Paper

A Software Platform for Research on Auction Mechanisms

Mariusz Kamola[a,b], Ewa Niewiadomska-Szynkiewicz[a,b], Krzysztof Malinowski[a,b], Wojciech Stańczuk[a], and Piotr Pałka[a]

[a] *Institute of Control and Computation Engineering, Warsaw University of Technology, Warsaw, Poland*
[b] *Research and Academic Computer Network (NASK), Warsaw, Poland*

Abstract—The platform for research on auction mechanisms is a distributed simulation framework providing means to carry out research on resource allocation efficiency mechanisms and user strategies. Both kinds of algorithms examined are completely user-defined. Interaction of algorithms is recorded and pre-defined measures for the final resource allocation are calculated. Underlying database design provides for efficient results lookup and comparison across different experiments, thus enabling research groupwork. A recognised, open and flexible information model is employed for experiment descriptions.

Keywords*—*auctions, market simulation, multi-commodity markets.

1. Introduction

The rules for interchange of goods are the legal setting determining market operation. Design of such rules is a fascinating, demanding and important task, often preconditioning efficient operation of economies. When present on a market driven by a set of rules, an entity always implements its own best strategy, developed subject to those rules. However, the entity's initial decision to participate depends on the market attractiveness, comprising its legal framework.

Developing rules for market operation such that desired aims are reached, or trading mechanism design, was one of subjects in collaborative research project "Next-Generation Services and Data Networks — technology, application and market aspects", supported by Polish Ministry of Science and Higher Education. The structure of activities for thematic group "Trading models for transmission services marketplace" is presented in Fig. 1. Note that market clearing, bidding and resource allocation strategies have been considered there as parallel tasks. Examining how they interact when put together is rarely available with analytical models, especially that they are developed by many research teams, and thus with various approach.

This is where the developed platform for research on auction mechanisms (PRAM) comes in, providing those research groups the common language and information model to express the settings of the market, the common entity-market interaction scheme, and the common repository of searchable results. PRAM is, chronologically, the finalisation of the project research activities, making it possible

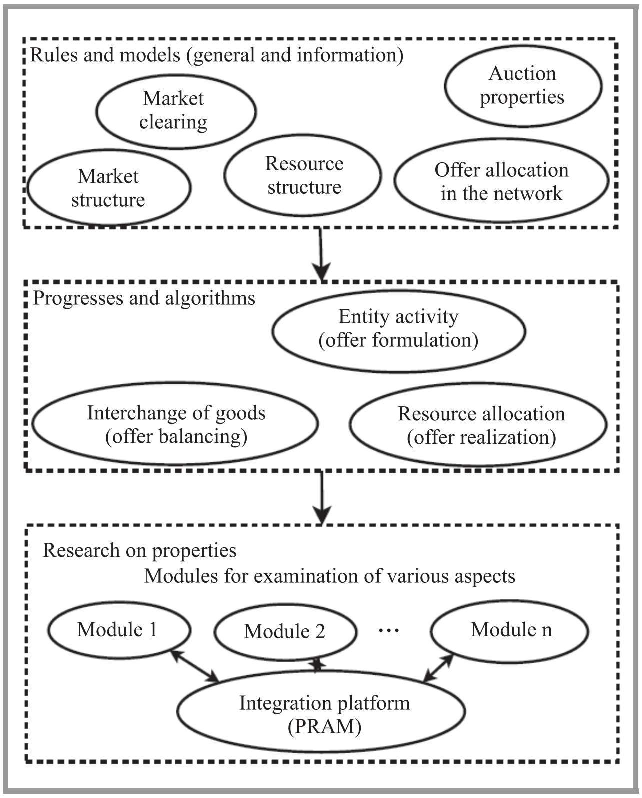


Fig. 1. The position of PRAM creation task within "Trading models for transmission services marketplace" thematic group.

to carry out simulation-driven analysis of strategies developed. It provides for verification of strategies while their assumptions are partially not met or the information about market state is incomplete.

The structure of this document is as follows. Section 2 presents existent and mature trading platforms, while PRAM architecture and functionality is explained in Section 3. This is followed by discussion on comparison criteria for bandwidth trading mechanisms in Section 4. Conclusions are given in Section 5.

2. Existent Frameworks for Interchange of Goods

Information, functional and physical architectures of exemplary trading platforms are presented below. This overview is an improved version of the material presented in [1]. Examples supporting multilateral trade, e.g. where many participants place sell and/or buy offers at the same time, have been selected. The examples come from various branches and support interchange of different kinds of goods. Great majority of presented examples are fully operational in the business, but some concepts still in research phase are presented as well.

2.1. FCC's Integrated Spectrum Auction System

The automated auction system (AAS) was the first Federal Communications Commission's system used to support frequencies auctioning. AAS required from bidders to use a dedicated software and to use dialup connections to FCC's call centre. Because of growing Internet popularity, AAS was decided to be upgraded to an online web application.

The current new integrated spectrum auction system (ISAS) has replaced the former AAS and Form 175 systems (the latter serving filling up frequency request forms). When compared to its predecessors, ISAS offers extended functionalities including request data validation, advanced data query, integration with other FCC forms, ergonomc interface, improved bid placement [2]. The system is now open to every Internet user.

ISAS system provides for simultaneous auctions with many rounds and the possibility to bid for a bunch of licenses.

2.2. WARSET – Warsaw Stock Exchange Trading System

The quoting at Warsaw stock exchange (WSE) is done via WARSET transaction system [3]. WARSET supports fully automated offers processing and transaction making. It is easily accessible and provides complex information about the market. Moreover, WARSET is now integrated with brokerage, thus facilitating bidding for broker's customers. The principal WARSET contractor, acting within a consortium, has provided the necessary hardware and the client application, making it possible for 37 brokerages collocated with WSE to connect instantly, and for the rest to connect via wide area network. Auxiliary transfer agents have been developed, like the one to interact with the Polish National Depository for Securities.

2.3. Polish Power Exchange

Polish power exchange (PPE) has been founded as a central component of the Polish energy market undergoing liberalization. Since its very beginning, PPE was in the fore while deploying novel solutions for energy trading. Within six months of PPE's operation the spot energy market has started, with its prices being the reference point in bilateral contracts. In 2003 PPE has been licensed by Polish financial supervision authority to operate an electricity marketplace.

In 2008 PPE has started commodity derivatives market. Derivatives for energy quoted there make it possible to calculate longer-term electricity prices, which enables big market players to forecast and optimize buy or sell prices. PPE runs on a state-of-the-art trading platform, provided by NASDAQ OMX – the biggest manufacturer of trading platforms in the world [4]. PPE is technically capable to serve whole Polish energy market.

2.4. MERKATO – Bandwidth Marketplace

New York-based bandwidth trading platform MERKATO did not count much on the market, but it was a remarkable enterprise. MERKATO was designed as an open and scalable platform for real-time bandwidth acquisition in Internet. The auction algorithm serving requests, invented and patented by The Invisible Hand Inc., was based on a progressive second-price auction. MERKATO had adopted a distributed approach: a microauction was organized for each network resource, as bandwidth. Therefore, bids for resource bundles had to be submitted and processed independently [5].

Market clearing was done every 5 minutes, by collecting bids, calculating equilibrium prices and allocating throughput to winners. Interestingly, the whole process was fully automated: once the winners got selected, the operators reconfigured their networks and access points accordingly. Moreover, MERKATO offered derivatives market where futures were traded in broad time-scale.

Unfortunately, MERKATO is not operational since 2007, and without an apparent successor or a competitor.

2.5. PeerMart – a Distributed P2P Auction System

PeerMart is a technology for auction-based resource interchange in peer-to-peer (P2P) networks [6]. P2P networks growth is driven by an idea of sharing own and using others' resources worldwide, by means of agents running on home PCs, without centralized management of any sort. However, P2P users often act egoistically, e.g. by not providing any resources and switching their PCs on only when they need to use others' resources.

PeerMart was designed to solve such problems by introducing incentives to share own resources. It utilizes double auctions in distributed setting, thus rewarding valuable content. Furthermore, redundancy mechanisms are applied to ensure system robustness in presence of non-cooperating agents.

Every resource is sold or bought in PeerMart via a double auction carried out by dispersed broker-peers (auctioneers).

2.6. Storage Exchange

Storage Exchange is another double-auction based trading platform [7]. The goods being traded is storage space: the sellers are storage providers or any businesses possessing free disk space, and the buyers are institutions in need of virtual disks.

2.7. Band-X – Architecture for Bandwidth Trade in IP Networks

The band-X system is not the operational bandwidth trading platform; it rather a mature concept of such system, developed at Drexel University [8]. Quality of service provided through DiffServ and IntServ technologies are the system technical foundations. Multilateral agreements are supported, as well as spot and derivative trading. The work focused on organizational aspects of the platform operation.

3. The Platform Architecture and Functionality

The platform for research on auction mechanisms is a tool supporting examination of behavior of models developed in the course of the project. There are two kinds of models: representing auctioning and resource allocation process, and representing market entities activities. The basic scope of PRAM application includes

– verification of theoretical model properties through simulation,

– estimation of model sensitivity,

– assessment of observed model properties in scenarios where selected models interact.

Specifically, PRAM helps in searching for Nash equilibria in games induced by models interaction, in analysis of those equilibria properties, and in estimating sensitivity of results for constraints defining market rules. Consequently, the platform makes it easy to compare market clearing mechanisms, and to infer about their practical efficacy. Specific use scenarios are examining robustness of those mechanisms when players exhibit unusual behavior (e.g., speculate) or estimating the maximum disproportion in players' market strength when a trading mechanism still remains efficient.
The main architectural PRAM assumption is ergonomics for mechanisms testing and ranking. It is also important to extend existing repository of models easily, by implementing new resource allocation mechanisms and new agents simulating user behavior.

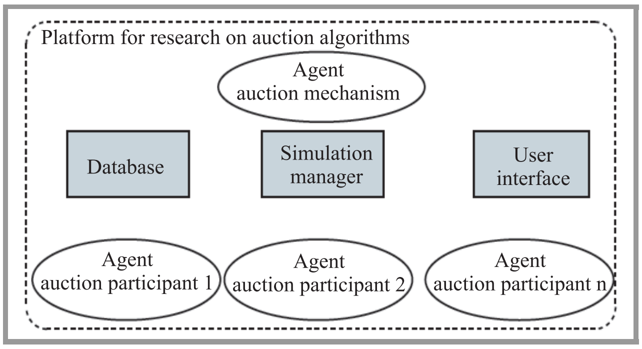


Fig. 2. The main PRAM modules.

PRAM architecture is a modular one, cf. Fig. 2. The information exchanged vertically by the modules are conforming to multicommodity market data model (M^3), developed earlier by project participants [9]. The modules of auction mechanism and auction participants are replaceable, while the middle layer modules constitute PBMA core. Simulation manager is responsible for starting and setting up links to agents, performing the simulation, and the cleanup. Database module provides persistence to experiment configuration data as well as intermediate and final results. User interface module defines forms for experiments creation, configuration, running and processing.

3.1. Multicommodity Market Data Model – M^3

M^3 is a method and format for a formal description of a market where trade of resources takes place. It has been initially developed to describe offer structure in the energy market in Poland. For its generality, it has been next used to model IP network bandwidth trade [10]–[13]. Used in PRAM it effectively describes properties and dependencies between goods being traded.
M^3 defines the following basic entities and relations between them:

– network nodes and arcs, describing the topology of the network where capacity trade takes place,

– market entities (users, providers) that buy or sell resources (capacity),

– resources being offered, with their proper attributes,

– offers, i.e. bindings of market entities and resources, offered or demanded at a specific price.

It is also possible to define compound resources, i.e. containing simple resources and other compound resources. Analogously, one can define simple and compound offers and market entities, exploiting the model generality and flexibility. However, it can also be applied without knowledge of advanced features, like aggregation facilities. It is possible to declare only key values: *offeredPrice*, *min/maxValue* and *shareFactor* (1 for sell, –1 for buy offers), leaving other unset. Fields *acceptedVolume*, along with *sell/buyPrice* parameters of commodity structure contain results of the market clearing process.

3.2. Functional Requirements

Functionality of PRAM is determined by three major assumptions:

- The platform will principally be used in research context, and applied for varying set of models interacting.

- Ease of historical results retrieval and comparison is central.

- Design patterns, data models and communication mechanisms applied must make PRAM a valuable proof of concept of a commercial trading system.

3.3. Platform Users and Resources Being Subject to Interchange

It is assumed that the main PRAM user is a researcher or designer of trading mechanisms, i.e. algorithms for resource allocation and quoting. PRAM simulation framework provides means to examine interaction of mechanisms and market entities, both being represented by software agents. Research via simulation aims to discover phenomena difficult to analyze and predict, like in scenario where all or part of market participants are human.

Within the project, PRAM is applied to bandwidth allocation in data networks, but it can easily be used in other, quite distant, application domains, like parallel problem solving (cf. [14]). The platform is a framework for market simulation; it implements its fixed components (the database, simulation manager and user interface) and exemplary replaceable components (user and auction mechanisms agents). Two auction mechanisms have been implemented: balancing communication bandwidth trade (BCBT – see [12]) and effective bandwidth auction mechanism (EBAM – see [13]). They constitute a good starting point for eventual further development and research.

3.4. The Architecture

PRAM design follows an open system concept, i.e. it facilitates rapid new agent prototyping. System architecture is multigrained, composed of federations of simulation components, like user agents that act synchronously, and the market mechanism, coupled via the simulation manager. Data persistence is provided by the underlying relational database module, and PRAM web interface is managed by the interface module. A single, universal application programming interface and communication protocol between the management module and agents have been designed. It makes possible to treat all agents uniformly, wherever possible, as black boxes. On the other hand, simulation manager operation is, to much extent, transparent for the agents. Functionalities of the five types of PRAM modules are described below.

Auction mechanism agent. It is a functional module responsible for market clearing, resource allocation and quoting. It runs, in principle, by performing optimization tasks, which can be done by external solvers (e.g. CPLEX, LPSolve) or built-in custom optimization routines. This module must implement the following operations:

1. *Auction initiation.* This operation is executed once at each auction start. It is invoked by the simulation manager, thus informing the auction mechanism that the auction has just started. The simulation manager passes information about the system (i.e. network topology, list of market entities, list of resources being traded, auction-specific parameters) to the auction mechanism. Most of the data are stored in M^3 format.
2. *Resource allocation.* This operation may be executed more than once, depending on the type of auction. The simulation manager invokes this operation with buy/sell offers as arguments. Operation results are resource allocations and prices set by the auction mechanism. The operation arguments and results are expressed in M^3 format.
3. *Simulation termination.* This operation is executed once for each auction. It is invoked by the simulation manager to communicate the auction mechanism that it is going to be destroyed because the simulation has just ended. The auction mechanism agent is given a chance to perform cleanup activities, like disconnecting from a remote solver.

User agents. It is a functional module implementing market entity behavior. Many types of user agents can take in a single simulation experiment, their emergent collective behavior being often impossible to be expressed analytically. The main results of user modules operation are trading buy and/or sell offers that, after being merged by the simulation manager, get presented to the auction mechanism. These modules must implement the following operations:

1. *Auction initiation.* Like for the auction mechanism agent, this operation is executed once at each auction start. This operation gives also an opportunity to pass any extra parameters to agents that parametrize their working, including, e.g. parameters of probability distribution, IDs of other agents that are going to form a cartel etc.
2. *Offer preparation.* On operation invocation actual resource allocation and prices are passed to agents. In response, agents return their new offers to the auction mechanism. For iterative mechanisms this operation is executed many times; for one-step mechanisms this operation is called twice (on the first call, there are no allocations yet; on the second call the auction is over and no user response is expected).
3. *Simulation termination*. Like for the auction mechanism, this operation is called only once, on simulation end.

Simulation manager. It is the PRAM central module, responsible for running experiments, i.e. resource allocation sessions. Its functionality covers both pure managerial activities and in-depth analysis of the data being exchanged. The simulation manager:

- manages spawning, cleanup and synchronization between agents;

- forwards data between user agents and the mechanism;
- merges separate offers into one M^3 model, presented to the mechanism;
- calculates predefined simulation outcome statistics.

The fundamental requirement is that multiple heterogeneous agents must be managed timely and reliably results in multithreaded design of the simulation manager. Communication state with each agent is handled in a separate thread, and the communication technology is Java Messaging Service (JMS). Commands spawning the agents are delegated from Java to the underlying operating system.

Database module. The database module manages persistence of M^3 structures and PRAM-specific data structures into a relational database. The data module is the only interface between other PRAM modules and the database, providing efficient mapping of Java objects into data tables, and the database is the only repository of any PRAM data, which does not preclude database direct access from other applications. Any M^3 data, before being stored into the database, need to be converted into plain old Java objects (POJO), using the converters generated automatically from M^3 XML schemas (XSD). PRAM database contains therefore:

- agents configuration (startup parameters, mapping between software agents and market entities),
- scenario definitions (selection of M^3 models, agent types, general scenario attributes and descriptions),
- M^3 intermediate and final scenario results (offers, prices, allocations),
- solution statistics calculated by PRAM.

Graphical user interface. PRAM user web interface makes it possible for a user to define and run test scenarios, and to filter, analyze and visualize individual and aggregated results. Using a diagram, graph or table form, a user can observe data that are:

- simulation-oriented – all output data (final and intermediate allocations, bids and prices) are shown for a selected scenario;
- resource-oriented – selected resource allocations and prices are shown for various testing scenarios;
- user-oriented – selected user bids and allocations are shown for various testing scenarios.

3.5. Data Flows between PRAM Modules

Figure 3 illustrates the data exchanged between PRAM modules. It must be emphasized that the central role of the simulation manager is evident as soon as the user requests to carry out the simulation, while the simulation environment is being initialized. This process, and the simulation itself, is executed in the following steps:

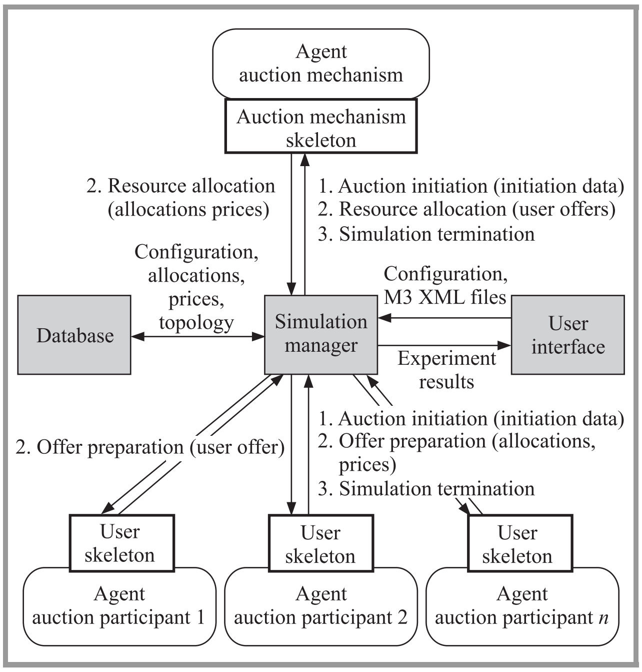


Fig. 3. Data flows between PRAM modules.

1. Configuration for simulation is read by the simulation manager from the database: network topology, market entities and, product and offer definitions, agents and simulation parameters are loaded.
2. Simulation manager spawns the agents (or waits for those spawned externally) and waits until they report they may start simulation.
3. The agents get initialized by the manager.
4. The simulation runs by alternately collecting user agents offers, merging them and forwarding to the auction mechanism agent. Mechanism reply, containing prices and allocations, is passed back to agents, and the procedure is repeated. Intermediate results are stored in the database.
5. The simulation is broken on mechanism request, or when the maximum number of iterations is reached. Finally, all agents are requested to decommit resources and quit.

PRAM has been implemented in Java language, using selected Java Enterprise Edition components as JMS and Java Persistence API (JPA – Hibernate implementation). Communication between platform components during simulation is presented in Fig. 4. All experiments data that can be stored using M^3, are stored in both plain XML text files, and as persistent POJO structures. For communication with agents, only plain XML is used, and any extra non-M^3 parameters are passed using Java native serialization.

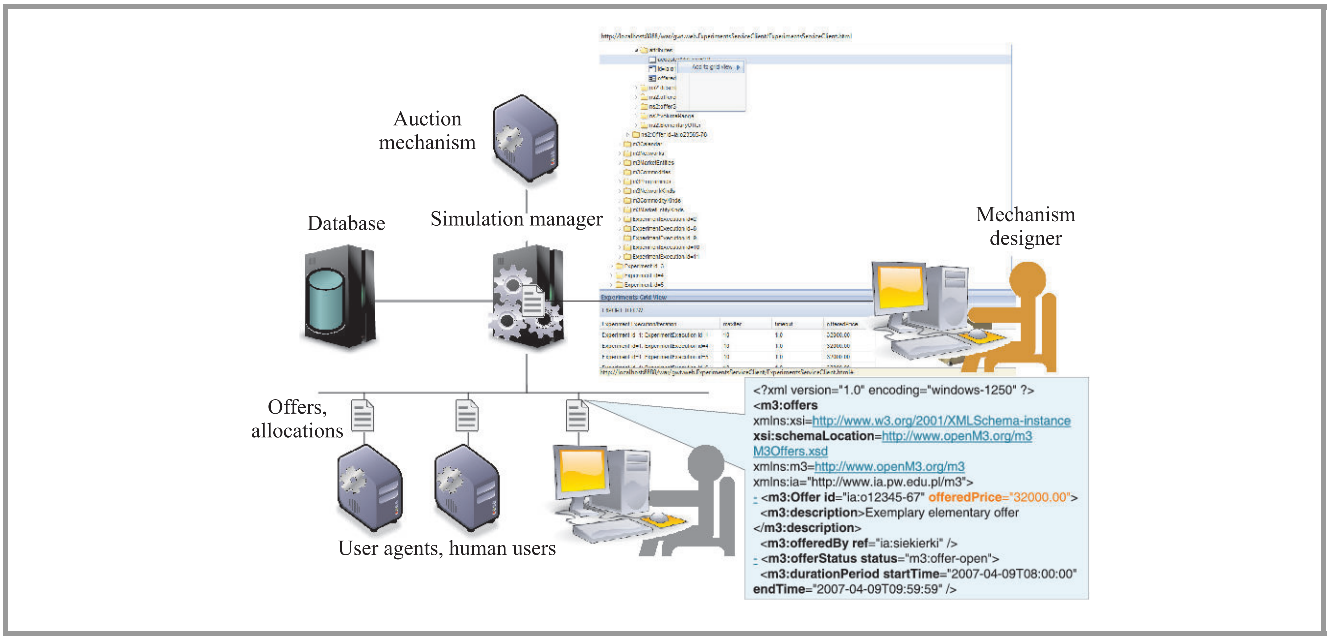


Fig. 4. Communication between modules during simulation.

Graphical user interface acts as a control station for the whole process of simulation and data analysis. GUI component, running on the server side as a web application, is loosely coupled with other PRAM components via Spring framework [15]. While operating PRAM, a user can utilize existent external software supporting network topology and calendar edition. The software is dedicated to use M^3 model, and to operate on M^3 files. It operates locally, and the resulting M^3 files can be uploaded to PRAM afterwards.

4. Comparison Criteria for Bandwidth Trading Mechanisms

The theory of mechanisms defines a number of mechanism properties. Many of those properties are desirable for any market mechanism being under construction. The properties can be perceived as criteria for mechanism ranking. Mechanism engineer, knowing about market peculiarities (e.g., legal layout, kinds and number of entities, kinds and structure of resources) may indicate mechanism properties that are considered important in a given situation. In a liberal market system the criteria important for individual entity are usually disjoint from the global criteria, i.e. important for the society as a whole. Apart from mechanism evaluation according to the two above viewpoints, one may consider the third approach: mechanism technical efficiency.

4.1. Global criteria

Economic efficiency. Social welfare is considered the principal measure of a mechanism efficiency. Social welfare is the total of real economic benefits from the trade of commodities. If the user best strategy in a market mechanism is to bid according to his valuation, then the social welfare can be calculated using the prices offered by market participants. Otherwise, social welfare can be approximated by so called economic benefit, i.e. the difference of the total value of goods being bought and the total value of goods being sold, using transaction prices instead of the valuations.

Incentive compatibility. A mechanism is incentive compatible if a user best strategy is to announce all his private valuation information, i.e. if a user has no incentive to bid untruthfully. A mechanism is incentive compatible if users' truthful strategies are their best strategies, and therefore are the market game equilibria. Incentive compatibility prevents any individual or collective actions diverging from the optimal strategy. A good measure of such prevention is a so-called allocation inefficiency.

Budgetary balance. A market is in budgetary balance if the money flow from goods acquisition is equal to the flow from goods sales. This means that a market driven by a mechanism enforcing budgetary balance does not require any subsidy, neither it generates any surplus.

Market concentration. Entities that have considerable market share may influence the clearing process and jeopardize assumed mechanism properties. Herfindahl-Hirschman index (HHI) is used to measure market concentration; it is calculated by summing squared percentage market shares for all market entities.

Pareto efficiency. The game outcome is Pareto-efficient when it is possible to increase profits of one market entity only at the cost of the profit loss by some other entity or entities. In other words, the outcome of the game is not dominated in Pareto sense by any other outcome. It is particularly instructive to apply the term of Pareto efficiency to resource allocation problem. The resource allocation problem solution in market economy is a detailed register or description of what resource has been assigned to whom. Solution space is defined by the current state of technology and the amount of available resources in the economy. The final allocation solution depends always on customers' preferences. Therefore, for given preferences, technology and resources if an allocation is Pareto-efficient, it is impossible to find another allocation improving somebody's profits without spoiling someone else's profits.

4.2. Individual Criteria

Individual profit maximization. Every single market participant is interested that the market mechanism makes it possible to maximize participant's profit. On incentive compatible markets individual profit maximization do really takes place. However, society expectations are often that market prices should stay as low as possible, for the benefit of customers. Under such demand one can still design a mechanism for individual manufacturer profit maximization. It requires the original problem to be reformulated so that society expectations are considered superior to profit maximization – they can be, for example, treated as constraints to mechanism outcome.

Absolute fairness – individual rationality. A mechanism is considered to be absolutely fair when none of the players will incur individual loss, i.e. the player profit will be positive. In fact, this simple criterion preconditions the player participation in the market.

Individual relative fairness. A mechanism is considered to be relatively fair from one's point of view if the other competitive offers are not favored at his/her costs. This broad term encompasses more specific criteria:

- Anonymity. Market players are treated anonymously if the order of their numbering does not influence the outcome.
- Symmetry. Any two players characterised by equal parameter values, i.e. players having the same preferences (in the sense of their utility functions) and capabilities (in the sense of quality, amount and geographical availability of services offered) should be given equal allocations outcome.
- Price uniformity. A mechanism is fair if the price of a service is equal for all customers.

4.3. Mechanism Technical Efficiency

The most important criterion of mechanism technical efficiency is the possibility of the mechanism to be deployed and successfully used. Successful implementation of a mechanism depends on complexity of the underlying algorithms and on algorithms robustness. Typical measures characterizing mechanism technical efficiency are:

- market clearing time,
- total market clearing time (for iterative mechanisms),
- number of exchanged messages (average per user),
- number of lost messages (i.e. messages that did not count in the process of market clearing).

5. Conclusion

The platform for research on auction mechanisms in its current state of development should be perceived as a groupwork environment used for design and simulation verification of resource allocation mechanisms. Genericity of PRAM algorithms for experiments data selection and evaluation have implied use of somewhat prolix M^3 data format and clumsiness of graphical user interface. The next step in PRAM development is to implement user agents being operated by a human. This will make possible to run simulation scenarios where some market users will be played by e.g. students, being confronted with each other as well as with automated software agents.

In the long run, PRAM commercialization can be considered. Although the platform has already adopted a number of concepts and technologies used in enterprise applications (tiered architecture, JMS, Hibernate [16], GWT [17]), making it a fully-fledged business application requires adding many new functionalities. They include: authentication and authorization, scalability, repository protection, SLA guarantees etc. The architectural solutions implemented so far in PRAM have been selected deliberately to facilitate such transition.

Acknowledgement

The research presented in this paper was partially supported by Polish Ministry of Science and Higher Education grant PBZ-MNiSW-02/II/2007-LUB.

References

[1] W. Stańczuk, P. Pałka, J. Lubacz, and E. Toczyłowski, "A framework for evaluation of communication bandwidth market models", *J. Telecom. Inform. Technol.*, no. 2, pp. 52–60, 2010.

[2] "ISAS briefing sheet", Federal Communications Commission, 7th March 2011 [Online]. Available: http://wireless.fcc.gov/auctions/conferences/combin2003/papers/ISASbriefingsheet.pdf

[3] "Gpw.pl – Trading system", 7th March 2011 [Online]. Available: http://www.gpw.pl/trading_system

[4] "IT System – Towarowa Giełda Energii", 7th March 2011 [Online]. Available: http://www.polpx.pl/en/20/it-system

[5] G. Giammarino, J.-F. Huard, and N. Semret, "Merkato: a platform for market-based resource allocation", in *Proc. Networking 2000, Intelligent Agents for Telecommunications Environments*, Paris, France, 2000.

[6] "PeerMart: A Decentralized Auction-based P2P Market", 7th March 2011 [Online]. Available: http://www.peermart.net

[7] M. Placek and R. Buyya, "Storage exchange: a global trading platform for storage services", in *Euro-Par 2006*, LNCS 4128, W. E. Nagel et al., Eds. Springer, 2006, pp. 425–436.

[8] D. M. Turner, V. Prevelakis, and A. D. Keromytis, "The bandwidth exchange architecture", Tech. Rep. DU-CS-04-07, Drexel University, 2004 [Online]. Available: https://www.cs.drexel.edu/files/ts467/DU-CS-04-07.pdf

[9] P. Kacprzak, M. Kaleta, P. Pałka, K. Smolira, E. Toczyłowski, and T. Traczyk, "Communication model for M3 – open multicommodity market data model", in *Proc. TPD'2007 Conf.*, Poznań, Poland, 2007.

[10] P. Kacprzak, M. Kaleta, P. Pałka, K. Smolira, E. Toczyłowski, and T. Traczyk, "M3: Open multicommodity market data model for network systems", in *Proc. XVI Int. Conf. Sys. Sci.*, Wrocław, Poland, 2007.

[11] P. Kacprzak, M. Kaleta, P. Pałka, K. Smolira, E. Toczyłowski, and T. Traczyk, "Application of open multicommodity market data model on the communication bandwidth market", *J. Telecom. Inform. Technol.*, no. 4, pp. 45–50, 2007.

[12] P. Pałka, K. Kołtyś, E. Toczyłowski, and I. Żółtowska, "Model for balanced aggregated communication bandwidth resources", *J. Telecom. Inform. Technol.*, no. 3, pp. 43–49, 2009.

[13] M. Karpowicz and K. Malinowski, "Network Flow Optimization with Rational Agents", NASK internal rep., 2009.

[14] M. Kamola, "Software environment for market balancing mechanisms development, and its application to solving more general problems in parallel way", in *Proc. PARA 2010 Conf.*, Reykjavik, Iceland, 2010.

[15] C. Walls and R. Breidenbach, *Spring in Action*. Manning Publications Co., 2005.

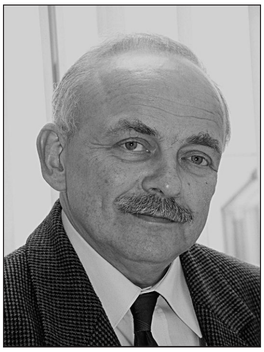
[16] Ch. Bauer and G. King, *Hibernate in Action*. Manning Publications Co., 2005.

[17] R. Cooper and Ch. Collins, *GWT in Practice*. Manning Publications Co., 2008.

Mariusz Kamola received his Ph.D. in Computer Science from the Warsaw University of Technology in 2004. Currently he is Associate Professor at Institute of Control and Computation Engineering at the Warsaw University of Technology. Since 2002 with Research and Academic Computer Network (NASK). His research area focuses on economics of computer networks and large scale systems.

E-mail: mkamola@ia.pw.edu.pl
Institute of Control and Computation Engineering
Warsaw University of Technology
Nowowiejska st 15/19
00-665 Warsaw, Poland

E-mail: Mariusz.Kamola@nask.pl
Research and Academic Computer Network (NASK)
Wąwozowa st 18
02-796 Warsaw, Poland



Krzysztof Malinowski Prof. of techn. sciences, D.Sc., Ph.D., MEng., Professor of control and information engineering at Warsaw University of Technology, Head of the Control and Systems Division. Once holding the position of Director for Research of NASK, and next the position of NASK CEO. Author or co-author of four books and over 150 journal and conference papers. For many years he was involved in research on hierarchical control and management methods. He was a visiting professor at the University of Minnesota; next he served as a consultant to the Decision Technologies Group of UMIST in Manchester (UK). Prof. K. Malinowski is also a member of the Polish Academy of Sciences.

E-mail: K.Malinowski@ia.pw.edu.pl
Institute of Control and Computation Engineering
Warsaw University of Technology
Nowowiejska st 15/19
00-665 Warsaw, Poland

E-mail: Krzysztof.Malinowski@nask.pl
Research and Academic Computer Network (NASK)
Wąwozowa st 18
02-796 Warsaw, Poland

Wojciech Stańczuk received his M.Sc. in Telecommunications from the Warsaw University of Technology in 2001. Now he is Ph.D. student and research assistant in the Institute of Telecommunications at Warsaw University of Technology. His scientific interests cover techno-economic aspects of telecommunication networks operation, including resource allocation and pricing as well as strategies for infrastructure investments.

E-mail: w.stanczuk@tele.pw.edu.pl
Institute of Telecommunications
Warsaw University of Technology
Nowowiejska st 15/19
00-665 Warsaw, Poland

Piotr Pałka is an Assistant Professor of the Operations and Systems Research Division in the Institute of Control and Computation Engineering at the Warsaw University of Technology, Poland. He received the M.Sc. and Ph.D. degrees in computer science in 2005 and 2009, respectively, both from Warsaw University of Technology. His research interest focus on the market mechanisms, especially the incentive compatibility on the infrastructure markets and on the multi-agent systems. His current research is focused on application of multicommodity turnover models.

E-mail: P.Palka@ia.pw.edu.pl
Institute of Control and Computation Engineering
Warsaw University of Technology
Nowowiejska st 15/19
00-665 Warsaw, Poland

Ewa Niewiadomska-Szynkiewicz – for biography, see this issue, p. 10.