach these goals, they introduced the economic concept of a club. Clubs consist of one Ultrapeer and its connected Leafs and are described as “content-based, self-organizing communities of peers”.

A node selects a club depending on the utility that it gets when joining the club. Each node tries to maximize its utility by attempting to join clubs, which would provide the highest utility to them. To determine the net utility that a club delivers, the sum of the costs incurred by each club member is subtracted from the sum of all the utility added by each club member. The utility and the cost term are scaled by two node-dependant factors. A club only accepts a node’s connection request if this will lead to a higher club utility. Club utility is defined as the sum of utilities of each connected node in this club.

The utility that a node *y* provides to a node *x* has been defined as the similarity *sim* of the content of *y* to the past queries of *x*, multiplied by the weighted amount of content *y* shares and multiplied by the sum of the bandwidth of *y* and the weighted distance between *x* and *y*. The similarity function (sim) is based on information retrieval methods analyzing the filenames of the shared content. The distance and bandwidth are also scaled by two node-dependant factors. The distance term is a function that assigns high values to nodes located close-by in the underlying physical network topology.

The costs that a node *y* imposes on a node *x* are defined as the similarity of the past queries of *y* to the content of *x*, multiplied by the amount of content shared by node *x*, multiplied by the sum of bandwidth minus the distance between *x* and *y*. The similarity and distance functions are the same as used in the value function. This formula assigns higher cost to Leaf nodes whose information needs have a high probability of being satisfied, have high bandwidth, and are located on a different backbone.

Although, as described in [5][6], this approach yields a better network performance for nodes sharing content than the standard Gnutella protocol in version 0.6 (and version 0.4), this algorithm does not generate an optimal P2P topology. The sub-optimality is caused by the fact that no specific utility function has been defined for Ultrapeers. It does not specify how inter-club connections are evaluated nor made. The utility function itself does not differentiate between successful queries (i.e. queries that returned results) and unsuccessful queries (i.e. queries that did not return results). Since unsuccessful queries are much more likely to be reissued than successful queries, a club that can answer the former is more useful than one that can answer the latter.

5 Topology Formation Algorithm: IUTopForm

Our algorithm is based on the club approach described in the previous section. However, it differs significantly in five items, namely, the system for predicting future information demands; the reputation system; the utility function for Leaf nodes; the utility function for interclub connections; and the consideration of bandwidth.

5.1 Prediction of Future Information Needs

The first characteristic of our algorithm is that it uses two different ways to predict future information needs. First, the algorithm considers the unsuccessful queries issued by a node and compares them to the other node's shared content. The underlying assumption is that a user will repeat queries that had no results, since a user's past file needs that could not be fulfilled remain relevant in the future. Second, the algorithm predicts future informational needs by comparing a node's shared content to the other node's shared content. The underlying assumption here is that users have a certain "taste" in files, i.e. their future informational needs can be predicted by their past informational needs. For example, users that only listen to certain genres of music (e.g. classical music, movie soundtracks, etc) will have a certain number of identical files. Therefore, the users who share the same taste are more likely to answer each other's queries.

There is a limit to the needed similarity in content, though. If two users have the exact same set of files, they cannot satisfy each other's informational needs. All files of one node are already in the other's collection. Therefore, a target similarity that a node wants to achieve can be specified as well. The best value needs to be determined empirically in future research.

5.2 The Reputation System

With the recent success of P2P networks, the number of users who are selfish (e.g. freeriders, who adhere to the rules in a way that benefits them but hurts other users) or who are malicious (e.g. hackers or firms like Overpeer Inc. who intend to disrupt the network operations) has increased significantly. This behavior of users could be captured through a reputation system, like the ones described by Kung & Wu [14], Dutta, Goel, Govindan & Zhang [8], Aberer & Despotovic [3], Abrams, McGrew & Plotkin [4], and Lee & Hwang [10]. Our topology formation algorithm assumes a reputation system that assigns higher reputation values to cooperative (i.e. not selfish

and not malicious) nodes. Those nodes earn reputation points by providing services to other nodes. How reputation is gained, where it is stored, and how it is verified is transparent to the topology formation algorithm introduced here. A node's reputation is an element of the utility function and is therefore taken into account for each utility calculation. Trustworthy nodes will more likely encounter other trustworthy nodes in a club they are part of than untrustworthy nodes. This also leads to the aggregation of untrustworthy nodes in their own clubs, minimizing their negative impact on the network.

5.3 Utility Function for Leaf Nodes

Our algorithm uses the following utility functions for making decisions about joining or leaving a club:

$$\text{LeafU}_x(\text{club}) = \Sigma_{y \in \text{club}} U(x, y) + a_1 \Sigma_{k \text{ connected to club}} \text{InterclubU}(\text{club}, k) \quad (1)$$

$$U(x, y) = a_2 |\text{files}(y)| (1 - |\text{sim}(\text{files}(x), \text{files}(y)) - \text{target}|) + a_3 \text{sim}(\text{unsucessfulQueries}(x), \text{files}(y)) + a_4 \text{distance}(x, y) + a_5 \text{reputation}(y) \quad (2)$$

The formulas (1) and (2) state that the evaluating Leaf *x* calculates the utility it gains from all nodes *y*, which are members of the club, and the utility it gains through the interclub connections. The variables a_1 to a_5 are weights that determine the relative importance of the different elements of the utility function. The function *files(z)* represents the content of a node *z*. The similarity function *sim()* is an information retrieval method as described in (Asvanund et al. [5]) and results in a value between zero (not similar) and one (similar). It operates on two sets of files by comparing every element of the first set to every element of the second set and returns the average similarity value. The variable *target* is defined (as described above) as the target similarity between the file collections that a node wants to achieve. The function *InterclubU()* will be described in the following paragraph.

5.4 Utility Function for Interclub Connections

Our topology formation algorithm considers utility function (3) to determine the utility that a club gains by making a connection to another club. Since the establishment of a connection between clubs follows the Gnutella 0.6 protocol specifications, the connection can only occur between Ultrapeers. Therefore, the Ultrapeer's properties are the determining factors in the decision to open a connection. The so-called Interclub utility is defined as:

$$\text{InterclubU}(c, k) = (b_1 \text{ numberOfLeafs}(k) + b_2 \text{ files}(k)) + b_3 \text{ distance}(c, k) + b_4 \text{ reputation}(k) \quad (3)$$

The variables b_1 to b_4 of the utility function (3) are weights, similar to a_1 to a_5. The utility function has four elements. First, the number of connected Leafs, which is an

estimate for the club k's likelihood of fulfilling club c's informational needs. Second, the amount of content *files(k)* offered by club k, which is also an indicator for the likelihood of fulfilling club c's informational needs. Third, the distance of the two clubs' Ultrapeers *distance()*. Since the distance function is a part of the Leaf utility function, Leafs are likely to be close to the Ultrapeer in the underlying physical network structure as well. Therefore, the distance between the Ultrapeer will give a good estimate of the average distance between the two club's nodes. The last and most important element is the club's reputation. Since, as described above, the Leaf utility function leads to an aggregation of trustworthy nodes in trustworthy clubs and untrustworthy nodes in untrustworthy clubs, this element of the interclub utility function leads the evaluating club to connect to a club with the same amount of trust as itself.

Summarizing the purpose of the inter-club utility function, it can be stated that the connected Ultrapeers will most likely be able to answer the club's queries, be close to the club's nodes, leading to fast response and download times and be trustworthy. This should lead to a very positive overall effect on the whole network, at least for cooperative nodes.

5.5 Disregarding Bandwidth

Estimating a node's bandwidth is inherently difficult and costly. For testing the actual bandwidth of a node, the link needs to be filled with traffic of some form, even if it is only for a short time. Therefore, most P2P applications allow the user to enter his connection type and use this information to estimate the available bandwidth. This, however, is very inaccurate and can be easily misused. Selfish users given the choice of setting the connection type will enter a connection type that gives them the highest benefit, e.g. DSL users who pay for byte volume might enter modem as the connection type to limit the cost for traffic. Because of these issues, our topology formation algorithm disregards bandwidth as a factor. It gives selfish users fewer possibilities to gain unfair advantages.

6 Simulation of a GNet

In order to investigate the performance of our algorithm compared to the original club algorithm, we programmed a simulation of a Gnutella network. For this, we took the simulator used in (Asvanund et al. [5] and Bagla & Kapadia [6]) as a reference implementation. It was generously provided by Ramayya Krishnan. The new version of the simulator, enhanced to be able to run our algorithm, is written in Java and based on the simulation framework J-Sim (Tyan [21]). Additionally, the open-source packet "Java 2D Graph" has been used for some minor calculations [7]. For logging purposes, log4j of the Apache Software Foundation is used [15].

The simulator has two parts, one for the creation of a scenario, the other for running the simulation of the previously created scenario. The split in two applications has the advantage that created scenarios can be reused and exchanged between users. It also allows a manual analysis and manipulation of the created scenarios. The scenario application creates a set of input files. The simulation application reads them in and

then starts a simulation run, executing the events as specified in the scenario files. Each simulation run is split into several rounds, as specified in the input files. A round consists of two phases, the evolution phase, in which Leafs and Ultrapeers make new connections and discard old ones, and the query phase, in which the nodes issue Queries for files they are interested in and return QueryHits. The simulator does not model file transfers, only the topology issues and querying. The reason is our assumption that good performance in the resource discovery indicates good performance in the file transfers as well. If many well-trusted locations of a file can be found and they respond with a low latency, then the download of that particular file should be reliable and fast. Therefore, after the query phase, it is assumed that all files for which locations have been found are being downloaded completely and added to the node's file collection. Whether those files will be shared in the next round depends on the node's settings. The simulation results in two datasets with statistics, one for each node per round and one for the whole simulation.

6.1 Creation of Simulation Scenarios

A simulation scenario description consists of text files describing nodes, files, events, and some global parameters. They are generated according to the values given as arguments to the scenario creation application.

The `nodes.txt` file contains information about each node. The most important is the node ID, a global identifier, the files initially shared by the node, information whether the node starts as an Ultrapeer, can become an Ultrapeer during the simulation, and whether it is a freerider or shares files. As described in Bagla & Kapadia [6], a statistical analysis of Gnutella network traffic has shown that 42% of nodes never share files. Therefore, the 42% of all nodes are not assigned files and labeled freeriders in our simulation. The remaining nodes are assigned a certain number of files depending on a long right tail distribution with an average of 270 files. This is modeled by a Weibull distribution with a shape factor of 0.576. The nature of this function leads to the situation that a node can start with zero files without being a freerider. In this case, the node will share any files that it acquires after a query phase. The file `nodes.txt` also contains the TTL that the node's Queries have and a set of coordinates. These coordinates determine the node's position on the network grid, which simulates the underlying physical network structure. The number of club and interclub connections a node can keep open is also specified here. Another important factor is the initial reputation, which is a random number equal to or smaller than the number of files shared. In addition to this, the file `nodes.txt` contains a list of file ID numbers. In the simulator, the similarity between two files is defined in terms of the distance of their ID numbers. Two files with IDs 15 and 16 are very similar, 2 files with IDs 15 and 1600 are not similar. Also, a number of file seeds is chosen, which is normally distributed with a mean of three. The file seeds represent the interest of users for a file type (e.g. type of music). Then, the file IDs are drawn according to a normal distribution around the file seed IDs, until the targeted number of shared files has been reached.

The `files.txt` file describes the content files existing in the modeled Gnutella network. Each content file has a file ID and a size. The maximum number of existing files is defined by a constant, which is a factor that is multiplied by the number of

existing nodes. The factor is set to 80 files per node. If a higher number is chosen, the average similarity between two sets of files will be lower, because the IDs of files that two nodes own are randomly chosen from a greater range and the similarity depends on the distance of the file IDs.

The `events.txt` file describes the events that will be sent to nodes in the network. There are two types of events, Evolutions and Queries. An Evolution event causes a node to renegotiate its connections. A Query event specifies the time within the simulation when it is issued and the file ID that the issuing node is looking for. Which file IDs are queried depends on the files a node has. All queried file IDs are normally distributed around IDs the node already owns. If the node shares no files, a random existing file ID is chosen as a file interest and the queried IDs are normally distributed around this file ID. The number of queried files follows a long right tail distribution with a mean of 13, as found in the analysis in Bagla & Kapadia. This is also modeled by a Weibull distribution with a shape factor of 0.576.

The `topology.txt` file contains the initial topology when the simulation starts. The file is empty in this version of the simulator, because the simulation starts with an evolution phase. It is nevertheless possible to manually specify a topology, which would be honored by the simulator.

The `globals.txt` file is used for global system variables, such as finish time and network grid size. Network grid size describes the size of the simulated physical network. Another variable defines how many Ultrapeers are allowed in the system. For our simulations, the value is set to 110%, allowing 10% more Ultrapeers than necessary. This way, all Leafs have the chance to use the maximum number of Ultrapeer connections. This leeway also allows for some competition between Ultrapeers. Ultrapeers who are attractive to nodes will be able to fill their open Leaf connection slots, whereas Ultrapeers who are not attractive will not find Leafs to fill their slots. The attractiveness of an Ultrapeer is determined by its Leaf and interclub connections.

6.2 Running the Simulator

The simulator allows specifying the utility model and whether or not to allow *Ultrapeer Status Transitions*. The utility model can be either `CMU` (Bagla & Kapadia [6], Asvanund et al.[5]) or `IU`. The switch *Ultrapeer Status Transitions* determines whether eligible Leafs can become Ultrapeers and vice versa, if the network needs more Leafs (i.e. there are Leafs who cannot find a club to join) or Ultrapeers (i.e. there are Ultrapeers who cannot find Leafs to join their club). However, a node will only change from or into an Ultrapeer if it could not establish these connections for two rounds.

6.3 The Evolution Phase of the Simulation

In the evolution phase, the nodes renegotiate the network topology. Every connection negotiation uses a three-way handshake as shown in Fig. 2. Each node will attempt to connect to a certain number of Ultrapeers per round, three if the node is a Leaf and five if it is an Ultrapeer. These numbers have been chosen according to the GNet specification.

An active node picks a random Ultrapeer to start connection negotiations. There are two different ways how the connection negotiations are performed. It depends on the type of node. If the node is an Ultrapeer, it will first send a *Connection Request* to another Ultrapeer. If the contacted Ultrapeer has a free connection slot for another Ultrapeer, it will send an *Invitation Message* to the Ultrapeer in question, otherwise it will only send an *Invitation Message* if the replacement of the weakest connected Ultrapeer results in an increased sum of interclub utilities. The active node receiving the *Invitation Message*, decides in the same way to send or not to send a *Confirmation Message*. When a *Confirmation Message* is sent, the sending Ultrapeer will add the receiving Ultrapeer to its list of connected Ultrapeers and the receiving Ultrapeer will add the sending Ultrapeer to its list of connected Ultrapeers. If there are no free Ultrapeer slots, a *Connection Close Message* will be sent to the weakest connected Ultrapeer. In general, if the situation has changed, a *Connection Close Message* can be sent anytime during the protocol.

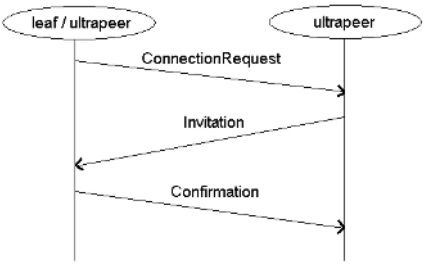


Fig. 2. The connection negotiation uses a 3-way handshake

If the active node is a Leaf, it will send a *Connection Request* to the Ultrapeer. If the contacted Ultrapeer has a free connection slot for a Leaf, it will return an *Invitation Message*. Otherwise, it will only send an *Invitation Message* if replacing the weakest connected Leaf results in a higher club utility. This means that all possible club configurations including the new Leaf and excluding one of the connected Leafs have to be evaluated (i.e. calculating the club utility). The Leaf receiving the invitation will send a *Confirmation Message* if the utility it would gain from this club is higher than the lowest utility of connected clubs or if it has a free connection slot. Otherwise, it will send a *Connection Close Message* to the Ultrapeer. If it accepts, it adds the Ultrapeer to the list of connected Ultrapeers, possibly closing the connection to a weaker club. Once an Ultrapeer has been added, it sends a list of its shared files to the Ultrapeer, so that the information can be used in subsequent connection negotiations. This list is also needed to simulate the Query Routing Protocol of the Gnutella 0.6 protocol.

6.4 The Query Phase of the Simulation

After the topology for the round has been negotiated during the evolution phase, the nodes will send Querys as specified in the Simulation Scenario. The querying process is handled as defined by the Gnutella protocol. Additionally, a reputation system is modeled in a simplified way. Nodes earn one reputation point by forwarding a query

hit and lose five reputation points by generating them. It is assumed that a mechanism exists to prevent cheating. Therefore, all simulated nodes act completely honest. The information about issued and forwarded queries and query hits is stored and used to calculate the metrics at the end of the simulation.

7 Analysis of the Simulation Results

To prove that our algorithm, IUTopForm, provides more incentives for users to cooperate than the original club algorithm, we compared them using the same scenario, with the same settings. We created a scenario with 1000 nodes, 89 of them being Ultrapeers, the others Leafs. In a simulated period of 100,000 time units, 5 rounds are completed, each starting with an evolution phase and ending with a query phase. Each node attempts to become member of three clubs, each club has a maximum of 31 Leafs and one Ultrapeer as members. Each club is connected to up to five other clubs. Query messages are sent with a TTL of four. The targeted similarity level between two file collections is set to 80 percent.

In order to show the effectiveness (close proximity to the requested content in the overlay topology and a large distance to untrustworthy nodes) of our incentive scheme (expressed through the utility functions), we consider the distribution of reputation values of nodes. Since nodes earn reputation by providing services to other nodes (i.e. cooperating), the average reputation value is used to distinguish between cooperative and non-cooperative nodes. From a node's average reputation value, it can be determined whether a node is cooperative or not.

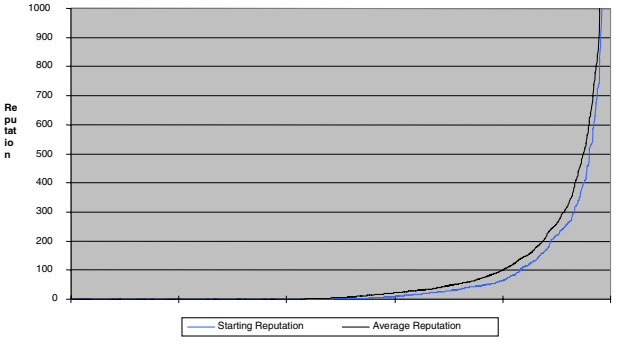


Fig. 3. Reputation distribution

Fig. 3 shows the starting and average (final) reputation value of nodes that belong to five different groups. All nodes are sorted out according to their reputation value and, then, split into five groups. Each group represents a quintile of the entire set of nodes. Initially, all nodes have been given a starting reputation value, according to a long tail distribution. Consequently, the first three quintiles have very low reputation values. Only the last two quintiles can be seen as partially and, respectively, fully cooperative. The average (final) reputation value, which represents the reputation value of a node

within a quintile at the end of the simulation, changed only slightly compared to the initial value.

Close Proximity of the Requested Content. To demonstrate that requested content is closer for cooperative nodes than for uncooperative nodes, we measure how many of the received query hits will be answered from nodes of the club of which the requesting node is a member.

As Fig. 4 illustrates, our algorithm achieves that cooperating nodes get better performance than using the CMU algorithm. Fully cooperating nodes clearly get an improvement in response rates. They participate in clubs, which can satisfy their file needs. All other nodes face lower performance if our approach is applied. Those nodes are penalized for not cooperating sufficiently. Since cooperation is the only way to improve bad performance, our approach forces peers to increase cooperation or live with even worse response rates than in the CMU approach.

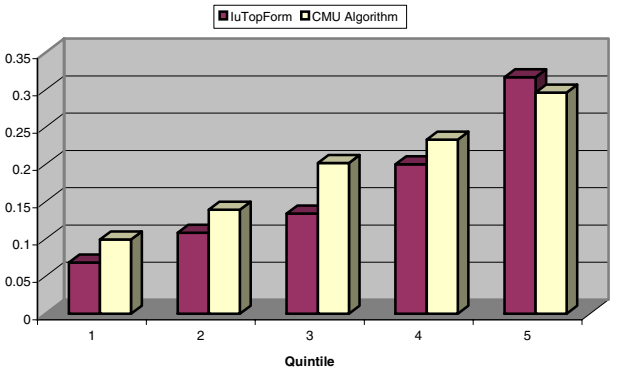


Fig. 4. Number of positive responses from directedly connected clubs

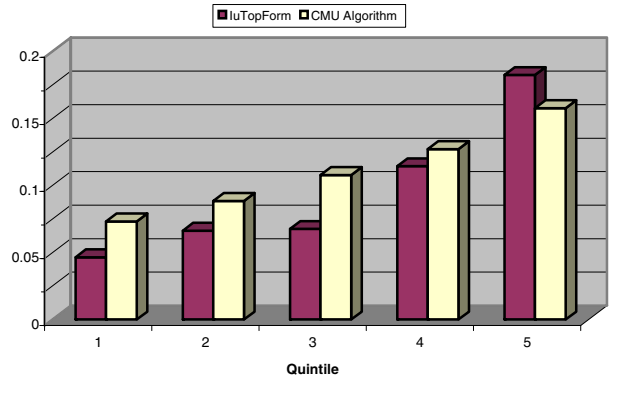


Fig. 5. Number of positive responses from all clubs

If one considers not only the connected clubs but also all clubs that a node can reach, a similar picture develops (Fig. 5). The weaker four quintiles receive worse service, receiving fewer responses to their requests. Cooperating nodes perform better under the IUTopForm approach, i.e. the strongest quintile of nodes benefits from our incentive scheme.

Large Distance to Untrustworthy Nodes. The metric that is used to describe the distance of a node to another node is the reputation difference. The reputation difference is calculated by subtracting the average reputation value of all connected nodes from the node's own reputation value and, then, taking the absolute value of the result. Table 1 shows the simulation results for this metric for all three types of connections between nodes.

Table 1. Our algorithm homogenizes the topology's reputation distribution

Average Reputation Difference between …	IUTopForm Approach	CMU Approach
A Leaf and its Ultrapeers	135.685	150.671
A Ultrapeer and its Leafs	106.735	112.430
A Ultrapeer and its connected Ultrapeers	77.081	134.471

As listed in Table 1, this metric shows a decrease in reputation difference for all possible connection types: A clear 11% decrease between a Leaf and its Ultrapeers, an indecisive shift of 5% within clubs, and a major decrease of 74% between Ultrapeers.

The situation within clubs remains the same, because the original algorithm already took the number of shared files into account. Since this number is roughly proportional to the reputation in this simulation. However, the introduction of the reputation term into the utility function did yield significant improvements for the other interclub connections.

8 Conclusion

Within this paper, we introduced a new incentive scheme that is used for a new, market-managed topology formation algorithm. Our algorithm, which has been inspired by the approach of Asvanund et al., adds the additional dimension of reputation to the incentive scheme. In addition to this, our topology formation algorithm incorporates a new model for predicting future information demands, argues not to include the consideration of bandwidth, and proposes two new utility functions (i.e. a utility function for Leaf nodes and a utility function for interclub connections).

The simulation results show that nodes with similar reputation value are close to 0065ach other in the topology of the P2P network (those nodes are clustered within the same club). Our market-managed topology formation algorithm has had the effect that nodes with a high reputation value receive better service than under the algorithm of Asvanund et al.. This shows that our algorithm forces users either to cooperate or to tolerate even reduced file-sharing performance.

References

[1] G. Eason, B. Noble, and I. N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955.

[2] Abdul-Rahman, A. and Hailes, S., “Supporting Trust in Virtual Communities”, In *IEEE Proceedings of theHawaii International Conference on System Sciences,*Maui, Hawaii, January 4-7, 2000.

[3] Aberer, K. & Despotovic, Z., “Maximum Likelihood Estimation of Peers' Performance in P2P Networks,” EPFL - Swiss Federal Institute of Technology, 2004.

[4] Abrams, Z., McGrew, R. & Plotkin, S., “Keeping Peers Honest in EigenTrust,” Stanford University, 2004.

[5] Asvanund, A., Bagla, S., Kapadia, M., Krishnan, R., Smith, M., & Telang, R., “Intelligent Club Management in Peer-to-Peer Networks,” Carnegie Mellon, Heinz School of Public Policy and Management & Information Networking Institute, 2003.

[6] Bagla, S. & Kapadia, M. H., “Peer-To-Peer Self-Organizing Communities,” Carnegie Mellon University, Information Networking Institute, 2003.

[7] Brookshaw, L., “Java 2D Graph,” Retrieved on September 28th, 2004, from http://www.sci.usq.edu.au/staff/leighb/graph/source/SpecialFunction.java, 1996.

[8] Cohen, B., “Incentives Build Robustness in Bittorrent,” In 1st Workshop on Economics of Peer-to-Peer Systems, June 2003.

[9] Dutta, D., Goel, A., Govindan, R. & Zhang, H., “The Design of A Distributed Rating Scheme for Peer-to-Peer Systems,” University of Southern California / Stanford University, 2003.

[10] Feldman M., C. Papadimitriou, J. Chuang, & I. Stoica, “Free-Riding and Whitewashing in Peer-to-Peer Systems,” ACM SIGCOMM04 Workshop on Practice and Theory of Incentives in Networked Systems (PINS), August 2004.

[11] Hee Lee, C. & Hwang, J, “Agent-based Modeling for Differentiated Admission in P2P Systems Using Evolutionary Game Theory Focused on Ownership Reputation,” Seoul National University / Syracuse University, 2004.

[12] Jennings, T., “Fido and FidoNet,” Retrieved on August 19th, 2004, from http://www.wps.com/FidoNet/ .

[13] Kirk, P., “Gnutella 0.6 - Defining a Standard,” Retrieved on August 19th, 2004, from http://rfc-gnutella.sourceforge.net/developer/index.html, 2003.

[14] Kung, H.T. & Wu, C., “Differentiated Admission for Peer-to-Peer Systems: Incentivizing Peers to Contribute their Resources,” Harvard University/ Academia Sinica, 2003.

[15] log4j project, “Logging Services - log4j,” Apache Software Foundation. Retrieved on September 28th, 2004, from http://logging.apache.org/log4j/docs/ , 2003.

[16] McManus, S., A short history of file sharing. Retrieved on August 19th, 2004, from http://www.sean.co.uk/a/musicjournalism/var/historyoffilesharing.shtm , 2003.

[17] Mello, J., “File Sharers Deserting Kazaa's FastTrack Protocol,” Retrieved on August 18th, 2004, from http://www.technewsworld.com/story/34305.html, 2004.

[18] RFC-Gnutella 0.6, “RFC-Gnutella 0.6,” Gnutella Developers Forum. Retrieved on August 19th, 2004, from http://rfc-gnutella.sourceforge.net/developer/testing/index.html , 2004.

[19] Ritter, J., “Why Gnutella Can't Scale,” No, Really.. Retrieved on Aufust, 26th, 2004, from http://www.darkridge.com/~jpr5/doc/gnutella.html , 2001.

[20] Rohrs, “Query Routing for the Gnutella Network,” Retrieved on August 26th, 2004, from http://rfc-gnutella.sourceforge.net/src/qrp.html , 2001.

[21] Tyan, H., "Design, Realization and Evaluation of a Component-Based Compositional Software Architecture for Network Simulation," Ohio State University, 2002.
[22] Usenet History, "Usenet Software: History and Sources," interbulletin.com. Retrieved on August 19th, 2004, from http://news.interbulletin.com/usenet_his.html.
[23] Walsh, K., & Sirer, E.G., "Fighting Peer-to-Peer SPAM and Decoys with Object Reputation," Proceedings of the Third Workshop on the Economics of Peer-to-Peer Systems (p2pecon), Philadelphia, USA, 2005.
[24] Wikipedia, "Gnutella," Wikipedia. Retrieved on August 19th, 2004, from http://en.wikipedia.org/wiki/Gnutella , 2004.
[25] Wikipedia, "Peer-to-peer," Wikipedia. Retrieved on August 18th, 2004, from http://en.wikipedia.org/wiki/Peer-to-peer, 2004.
[26] Yang, M., Chen, H., Zhao B.Y., Dai, Y., & Zhang, Z., "Deployment of a Large-Scale Peer-to-Peer Social Network," WORLDS04, 2004.

Adapting a Captive Portal to Enable SMS-Based Micropayment for Wireless Internet Access

Jaume Barceló, Miquel Oliver, and Jorge Infante

Network Technologies and Strategies Research Group, Universitat Pompeu Fabra,
Passeig de Circumval.lació 8,
08003 Barcelona, Spain
Tel.: (+34) 93 542 29 42; Fax: (+34) 93 542 24 51
{jaume.barcelo, miquel.oliver, jorge.infante}@upf.edu

Abstract. This paper introduces a micropayment mechanism suitable for Wireless Internet Access Providers. It is proposed that the users obtain the credentials that allow them to surf the web after sending a Premium-rated SMS, thus avoiding a direct payment relationship between the user and the WISP. Mobile users are familiar with Premium SMS and consider them a secure and convenient payment method, because of the existing trust relationship between users and Mobile Network Operators. A third party named SMSBroker that acts as intermediary between the MNO and the WISP is also required in practice. The concept has been implemented and tested in a real wireless access network.

1 Introduction

With the development of low cost hardware for wireless networking based on IEEE 802.11b/g technology, many access points are deployed in cities around the world and wireless access has already become an ubiquitous way to connect to the Internet.

At present, second and third generation mobile networks offer connectivity with well-defined authentication and authorization procedures for a large customer base, but at lower speed and higher costs than WiFi Networks. In this article we propose a Wireless Internet Service Provider (WISP) that takes advantage of the billing relationship between a user and a Mobile Network Operator (MNO) to charge for the offered service. Our scheme is based on the use of Premium-rated Short Message Service (PSMS).

2 Micropayment

A micropayment system has to allow the payment of small quantities (up to 1 Euro) for digital goods and services. [1] identifies the key characteristics determining the success of a micropayment schemes as trust, ease of use, pervasiveness and transaction speed.

B. Stiller, P. Reichl, and B. Tuffin (Eds.): ICQT 2006, LNCS 4033, pp. 78–89, 2006.

Many Internet-based micropayment solutions exist and [2] provides an extensible way to embed in a web page all the information necessary to initiate a micropayment (amounts and currencies, payment systems, etc).

Although this solution will allow content providers to charge for their Internet digital goods, it might not not be the best solution when the service to be sold is the Internet access, since an Internet-based micropayment requires an already working Internet connection to succeed.

The mechanism that we have adopted in our solution involves the sending of a *premium rate* (overpriced) SMS by the customer. *Premium rate* SMS (PSMS) payments started appearing in 2001 and have since then become a norm for spontaneous payments in conjunction with TV and Radio broadcasts. Recently they have been incorporated as an additional marketing channel that permits user interaction and instantaneous feedback.

The success of phone-based payments is based on the following factors:

- No additional registration required for the user: any mobile phone owner with access to a telecom network with premium services could make payments through this medium. Users already have trusted billing relationship with their MNO, either using a prepayment card or a monthly bill.
- Ease of use, the user needs only to send an SMS.
- Pervasive. Almost everyone has a mobile phone handy and is familiar with the sending of SMS.
- The transaction speed, usually between one an two seconds, is perceived as fast by human users.

The main drawback of SMS-micropayments is the high transaction cost. If a customer pays about 1 Euro for a good or service, the merchant receives less than the half of it. This price overhead makes this solution useful only for occasional or impulse users. Other market segments, including frequent intensive users might be charged using different payment schemes, such as a flat monthly bill.

3 Money and Message Flow

Fig. 1 introduces the actors involved in our proposal. The first actor is the user and needs both a GSM/UMTS terminal and a WiFi enabled terminal. The second element is the Mobile Network Operator (MNO) that owns the Radio Access Network and a Core Network.

The SMSBroker acts as a gateway between the MNO network and the Internet. Actually it is connected to all the MNO operating in the country, thus making it possible to send and receive messages from any user. The SMSBroker has a SMSGateway that parses SMS coming from the MNO interfaces and resends them as HTTP requests to the Access Server of the WISP. The WISP has deployed a number of Access Points that form the Access Network, separated from the Internet by a dynamic firewall known as Access Server.

Fig. 1 also shows how the money is divided between the actors. The values have been taken from the Spanish case and might differ in other countries. However, they can be used as a reference for the European market.

PSMSs have fixed prices:0.15, 0.30, 0.60 and 0.90 Euro. Conversations with local WISP lead us to the conclusion that they were willing to offer 20 minutes of Internet access for 0.40 Euro. This means that the final user has to be charged 0.90 Euro, since less than 50% of the PSMS price reaches the WISP.

The user pays 0.90 Euro by sending a PSMS. This amount of money plus VAT (16% in our country) is going to be charged to the user's prepaid card or in a monthly bill. The MNO retains about 0.30 Euro and pays the rest to the SMSBroker which in turn pays about 0.40 to the WISP.

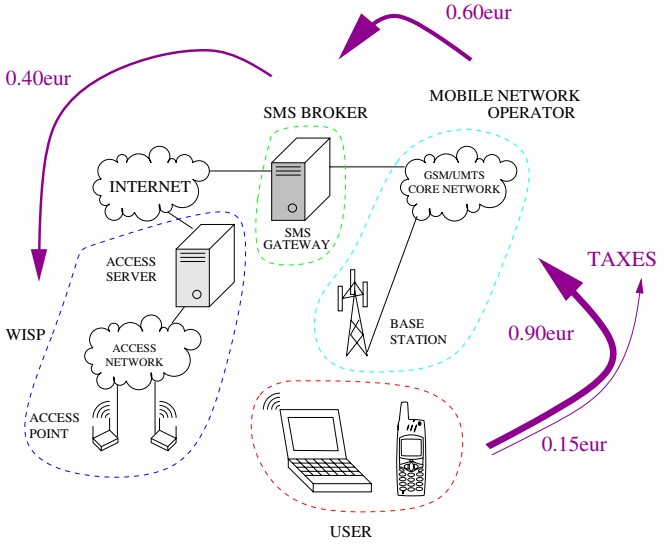


Fig. 1. Actors involved in a SMS-Enabled Internet access and the money flow among them

The purpose of the next figure (Fig. 2) is to clarify the message interchange between the different entities that finally grants user access to the Internet.

Message labeled one (1) in the figure is the initial PSMS that the user sends. The MNO forwards it to the SMSGateway. The SMSGateway generates a HTTP request including the user's mobile phone number (Mobile Subscriber Integrated Services Digital Network number, MSISDN) as a parameter.

A servlet in the Access Server receives the HTTP request. It stores the MSISDN – or an MD5 hash of the MSISDN if privacy is a concern – together with a randomly generated 4-digit numerical password. A welcome message and the password are sent as an answer to the HTTP request.

This answer is labeled as two (2) in the figure and reaches the SMSGateway that converts it into an SMS and sends it to the user through the MNO.

Now the user has the password and turns on a laptop that gets networking configuration using DHCP [3] and opens a browser. A form appears on the screen asking for credentials and the user fills it in using the mobile's phone number and the received password. This is labeled as three (3).

After the authentication and authorization procedure, the user is allowed to browse the web for 20 minutes, labeled as four (4).

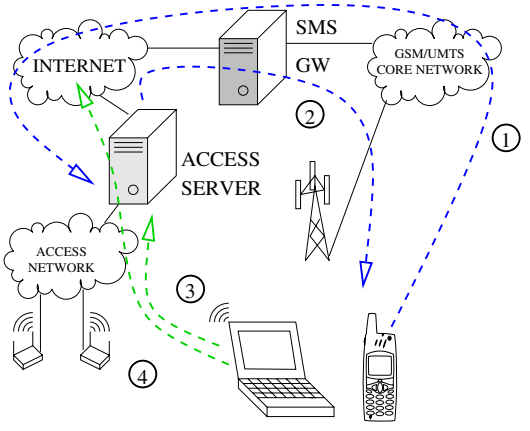


Fig. 2. Messages interchanged to grant Internet access to the user

The user does not need to install any special software, since the configuration and authentication process is performed using well known protocols such as DHCP and HTTPS.

4 PSMS and SMSBroker

Premium rate SMS is offered by many MNO in Europe. The operator has a monthly billing relationship with millions of users. A merchant can purchase a phone number from the operator worth, for example, 0.90 Euro. Every time a user sends an SMS to that number, 0.90 Euro will be added to that user's phone bill and the MNO will pass through a subset of that money to the merchant.

The merchant probably would need to obtain *the same* number from all MNO operating in the country, and additionally would need direct connection to every MNO network. This can be unaffordable for a medium or small business.

The solution to this situation is to take advantage of a third party called SMSBroker. This actor obtains a number associated with given PSMS price from the different MNO, the same number from all MNO. This number can be used to offer different services from different providers, sharing the costs. In our case, the number is *7212* and the first parameter (word) in the message has to be *upf*. The number together with the first parameter identify uniquely a service provider.

Using a SMSBroker removes the requirement of the direct connection of the service provider with all the MNO networks. Actually, the server provider only needs a connection to the Internet. This is because the SMSBroker connects to the MNOs and provides a SMSGateway that converts PSMS messages in HTTP requests that then are forwarded through the Internet.

As explained before, the destination number of the message together with the first parameter identifies a service provider, that is, a host name and port number where the PSMS are going to be forwarded.

The answer to the HTTP request originated by the SMSGateway can be a text SMS, but also a logo or a ringtone. The special code *zero vertical slash* (0|) have to be attached at the beginning of any answer to indicate to the SMSGateway that the content should be interpreted and resent as a text message.

5 Access Server

The Access Server performs mainly two tasks: Answering PSMS messages forwarded by the SMSGateway and authenticating the users controlling the access to the Internet. To accomplish the second task, the *NoCat captive portal* has been used.

5.1 Captive Portal

When providing public Internet access, users must be securely identified when they connect, and then allocate only the resources they are entitled to. The built-in security features of 802.11b are designed to create a private network with trusted clients but they aren't well suited for public-access networks. According to [4] a captive portal is a router or a gateway host that will not allow traffic to pass before the user is authenticated. It is essentially a mechanism to prevent users from accessing network resources (usually Internet access) until they have authenticated with a server. Typically a captive portal is used at wireless hotspots. allowing the user to log in, authenticate and use the network according their privileges. The users do not need to know a particular address to authenticate. Whenever unauthenticated users attempt to browse, they are transparently redirected to the authentication page. Two of the most well known open source implementations of the captive portal concept are WifiDog [5] and NoCat [6] [7]. The first one is a fully embeddable solution that can run on the Access Point. The second is written in Perl and needs two Linux servers to run: NoCatGateway and NoCatAuth

In our approach, the NoCat captive portal have been modified to host the desired functionality of allowing Internet access only to those users that send a PSMS message to pay for the service.

The NoCatAuth is implemented as CGIs running on the Apache web server and is in charge of the following tasks:

- Presents the user with a network login prompt via an SSL-protected Web page.
- Verifies user credentials.
- Securely notifies the wireless gateway of the user's status, and authorizes further access.

On the NoCatGateway side, the software

- Manages local connections.
- Sets bandwidth-throttling and firewall rules modifying the Linux *iptables*.
- Times out old logins after a user specified time limit.

The system was designed to preserve trust. The gateways and end users only need to trust the Auth system, which is secured with a registered SSL certificate. Passwords are never given to the wireless gateway (thus protecting the users from any malicious node owners), and gateway rules are modified only by a cryptographically-signed message from the Auth system, protecting the gateway from users or upstream sites trying to spoof the Auth system. To make this possible, the Auth Server's public PGP has to be distributed among the gateways.

The Connection Process. The connection process involves several phases that are detailed in Fig. 3:

1. Redirect. The users in the WISP coverage area are immediately issued a DHCP lease.All access beyond contacting the Auth service is denied by default. When users try to browse the Web, they are immediately redirected to the gateway service, which then redirects them to the Auth system's SSL login page (after appending a random token and some other information to the URL line).
2. Connect Back. Once the user has logged in correctly, the Auth system then prepares an authorization message, signs it with PGP, and sends it back to the wireless gateway. The gateway has a copy of the Auth service's public PGP key, and can verify the authenticity of the message. Since part of the data included in the response is the random token that the gateway originally issued to the client, it's very difficult to fake out the gateway with a replay attack. The digital signature prevents the possibility of other machines posing as the Auth service and sending bogus messages to the wireless gateway.
3. Pass Through. After message verification, the NoCatGateway modifies its firewall rules to grant further access, and redirects the users back to the site they were originally trying to browse to.

To keep the connection open, a small window is opened on the client side (via JavaScript) that refreshes the login page every few minutes. Once the user moves out of range or closes the *renewal* pop-up window, the connection is reset and requires another manual login.

Authorization Sources. NoCatAuth admits several authentication sources, including password files, *Radius*, *MySQL* database and the flexible *Pluggable Authentication Modules* (PAM). We decided that the database method was the most convenient for our purposes because it allowed the inclusion of time-management information and could be easily accessed both by NoCat and by the servlet that

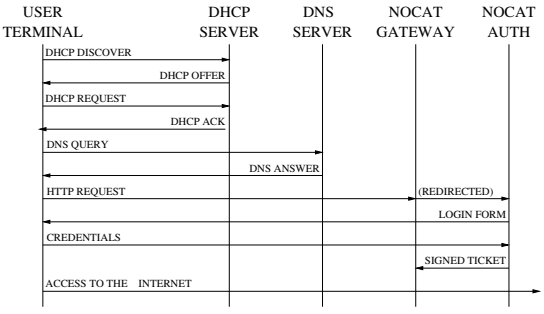


Fig. 3. Connection process for the NoCat captive portal

attended the messages sent by the users. We used the database scheme included in NoCat as a starting point.

The most relevant table is called *member* and stores the user information. It contains the following fields: *url*, *description*, *created*, *modified*, *status*, *login*, *pass* and *name*. The most important are *login* and *password*, that contain the credentials that a user has to present for identification.

5.2 Modifications to the Original NoCat

In order to implement the time-management information to control that the user receives 20 minutes of Internet access for each PSMS sent, the NoCat software and database scheme required some modifications. Two more fields were added to the *member* table mentioned above.

Changes to the Database. One field called *MinutesLeft*, that stores the credit (in minutes) of each user was included. When sending a PSMS, the user gets 20 minutes credit, and can increase this credit in 20 more minutes by sending another PSMS.

Another new field *pass_clear* contains the password in clear text and it is used to remember the password to the users whenever they send a PSMS to increase their credit. This passwords consist on four-digits numbers. It is obvious that these are not strong passwords, but convenience prevailed over security. It more important that these passwords are easy to remember and type. Therefore a design decision was made to make them similar to the Personal Identification Numbers (PIN) already used in mobile phones.

A table called *history* was added to the database. A user that exhausts the credit is deleted from the *member* table and all the information related to that user is moved to the *history* table. Finally, another table was included to store all the PSMS received together with the sender's number.

Renewal Window. NoCat requires that every user maintains the session (and thus the firewall) open by means of a *renewal window* (See Fig. 4). This is actually a browser window opened using JavaScript at login time that contains a form

with the user information and refreshes automatically periodically, notifying the gateway that the user is still active.

The renewal period was chosen to be 60 seconds, and the event is used to decrease in 1 the credit in minutes of the user. The renewal window is used to inform the user of the credit left, using a JavaScript counter that shows minutes and seconds. The same window provides information to the user about how to increase the credit, basically by means of sending another PSMS.

Every minute the information of the pop-up window is updated. After sending a PSMS to increase the credit, the database is updated immediately, but the user has to wait until the seconds counter of the pop-up window reaches the value zero and the renewal actually happens to see the updated credit in the pop-up. This is because HTTP is a client initiated protocol, and therefore the server has to wait until the client initiates the transaction to provide the updated information.

To sum up, every renewal implies reading the value in the field *MinutesLeft*, subtracting one to that value and re-sending the updated value to the user's pop-up window.

The user can actively close the session by clicking on the *logout* button of the pop-up window. At this moment a bye-bye message remembering the password is presented to the user, that will retain the minutes left for the next session.

Conversely, the user can close the browser window, leave the coverage area or turn off the computer without actually logging out. In this case, the gateway detects that the renewal has not occurred and closes the session for that user. As before, the user retains the minutes left for the next session.

When renewal occurs and the field *MinutesLeft* reaches the zero value, the pop-up window does not contain the counter any more, but a message informing that the connection is being closed. At this time the user is removed from the *member* table in the database and included in the *history* table. A minute later, the firewall is actually closed for that user.

5.3 Processing Requests

The PSMS are received by the Access Server in the form of HTTP GET messages at the host and port specified at the SMSGateway. The SMSGateway can be configured to provide some information in addition to the actual the content of the PSMS, such as date, time, MSISDN and MNO identifier. In our case, the user's mobile number was required.

Listening at port 8080 in the Access Server's Internet interface there is an Apache Tomcat 5.5 running to attend the HTTP requests. The Web Application takes the form of a servlet that checks if the request contains the keyword *wifi*. If so, it takes the sender's MSISDN and compares it to the field *login* of the existing entries in the table *member* described above.

If the user is not yet in the database, the servlet generates the four-digit password, creates a new entry in the *member* table with that user information, and answers the user with a welcome message and the password.

If the user already exists, it adds 20 to the *MinutesLeft* field that contains the credit. The answer to the requests informs the user that the credit has been

successfully updated and sends the password as a reminder. This answer is sent to the SMSGateway that converts it into the answer of the original PSMS sent by the user.

The users will probably remember the passwords or otherwise save the answer SMS containing the password for later use. If not, they will have to send another PSMS to receive the password again, and increase their credit in 20 minutes as a side-effect.

All messages received at the Access Server are stored in the database for accounting purposes.

6 The User Experience

This section describes, step by step, the user perception of the service. He or she arrives at the WISP coverage area and turns a computer that receives DHCP configuration. The Access Server is set as the default gateway. The user types any URL and the DNS request is resolved transparently. The HTTP request is intercepted and redirected by the *captive portal*. What the user can actually see in the browser is a login form. It consists on a logo identifying the service, a login and password fields and the terms of use.

The user is informed that is required to send a PSMS to number *7212* with the keywords *upf* and *wifi*, and how much is going to be charged for that message. The user sends the message and immediately receives the welcome message and the four-digit password.

Following the instructions, the user introduces the mobile phone number as a login and the password. After clicking on the login button, a message informing about the success of the authentication is presented for 5 seconds. After that, the user is redirected to the originally requested URL. The renew window pops-up, showing the credit in the form of a count-down counter (Fig. 4).

Now the user can surf the Net as long as the pop-up window is kept open. When the user does not need Internet access anymore, the logout button can be clicked to close the connection and save the remaining credit for another occasion (Fig. 5). If the Internet is needed again, the login and password have to be reintroduced in the login form.

The connection is closed also in the case that the pop-up window can not be refreshed. Possible causes are lack of wireless network coverage or the user closing the window or turning the computer off. As in the previous case, the session is closed and new session has to be opened if the user wants to surf again.

At any moment, normally when the credit is about to expire (the countdown timer approaching to zero), the user can send another message exactly equal to the first one to increase the credit in 20 minutes. In this case the answer says that the credit has been successfully increased and contains the password as a reminder. After some seconds (between 1 and 60) the pop-up window updates to show the increased credit.

6.1 Deployment of the Service in a Multiple WISP Access Network

The service has been deployed in an Wireless Open Access Network (OAN) [8] [9] [10], that is an access network shared by different providers. It was not feasible to create a large wireless access network only to test the new service, but it was easy to integrate the new service into existing infrastructure of the OAN. The coverage includes the University campus distributed in five different locations across the city, and is going to grow and merge with a municipal wireless access network and other universities.

A user connecting to the OAN and trying to browse is offered a list with all the different providers, nine in our OAN deployment. Some of this are well-established services with more than one thousand registered users. Other providers are research-oriented or under development and have only a few registered users. This is the case of the SMSMicropayment service. Since the OAN is supported by the University, it can be used only for academic and research purposes. Therefore, only a small group of selected volunteers where invited to test the service.

Fig. 4. The pop-up window to keep the connection open. A count-down timer indicates the credit.

Fig. 5. Bye-bye window that appears after clicking the logout button

6.2 The Test

The test was run from December 1st to December 20th. Seven of the invited testers actually participated sending a total of 30 PSMS with the key words *UPF WIFI* to the number *7212*. Each of this messages had a cost of 0.30 Euro (plus VAT), much lower than 0.90 planned in the business case. The reason of using a lower fare is that it was only a test and only the MNO expenses had to

be covered. Seven volunteers opened a total of 54 Internet-access sessions during their tests. The results obtained by laptop users where positive, however PDA users were not able to benefit from the service because the PDA browser did not open the *renewal* pop-up window. The result was that the Internet session closed every minute and the user had to re-introduce the telephone number and password every time.

The test was performed at the university premises using the university OAN. Therefore the test had to be necessarily bounded and limited, to avoid raising suspicion on someone obtaining economic profit exploiting the University facilities.

7 Future Work

7.1 Combining Payment Methods

It is simple to extend the presented PSMS-Micropayment approach to include other payment methods. Being the wireless access network a metropolitan network with coverage inside cafes and in terraces, the cafe owner could buy a large number of usernames and passwords to the WISP at a reasonable price.

Then this usernames and passwords would be delivered to the cafe customers together with their drinks. The customers might choose to enjoy their Wireless Internet Access immediately or save it to use it later in any other coverage area. The customer that finishes the 20 minutes received with the drink and wants to keep browsing can either order another drink or send a PSMS to increase the credit.

Another possibility would consist in selling prepayment scratch-cards at kiosks with a considerably larger credit and lower price per minute.

Once the user is on-line, after making the first micropayment, subsequent browsing-time extensions purchases could be done using a myriad of Internet-based payment methods (e.g. paypal) that are much more flexible and do not have the economical overhead associated to PSMS.

The recent apparition of new PSMS worth 1,20 Euro offers a new alternative for users that plan to use the net longer than 20 minutes. Probably the WISP would be able to provide 40 to 60 minutes of Internet browsing to the user and still obtain a profit.

7.2 A PDA-Friendly Captive Portal

Most of nomadic users that are potential clients for public wireless Internet access use *Personal Digital Assistant* (PDA) with small screen and limited web browser. The NoCat captive portal requires that the browser opens a pop-up *renewal* window, something that is not in the capabilities of many PDAs. Therefore, one of the most urgent lines of work is the substitution or modification of NoCat to make the system PDA-friendly.

8 Conclusions

This paper describes PSMS-based micropayments and why they are convenient to charge the users for Wireless Internet Access. A model that involves the Mobile Network Operator and the SMSBroker, in addition to the user and the WISP, is offered as a solution. The cash flow between the actors and the message interchange that allow the user to connect to the Internet are analyzed and exemplified.

An open source captive portal have been modified to obtain a Wireless Access Server with the desired functionality. Then, the overall proposal has been implemented and a WISP have been deployed in a real Wireless Access Network to test the solution, and has been positively validated by users.

References

1. Kniberg, H.: What makes a micropayment solution succeed. M. eng. thesis, KTH/Applied IT, Sweden (2002)
2. Michel, T.: Common markup for micropayment per-fee-links. W3C Working Draft (1999)
3. Droms, R.: Dynamic Host Configuration Protocol. RFC 2131 (Draft Standard) (1997) Updated by RFC 3396.
4. Potter, B., Fleck, B. In: 802.11 Security. O'Reilly (2003)
5. Fils, I.S.: Wifidog. Downloadable open source captive portal (2005)
6. Flickenger, R.: Nocatauth: Authentication for wireless networks. O'Reilly Network (2001)
7. Erle, S.: Nocatgw and nocatauth. Downloadable open source captive portal (2003)
8. Battiti, R., Cigno, R.L., Sabel, M., Orava, F., Pehrson, B.: Wireless lans: From warchalking to open access networks. MONET **10**(3) (2005) 275–287
9. Infante, J., Oliver, M., Macian, C.: Wi-fi neutral operator: Promoting cooperation for network and service growth. In: ITS Conference on Regional Economic Development, Pontevedra, Spain (2005)
10. Barcelo, J., Macian, C., Infante, J., Oliver, M., Sfairopoulou, A.: Barcelona's open access network testbed. In: IEEE Tridentcom, Barcelona, Spain (2006)

Secure Billing for Ubiquitous Service Delivery

Les Green[1] and Linas Maknavicius[2]

[1] University of Technology, Sydney, Faculty of IT
P.O. Box 123, Broadway, NSW, 2007, Australia
lesgreen@it.uts.edu.au
[2] Alcatel Research & Innovation
Route de Nozay, 91460 Marcoussis, France
linas.Maknavicius@alcatel.fr

Abstract. This work presents a secure interaction framework for establishing ad-hoc billing paths between untrusted players in a global telecommunications network. User authentication is supported through the introduction of a Billing provider, responsible for identifying users and acting as a proxy financial entity to visited service providers. Authentication, Integrity, Validity and non-repudiation issues are addressed, along with the processes involved.

Keywords: Security, Billing, Ubiquity, Authentication.

1 Introduction

The current trend in service delivery is a movement towards wireless networks. This is largely due to two factors.

First, consumers desire the convenience of mobile connectivity. Freedom from the confines of wires is sought and consumers are prepared to pay a reasonable price for that luxury. That price may be a lower quality, higher risk of disconnection or simply higher monetary cost [1]. If these factors can be maintained, a user's Quality of Experience may be vastly improved.

Second, providers seek to minimise the large capital and operational expense in construction and maintenance of a wired local loop. This is especially evident in countries with insufficient existing fixed infrastructure. For example, Latin America is the largest current broadband wireless market in relation to population with significant pre-WiMAX deployment [2].

Recently, there has been increased effort towards ubiquity in wireless services, converging heterogeneous networks in respect to technologies and administration[3][4]. Within ubiquitous service availability, an end user is free to roam within the constraints of any possible network connectivity. Consequently, application services utilised by end users may be delivered via numerous network service domains. The owner of each domain involved in a service delivery at any point in time will want to be reimbursed for the services it has provided. A method of billing for the provision of these ubiquitous services is sought.

Within current generation networks, research has suggested [5] that the cost of providing and maintaining a mobile network billing system may be anything up to 50% of the total infrastructure investment and annual turnover. Seamless, automated

B. Stiller, P. Reichl, and B. Tuffin (Eds.): ICQT 2006, LNCS 4033, pp. 90 – 101, 2006.

billing which can provide added value to the service delivery chain is therefore a large concern for current and new generation network operators.

1.1 The User Perspective

Users desire simplicity and predictability [6]. In terms of billing, an end user will not want to pay individually each domain which has provided (a fraction of) a network service. This opens the way for billing aggregators and speculators. Billing aggregators package provided services into one bill whilst speculators speculate on future costs and so quote a fee for some future period. Say the next six months. Speculation offers a method to manage risk and gain in an economic system.

A player may be both an aggregator and a speculator, with end users appointing such a player, following referred to as a Billing Provider, to manage billing on their behalf.

With the global increase of wireless connectivity via a multitude of available access technologies, both in the licensed and unlicensed spectrum, it is not feasible for a billing provider to have pre-existing agreements with every possible network access provider around the world into which an end user could roam. Ubiquitous service availability may require user connectivity from a domain with which a user and associated billing provider is unfamiliar. We are left with issue of establishing a secure billing path for services in an ad-hoc capacity amongst untrusted players.

2 A Secure Billing System

A method is proposed whereby an end user, or nominated user agent is responsible for negotiating service contracts autonomously, and is supported by a means to verify its identity as a basis for trust in an unknown network. Services are automatically billed to users via ad-hoc billing paths.

If automated billing is to be adopted, especially involving unknown players, participants should feel secure in their use of the system. This holistic view of security can be decomposed into two components.

- Trust in the system.
- Trust in the other players.

2.1 System Security

Trust in the system can be established by ensuring known security concerns have been adequately addressed. There are three widely accepted components to information security. Confidentiality, Integrity and Availability.

Confidentiality has been defined by the International Organization for Standardization (ISO) as "ensuring that information is accessible only to those authorized to have access"[7]. Essentially, confidentiality is concerned with keeping information *private*, accessible only by correctly *authenticated identities* with appropriate *authorisation*.

ISO17799[7], an internationally recognised generic information security standard describes integrity as “safeguarding the accuracy and completeness of information and processing methods”. Information integrity describes the consistency of data. It

is concerned with ensuring the data has not been altered or corrupted either intentionally or accidentally. A step beyond information integrity is validity. Data which is valid is correct for its intended purpose. This may involve semantic checking or a logical proof of reason.

The accuracy and completeness of processing methods aims at ensuring a system is used correctly. A system with ill defined processes may be vulnerable to compromise through external hacking or improper use from authorised entities. A recent area of concern is the hacking technique of Social Engineering or "scam" attacks. Such attacks are difficult to control as they often occur outside the bounds of the system designed. A billing system should address information and process integrity and may go further to ensure information is valid in its intended context.

Finally, a definition of availability is given as "ensuring that authorized users have access to information and associated assets when required."[7] A system should be built to adequately handle any foreseeable load and be designed as to prevent attacks which may cause denial of service to authorised users. Positively securing availability is difficult in today's massively connected world. Methods such as access frequency limits and distributed load balancing are a step towards securing availability.

A fourth component of security often included is the aspect of accountability. Accountability involves non-repudiable actions and traceability. For an action to be non-repudiable, the performer cannot deny having performed that action. The aspect of traceability is an issue of specific implementations and is not considered a concern of the communication system.

2.2 Trust Between Players

Trust is concerned with the maintenance of expectation. Establishing trust in another player involves verifying their identity and ascertaining their probability of satisfying or exceeding expectation. Correct identification ensures a player with which you believe you are interacting is in fact that player. Given the correct identification of players, an attempt to model their trust can be made.

The probability of a player performing to expectation may be determined largely from their reputation, built from the outcome of one or more past interactions between players. REGRET[8], a reputational model for trust, approaches reputation from three dimensions. Individual, Social and Ontological.

The Individual dimension models the direct interaction between two players. When the same two players have repeat interactions, private trust models may be built about opponents.

In situations where once off interactions between players are common, Social reputation becomes important as players may have no previous private history on an opponent. In this situation, membership to a group may give lead to opponent reputability, or a collaborative reputation model may be consulted, built from the outcomes of multiple individual interactions.

As an alternative to a single reputability score, the Ontological dimension of reputation divides reputability into separate aspects, each applicable to a domain of concern. Multiple aspects of reputation may be combined by a player to form a view of another.

The secure billing model discussed does not attempt to model or handle reputation, but does deal with the identification aspect of trust. Players within the system are securely identified, but no assumptions are made on players regarding their discovery of the reputation of others.

2.3 System Aims

The main aim of the system is to enable ubiquitous, secure, ad-hoc billing for mobile networks. The goal of this paper is to outline the interaction processes involved in the system, and its secure approach. The way in which players implement the functionality is not important here, providing they adhere to the interaction specification.

In order to provide ubiquity and ad-hoc capability, the billing processes are designed to enable autonomous authorisation for resources controlled by each player. I.e. each player has ultimate responsibility for its own resources. For a user, this is its payment funds, for a service provider, it is the services provided, and for a billing provider, it is its reputation and the provided proxy financial responsibility on behalf of a user.

Following from the security concepts discussed above, the security goals for the system are as follows:

- Ensure positive identification of players involved.
- Enable the transmission of sensitive data, ensuring the secrecy of such information is respected when required.
- Ensure the integrity of information during transmission.
- Enable provable checking of message validity when required.
- Ensure messages are non-repudiable.
- Provide complete and well defined processes, considering possible attacks on the system.

Sections V, VI and VII discuss the security features of the billing system, addressing the above aspects of security.

Before specifying the interaction framework, we begin with a discussion of the players and their roles in the system.

3 Players in the System

There are three entities involved in the billing system discussed. Each entity is represented in the system by an electronic agent and is responsible for its own resources

3.1 End User

End users are the end points of the “billed” telecommunications zone. They are represented by User Agents, residing on a network access device. The end user (or user agent) may have access to one or more physical or administrative network domains simultaneously, and may use billable services provided by these networks once an appropriate billing path has been established.

3.2 Network Service Provider

Network Service Providers control access to one or more services delivered over a telecommunications network. Resources supporting the service may or may not be owned by the service provider. They may be a service retailer, reselling services over a network managed by a different entity, or may themselves manage a network and provide services at varying levels of the OSI stack, from raw IP access, to a managed video conference service for example.

3.3 Billing Provider

Every customer has an associated Billing Provider whose status as a trusted and reliable player is generally known. The billing provider acts as a financial proxy for its customers to visited service providers.

In a global roaming environment, involving varying laws and regulations applicable to users and providers, a Billing Provider is a common ground through which a standard billing relationship can be established. Given a larger buying power than a single end user, the billing provider may negotiate for bulk rates from service providers. A billing provider may be a user's home network service provider or a third party billing provider.

The Billing Provider is responsible for authenticating the user agent to foreign networks as required. It provides the billing service through which network service users and providers can quickly, dynamically and securely establish a billing path. The following activities are the responsibility of the Billing Provider in the delivery of the billing service:

- Bill Aggregation and Speculation – Combine charges accumulated by users for services delivered by visited providers and package them into one bill for a customer.
- User Authentication – Provide a trusted, secure identification mechanism to support roaming users.
- Provider Remuneration – Establish financial paths for remuneration of providers for services delivered to the billing provider's customers. This includes negotiation of customer pricing as required.

4 Processes Involved

Borrowing from Internet Single Sign On (SSO) technologies[9][10], the following security model is proposed, based on asymmetric encryption[11][12] and the Public Key Infrastructure[13].

It is well known that security is only as strong as the weakest link. The Public Key Infrastructure has risks associated [14], however a widely accepted aim of security is to make the effort involved in breaking the security larger than the reward. All risk can not be eliminated and PKI is the current leading approach to many aspects of security.

4.1 User Authorisation and Authentication

User Authorisation is the process that occurs between a user agent and the service provider to allow the user to request services from the provider. A natural requirement

a provider may have on authorising a user is that the user agent is first authenticated. I.e. Its identity is verified by a trusted party. Refer to Figure 1 for a numbered diagram corresponding to the below points.

1. The user generates a public and private key pair and registers the public key with the Billing Provider. This step happens once and may be performed manually by the user via a web interface on the billing provider's website or by another offline means. The Billing Provider must be sure that the public key it has registered for a user belongs to the appropriate customer.

   The system relies on the user's private key remaining known only to the user. If this secrecy is breached, the user must generate a new key pair and register the new public key with the billing provider, revoking the previous public key. User public keys remain valid throughout a specific time period. All entities in the negotiation environment should therefore have synchronised clocks within a broad tolerance to ensure correct authentication. This requirement is easily satisfiable using the existing and widely used network time protocol [15].
2. When a user agent wants authorisation to request services from a provider, it creates a User Authorisation Request (Figure 2) comprised of its billing provider details and customer identifier and sends it to the service provider.
3. At this point, the Authorisation request may be rejected by the service provider for any reason, such as a banned billing provider or maybe because the service provider is simply too busy.
4. The service provider then builds a User Authentication Request (Figure 3) from the original signed User Authorisation Request, including a request for the Billing Provider's Identity Credentials (Digital Certificate) if the identity of the billing provider is not known in advance. The Authentication Request is then sent to the billing provider.
5. The billing provider checks the User Authentication Request by validating the signature on the enclosed User Authorisation Message against the stored public key on the customer record.

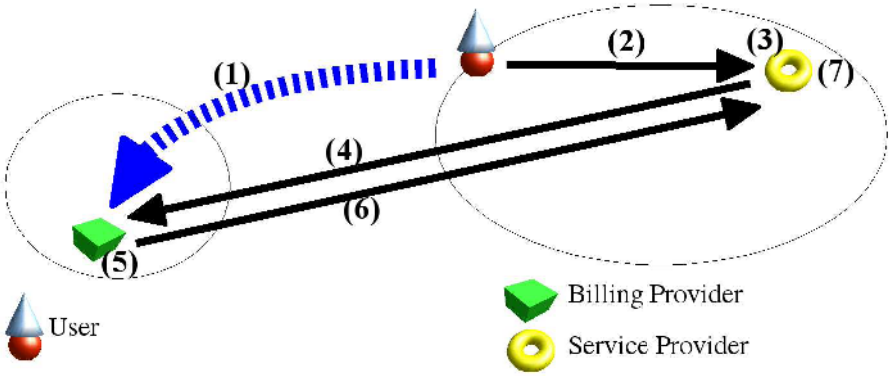


Fig. 1. User Authentication

6. If user authentication is successful, the Billing Provider returns a Success Message (Figure 4) bundled with the user's public key and a Service Provider Billing Authorisation which authorises the Service Provider to issue bills for

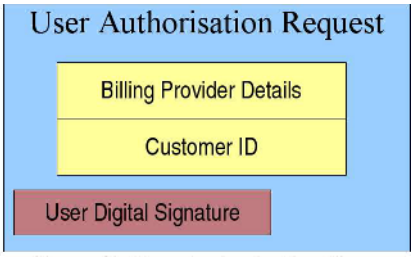


Fig. 2. User Authorisation Request

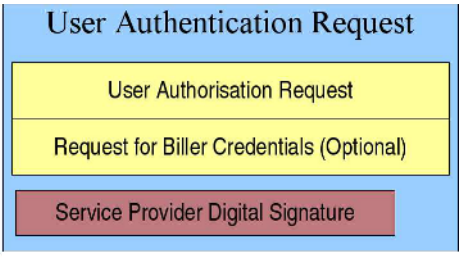


Fig. 3. User Authentication Request

the supplied public key. This billing authorisation may include constraints on the provision of services or accumulation of costs by the user, or it may be an open authorisation to bill as required. The Billing Provider's credentials are also supplied if initially requested, to certify the billing provider's identity.

The billing provider must, for each customer, maintain a list of Service Providers which have been authorised as billers for the particular customer's public key. This is to ensure that if for some reason the customer's purchasing rights are revoked, or the secrecy of the private key associated with the public key has been breached, the Billing Provider can contact each Service Provider to revoke the public key.

The Billing Provider must receive proof from each Service Provider it has registered as an authorised biller that it has acknowledged the revocation of the public key. One appropriate means of non-repudiation is through the use of digitally signed messages in concert with valid Digital Certificates. A Billing Provider cannot claim it did not have a particular service provider registered as a recipient of a key because the User Authorisation Success Message provides a digitally signed proof that a service provider was given authorisation to send the billing provider bills signed by a particular public key.

7. Upon receiving a success message, the service provider validates the billing provider's credentials. The returned user's public key is then used to validate the signature from the initial User Authorisation Request received from the User Agent. On successfully passing authorisation, a trusted path for future services is established, and the user can proceed to request services from the provider.

   Users' public keys are stored by service providers for use in validating future Service Level Agreement (SLA) requests. In this way, the service provider need not validate with the billing provider for each SLA provisioned.

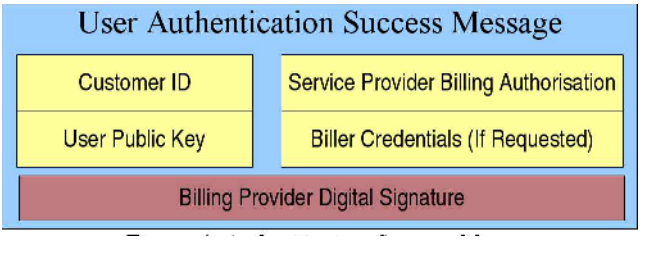


Fig. 4. Authentication Success Message

4.2 Billing Path Establishment

Creation of secure, ad hoc billing paths from End Users to Service Providers involves establishing two billing relationships. A billing relationship must be forged between an end user and their billing provider. Such relationships are common in today's telecommunication environment between Internet subscribers and their ISPs. This customer facing relationship is tied in heavily with marketing strategies and individual business models chosen by the billing provider and is not explored further in this work. It is assumed a method of billing exists between end users and billing providers. However, the way in which a billing provider is charged from service providers for services used by an end user is the interesting research point and is addressed here.

The Service Provider to Billing Provider relationship process allows Network Service Providers, visited by a roaming user to form ad-hoc billing relationships with unknown Billing Providers associated with those roaming users. To establish a solid financial relationship, billing providers and service providers should be certain of each others identity and must agree on other billing details such as invoicing frequency and payment method employed. Discounting specifics and other pricing related details may also be included in a billing method.

Payment methods adopted may be be a bank routing and account number, credit card number or any other of the numerous payment systems available. Payments are not addressed further in this work. It is sufficient to say that at Biller/Service Provider negotiation time, the payment method to be used should be defined. ISO 20022[1], IFX[2] and Rosettanet[3] all offer tools to mark up payment information.

To remain a generic, open and extensible system, specifics of the final billing method between service provider and billing provider should not be limited to a single implementation. Ultimately the billing method should be based on individual provider requirements. A common process for establishing or "bootstrapping" the billing process must be defined so a suitable billing method for use can be agreed on ad-hoc.

The Bootstrap Mechanism. The mechanism used to establish a billing method between the service provider and billing provider is based on the fulfillment of individual requirements. Both players in a potential billing relationship may have different requirements of the final billing method specification. The following process ensures all requirements are fulfilled so an appropriate billing method can be found.

Based on ontological knowledge representation, the mechanism occurs in three stages:

1. Initially, each party informs the other of a list of OWL[16] ontologies which it can understand, and in which the billing method may be specified.
2. In the second stage, the service provider informs the billing provider of the information it requires to bill for services. These requirements are expressed using ontological concepts exchanged in the previous step.

   Using these requirements, the billing provider then constructs a billing method template including components satisfying the informational requirements of the service provider, and components satisfying its own requirements of a

[1] http://www.iso20022.org/
[2] http://www.ifxforum.org/
[3] http://www.rosettanet.org/payment

billing method. The billing method template may only be composed of ontological elements common to both sets of ontologies specified in step 1.

3. The third stage in establishing a billing method is forming an agreement on the concrete values to be used in the billing method. Such values may include pricing specifics or bill frequency, etc. This stage follows an offer / counter offer / final offer argumentation strategy at which point either the negotiation succeeds and the outcome is a concrete billing method instance, or the negotiation fails and the service provider may not provide services to the billing provider's customers. A Service Negotiation Protocol (SrNP)[17] has been proposed by the TEQUILA[18] and MESCAL[19] projects and is well suited to this argumentation component.

A billing provider may also be a network service provider – and hence at some point may act as a Service Provider to the foreign Service Provider's home customers. Both parties have something to gain by establishing an optimal billing agreement.

At initial Biller/Service Provider relationship establishment, for instance when a customer of a Billing Provider wanders into a unknown Service Provider's zone and wishes to use its services, the Service Provider and Billing Provider have a requirement to establish ***some sort*** of agreement ***before*** the user can use the services. This may be relatively urgent. The Service Provider may have a "base" pricing scheme which is used when it has accumulated little or no information on the Billing Provider or User and therefore has no indicator of trustworthiness of the player. The Billing Provider is left in a take-it-or-leave-it situation with the service provider until a stronger relationship can be formed.

In contrast, *adjustments* to the billing method formed between a service provider and billing provider are infrequent, may have no strict time requirements for convergence, and presumably happen over high speed network links. A more complex and optimal negotiation strategy can therefore be employed.

5 Secure Communication

Security has been discussed in terms of Authenticated Identification, Reputation, Authorisation, and secure, non-repudiable processes. Billing for ubiquitous services involves communication between concerned players. To address all aspects of security, this communication between players should also be secure. Issues such as message integrity, privacy and accountability should be addressed.

5.1 Communication Mechanism

Secure billing is part of a greater project exploring service ubiquity through electronic negotiation, titled "Managing Quality of experience Delivery In New generation telecommunication networks with E-negotiation" (QDINE). A primary goal of the project is the development of a secure, comprehensive, open service negotiation framework built on intelligent agents. The agents use economic principals to enable ubiquitous, mobile service provisioning. Services are described within Service Level Agreements (SLA).

There are three agents involved in the billing system. The User Agent, Network Service Provider Agent and Billing Provider Agent. Agents communicate using FIPA Agent Communication Language (FIPA-ACL)[20]. ACL messages may be sent via numerous methods. These message transports may be standardised or implementation specific. Some existing message transports are RPC and IIOP.

Message exchanges are grouped into interaction protocols, with agents adopting only the protocols necessary for their personal tasks. Content within the ACL messages is expressed in the Web Ontology Language (OWL)[16].

OWL ontologies have been created for use in specifying Service Level Agreements, describing interaction protocols between agents and formalising the content of ACL messages.

5.2 Message Privacy

Messages passed between agents in the system may travel through untrusted networks. Payment details or other sensitive information may require privacy within participating agents. Public Key Encryption is used to secure sensitive messages from unintended recipients.

Encryption may occur at the socket layer via SSL or TLS. Additionally, the content of ACL messages may be encrypted. Encryption at the network layer through IPSec, although possible, is not likely as the interactions between agents are brief and asynchronous.

A security add-on for ACL called X-security has been developed [21] allowing encryption of ACL message content. Additionally, as the content language used in the QDINE project is based on XML, the standard XML encryption [22] methods may be used.

5.3 Accountability

Service Level Agreements are an electronic contract for a service. To ensure these contracts and the associated bills are not disputable, the automated interactions should be traceable and positively identify the sender.

As part of the QDINE framework, Billing Providers and Service Providers should have a commonly accepted form of identification. Valid digital certificates from trusted certification authorities are widely used for this purpose. Additionally, as outlined above in *User Authentication and Authorisation* section, the billing provider is responsible for guaranteeing a user's identity.

All agents in the framework have a private and public encryption key. Messages sent between agents in the framework must be signed with the sender's private key. In this way, every message sent can be provably attributed to a particular private key and hence, one entity.

5.4 Integrity

Ensuring the integrity of a message involves proving the message has not been tampered with after being sent. A common method of achieving this is by building a

secure digest of a message and signing it with the sender's private key. In doing so, the above requirement of Accountability is also addressed.

A secure digest is unique to the message used to build the digest. If the message is altered in any way, a different digest will be built on application of the digest algorithm. To generate a secure digest, WHIRLPOOL[23][24], or one of the SHA-2[25][24] family of algorithms may be used. To date, these algorithms remain uncompromised and are recommended for standardised secure signing of data.

The X-Security add-on for ACL may be used to attach signature and digest information to an ACL message, alternatively, or additionally, content of the messages may employ the XML digital signature framework [26].

6 Conclusion and Future Work

This work has explored the need for a secure, comprehensive and open billing solution for ubiquitous service delivery over mobile networks. Aspects of security are explored and the aims of the billing framework are presented in respect to these security aspects. The billing framework is introduced with a discussion of the players involved, along with their roles.

The framework is described in terms of the the interaction processes involved, presenting a method to establish secure, ad-hoc billing paths from provider to end user. Within the system, players maintain autonomous responsibility for their own resources.

Identification, transmission privacy and data integrity, along with non-repudiation issues are addressed through the use of different components of the public key infrastructure. The use of OWL messages enables automated validity checking. Additionally, the system is highly available due its distributed nature. I.e. All players are responsible for their own resources and the authentication management is distributed amongst unlimited billing providers.

The introduction of a Billing Provider and associated processes to promote security, handle ad-hoc relationship management and encapsulate speculation within the financial system is an innovative component with respect to current state of the art.

An interesting component of the work is the inclusion of an ontological approach to secure communication and the properties of such an approach.

To date, a framework has been designed for SLA negotiations[27]. Future work will see the implementation of this secure, ubiquitous billing work as an integration into the SLA negotiation framework. The billing component will be analysed for security weaknesses in a formal process.

Acknowledgments

This research is performed as part of an Australian Research Council Linkage Grant, LP0560935 between Alcatel and the University of Technology, Sydney. It extends work already performed on the Negotiation of Service Level Agreements for New Generation Networks, performed as part of the Alcatel Research Partnership Program (ARPP).

References

1. Jafarkhani, H., 2005, *Space-Time Coding: Theory and Practice*, Cambridge Uni. Press
2. Gabriel, C., 2005, *A Global View of Pre-WiMAX Deployment*, from: Wimax Trends: The Net's Leading Resource for WiMAX Technology & Solutions, *source:* http://www.wimaxtrends.com/articles/excerpt/e101005a.htm, *accessed:* 03-03-2006
3. The FON project: WiFi Everywhere, *Source:* http://en.fon.com/, *Accessed: 03-03-2006*
4. Computer Business Review Online, 2006, *T-Mobile first to bridge 3G, EDGE, GPRS, and WiFi*, March 13, *source: http://www.cbronline.com/article_news.asp?guid=C0AB6620-5C75-49A9-95B9-0F30F14C3884, accessed: 04-04-2006*
5. Cushnie, J., Hutchinson, D., Oliver, H., 2000, *Evolution of Charging and Billing Models for GSM and Future Mobile Internet Services*, QofIS '00: Proceedings of the First COST 263 International Workshop on Quality of Future Internet Services, London, UK
6. Nielsen, J., 2000, *Designing Web Usability: The Practice of Simplicity*, New Riders Publishing, Indianapolis
7. *ISO/IEC 17799:2005, Code of practice for information security management*
8. Sabater, J.,Sierra, C., 2001, *REGRET : a reputation model for gregarious societies*, Fourth Workshop on Deception, Fraud and Trust in Agent Societies, New York
9. Josephson, W. K., Sirer, E. G., Schneider, F. B., 2004, *Peer-to-Peer Authentication With a Distributed Single Sign-On Service*, Third Intl. Workshop IPTPS, San Diego, CA
10. The Java Open Single Sign-On Project, *Source:* http://www.josso.org, *Accsed:* 03-03-06
11. Hellman, M. E., 2002, *An overview of public key cryptography*, IEEE Communications Magazine, *vol.* 40, *num.* 5, *pp.* 42 – 49
12. Kaliski, B., 1993, *A Survey of Encryption Standards*, IEEE Micro, *vol.* 13, *num.* 6, *pp.* 74-81
13. The Public-Key Infrastructure Charter, *Source:* http://www.ietf.org/html.charters/pkix-charter.html, *Accessed:* 03-03-2006
14. Ellison, C., Schneier, B., 2000, *Ten Risks of PKI: What You're not Being Told about Public Key Infrastructure*, Computer Security Journal, *vol.* 16, *num.* 1, *pp.* 1-7
15. Mills, D. L., 1992, RFC 1305 - Network Time Protocol (Version 3)
16. OWL Web Ontology Language, *Source:* http://www.w3.org/TR/2004/REC-owl-guide-20040210/, *Accessed:* 26 Nov 2004
17. *MESCAL Deliverable 1.2, Source: http://www.mescal.org/deliverables/MESCAL-D12-public-final.pdf*
18. *TEQUILA - Traffic Engineering for Quality of Service in the Internet, at Large Scale, Source:* http://www.ist-tequila.org, *Accessed:* 02-02-2006
19. *MESCAL - Management of End-to-end Quality of Service Across the Internet at Large, Source:* http://www.mescal.org, *Accessed:* 02-02-2006
20. FIPA ACL Message Structure Specification, *Source:* http://www.fipa.org/specs/fipa00061/, *Accessed:* 02-02-2006
21. X-Security - Communication Security in Multi-Agent Systems, *Source:* http://agents.felk.cvut.cz/security, *Accessed:* 02-03-2006
22. XML Encryption, *Source:* http://www.w3.org/TR/xmlenc-core/, *Accessed:* 02-03-2006
23. Barreto, P. S. L. M., Rijmen, V., 2000, *The WHIRLPOOL Hashing Function*, First open NESSIE Workshop, Leuven
24. ISO/IEC 10118-3:2004, *Part 3: Dedicated hash-functions*
25. C S R C - Cryptographic Toolkit: Secure Hashing, *Source:* http://csrc.nist.gov/CryptoToolkit/tkhash.html, *Accessed: 03-03-2006*
26. XML Digital Signatures, *Source:* http://www.w3.org/TR/xmldsig-core/, *Accessed:* 02-03-2006
27. Green, L., 2004, *Auto Negotiation of Service Levels for NGNs - T2D2 - System Architecture*, University of Technology, Sydney

Author Index