International Transactions on Systems Science and Applications
Volume 1 Number 4 (2006) 375-380

ITSSA
www.xiaglow-institute.org.uk/itssa/

Evaluation of a Multi-Agent System for Hospital Patient Scheduling

Anja Zöller [1] Lars Braubach [2] Alexander Pokahr [2] Franz Rothlauf [1]
Torsten O. Paulussen [1] Winfried Lamersdorf [2] Armin Heinzl [2]

[1] Department of Business Administration and Information Systems, University of Mannheim, D-68131 Mannheim, Germany
Email: {zoeller; rothlauf; paulussen; heinzl}@uni-mannheim.de
[2] Department of Computer Science, Distributed and Information Systems, University of Hamburg, D-22527 Hamburg, Germany
Email: {braubach; pokahr; lamersdorf}@informatik.uni-hamburg.de

Abstract: The problem of patient scheduling in hospitals is characterized by high uncertainty and dynamics in patient treatments. Additional complexity in the planning and coordination processes is caused due to interdependencies of autonomous and administratively distinct units which are involved in the treatment of a patient. For real-world scheduling scenarios traditional scheduling methods are often either too limited in their expressive power regarding the representation of real-world problems or fail in solving real-sized problems in a timely fashion. In contrast, multi-agent systems are a promising approach to overcome these restrictions. This paper extends previous evaluations a multi-agent system for patient scheduling and studies a close-to-reality testing environment. The scenario is based on a field study and includes the interplay of multiple sources of uncertainty to evaluate the applicability of our approach in practice. The experimental results show that the evaluated multi-agent system outperforms existing status quo approaches for patient scheduling in hospitals.

Keywords: collaborative business processes, service-oriented architectures, agent technologies.

1. Introduction

Hospitals are service providers with the primary aim to improve the health state of their patients, where the treatment of the patients is the main value-adding-process in hospitals [7][11]. Hospitals consist of several autonomous, administratively distinct wards and ancillary units [5][16][22]. During hospitalization, the patients reside at the wards and visit the ancillary units for treatments according to their individual disease. The patient scheduling is concerned with the (optimal) assignment of medical tasks for the patients to the (scarce) hospital resources [30]. However, hospital patient scheduling is confronted with such a high degree of uncertainty that Schlüchtermann and Gierl [9][30] assess a short-time planning horizon of only one day whereas, for manufacturing control, Wöhe [34] assumes a short-time planning horizon of one to two weeks. In hospital patient scheduling, the patients arrive continuously at the hospital and the necessary medical treatments are often not completely determined at the beginning of the treatment process. Moreover, the new findings during diagnostic examinations change the (medical) priority of the patients, invoke additional treatments or examinations, or make other medical actions obsolete [22]. Furthermore, the durations of treatments and examinations are stochastic [1][23][29]. Finally, complications and arrivals of emergency patients which are in urgent need for treatment result in schedule disturbances. Because the ancillary units only have a local view and can not consider the complete pathway of the patients, no inter-unit process optimization is possible (i.e., the medical tasks for the patients cannot be scheduled and coordinated in an efficient manner). This causes undesired idle times as well as overtime hours for the hospital resources and extended patient stay times.

Concerning such real-world scheduling problems, classical AI and OR-based methods are often either too limited in expressive power regarding the representation of real world problems or lead to intractable problems with formalisms failing to solve real-sized problems in a timely fashion [10][31][32]. These approaches lack properties like flexibility, adaptivity, and reactivity, being based on methods that neglect dynamic changes and disturbances during a fixed planning period.

Multi-agent systems (MAS) are a promising approach to overcome such restrictions by providing properties like autonomy, reactivity and proactivity [15][35]. MAS are supposed to be suitable for real-world problems that have a special need for flexibility and adaptivity to dynamic changes and that have a decentralized planning structure. Meanwhile, different multi-agent systems have been designed that address problems of dynamic environments and disturbances in scheduling problems.

Liu and Sycara [18] developed a MAS to solve job-shop scheduling problems, requiring real-time scheduling and execution. They decomposed the job-shop scheduling problem and distributed it on job-agents and resource-agents where each agent solves its sub-task. The coordination of the partial solutions is provided by constraint partition & coordinated reaction. Each agent communicates its results to affected agents and reacts to violations of restrictions.

Brennan and O [3] modeled a manufacturing system with job agents, machine agents, station agents (responsible for a workstation containing several homogeneous machines), and a mediator agent which, similar to a yellow page agent, provides information on which resource can perform a particular type of operation. Four different coordination strategies, based on the contract-net protocol and on auction-based bidding, are tested. The change in solution quality is evaluated while increasing

ISSN 1751-147X cd-rom / ISSN 1751-1461 print
xai: itssa.2006.09.044

the number of jobs to be scheduled.

Lee et al. [17] considered agent societies of different structure and compared the performance of a hierarchical and a mesh structure with different coordination protocols. The MAS has to perform a task where all agents must work together in order to achieve it. The task can be split up into several smaller subtasks, for which each agent can bid for. The paper studies the changes in computational time when increasing the number of tasks or number of agents.

Huang et al. [14] designed a MAS for distributed medical care, facing challenges regarding the distribution of data and control, information uncertainty, and environment dynamism. The coordination mechanism is based on commitments and conventions between different types of agents. The task allocation and coordination is done by managing agents that manage the execution of tasks and by contractor agents that execute the task.

Decker and Li [4][5] modeled a MAS for hospital patient scheduling with complex medical procedures. They took a function-centered view and modeled nursing wards as autonomous agents. They developed a generalized partial global planning (GPGP) approach as a constraint-based coordination mechanism. It is constructed to avoid resource conflicts and patients are treated as exclusive resources that are handled by a special mechanism.

To evaluate the developed MAS, in many cases feasibility proofs or proofs of concepts were performed addressing aspects like scaling problem size, uncertainty, or the dynamics of the domain. However, especially for hospital patient scheduling, the question is important whether these MAS approaches can be successfully applied in real-world applications, where several aspects of uncertainty and dynamic environments occur together. Therefore, further evaluation studies have to be done, examining the applicability of the developed coordination mechanisms and MAS architectures in uncertain and real-world scenarios.

Important for evaluation studies is the selection of adequate test problems [13], as there is a trade-off between abstract, simplified testbeds, and testbeds that are close to real-world environments. For abstract testbeds it is easier to generate generalizable results. However, obtained conclusions often do not hold for real-world problems. Therefore, only testbeds that are close to reality allow meaningful conclusions on the applicability of such approaches in real-world applications [12]. For the existing standard benchmarks, Hanks et al. [12] identified the problem that scientific progress may only focus on the ability to better solve these abstract benchmarks instead of bringing progress for real problems.

It is the aim of this paper to perform a close-to-reality evaluation of the MAS proposed by Paulussen et al. [21][22] which has been developed for hospital patient scheduling. The evaluation will be conducted in a simulation study that is based on empirical data from a field study on patient scheduling in a German hospital.

The paper is structured as follows. Section 2 describes the design of the MAS proposed by Paulussen et al. [21][22]. We review the underlying coordination mechanism and the implementation of the MAS. Section 3 provides the test scenario that was developed based on the data of our field study. Finally, in Section 4, the MAS approach is tested against the status quo of planning and the experimental results are presented.

2. Conceptual Framework and Implementation

2.1 Conceptual Framework

The multi-agent system of Paulussen et al. [21] represents patients and ancillary units as patient-agents and resource-agents. The patient-agents compete for treatment appointments as scarce resources in a fictitious market place. For that purpose the resource-agents auction off the time-slots corresponding to their capacity. Consequently, if a resource gets free, its next time-slot is assigned to the patient-agent with the highest bid. Each treatment represents a utility for an agent by improving his health state or avoiding a worsening. Thus, based on an individual worth-function each patient agent determines the benefit of a treatment as the price, it is willing to pay for it. The rationality behind this approach is that the patient-agent who gains the highest utility from a specific time-slot is willing to pay the highest price for it (up to the expected utility). To specify these individual worth functions of patient-agents, health state dependent cost functions are introduced, as the priority of the patients is determined by their health condition. In doing so the illness of a patient is viewed as dis-utility (decrease in quality of life) [22]. For the necessary cardinal measurement of health, Paulussen et al. use the concept of years of well being [24][33],which allows to describe the health state progress over time. Because the loss of utility adds up as long as the illness is not cured, this dis-utility is interpreted as opportunity costs for not curing the disease right away [22]. Moreover, the health state of a patient can either remain constant or can decrease over time. Therefore, a health decrease rate is included in the worth function of a patient that defines the stability of a health state.

2.2 Implementation

The prototype implementation is organized in three separate layers: The coordination layer, the hospital layer, and the infrastructure layer [21]. The coordination layer is comprised of the different coordination mechanisms, each of which can be applied to perform the treatment scheduling. The coordination mechanism has been designed and implemented using agent-oriented tools and concepts. More details about the implementation of the coordination mechanisms can be found in [2][23]. The hospital layer is designed to support the execution of the coordination by providing the facilities to perform simulation runs or to run the system as an application. When a simulation run is initiated, the information from the hospital model is used to create the hospital infrastructure consisting of initial patient and resource agents. During the run, the system agent uses different random distributions to approximate real arrival rates of patients and other occurrences like emergencies. Using this information, the system can decide when the next arrival or emergency will take place. The system agent is conceived to emulate all simulation external occurrences. Hence, for running the system as application instead of simulation it is merely required to adapt the system agent to react on some user interface and setup the time service with real time. The infrastructure layer provides system-level services for the implementation such as agent management and execution, as well as persistency. Basic agent services as the agent life cycle management, agent communication and search facilities are provided by a FIPA-compliant agent middleware platform

[26]. These basic services are enhanced with a rational agent layer following the BDI-metaphor [28] which enables the use of goal-oriented concepts at the design and implementation level. Hence, it facilitates the development with the introduction of high-level agent-oriented programming concepts [25]. The persistency infrastructure consists of a relational database management system which is connected with an object-relational mapping layer. The mapping layer enables object-oriented access to the data by making the underlying relational database model transparent.

3. Test Scenario

This section describes the test scenario that we generated based on a field study in a German hospital. The test scenarios is implemented in a simulation environment and is based on a data sample containing 3,448 data sets with information on medical tasks for 792 inpatients from admission to release.

In continuation to the preceding evaluations of the multi-agent approach from Paulussen et al. that focused on ceteris paribus examination of certain disruptive factors [21], the aim of this scenario is to build a close-to-reality testing environment that includes the interplay of multiple sources of uncertainty. Thus, it allows to judge the applicability of this approach in practice. The following sections describe the test scenario in some more detail.

3.1 General Hospital Layout

Our testbed is characteristic for a medium-sized German hospital with about 500-600 inpatient beds and 12 diagnostic resources in five different ancillary units. There are four parallel diagnostic work stations for endoscopy, three for the circulation laboratory unit, three for X-rays, one station for computed tomography and magnetic resonance imaging (CT/MR unit), and one station for nuclear medicine. Altogether, these units offer about 300 different medical examinations and treatments. In our model, we considered variable processing times for the different services in the ancillary units that are based on the data of our field study.

3.2 Patient Arrival

For a standard working day, the average amount of patient arrivals is 80 to 90 patients per day including an emergency rate of 5% of all arriving patients. The patient arrivals are exponentially distributed over the day and a patient gets the first medical assignment right after his or her admission. The further course of the patient treatment follows the hospital process organization with daily ward rounds in the morning. This situation leads to an accumulation of assignments at the beginning of a day and results in a peak load the scheduling system has to cope with.

3.3 Health States and Decrease Rates

In our field study, we observed a classification of patients along their severity of illness and the stability of their constitution. We modeled this classification by using different health states and health decrease rates following [21]. Thus, a health state of 1 represents full health and a health decrease rate of 0 is a stable constitution, whereas a health decrease rate of 0.0017 characterizes an instable constitution of an emergency patient, indicating the need for immediate treatment.

3.4 Clinical Pathways

At first glance one might expect that patients with the same diagnosis merely get the same treatment and that most of the assignments may be predicted with a probability close to 100%. However, the data of our field study indicates very inhomogeneous treatments. The probabilities of the top ten assignments regarding a certain diagnosis showed a median of 36% reaching from 9% to 81% maximally. These variations result form the uncertainty of the diagnosis at the beginning of a treatment and later on from necessary patient individual medical tasks. Thus the amount, type, and sequence of medical tasks that are assigned to the patients are uncertain and each patients' clinical pathway individually varies with new findings from previous examinations. This effect is represented in our simulation by iterative treatment phases [8]. After the admission, the patient is either in the state of treatment or released. After each ward round the patient turns to the next treatment phase again, or if treatment is completed he or she is released. The state transition properties depend on the iteration number of a patient and with increasing number of treatment phases the probability of a release increases. Thus, we model the patient treatment as a Markov process with history [6][27], see Fig. 1(a).

The assignments (examinations, treatments) for a patient in a ward round depend on his or her current treatment phase. For example, the first phases are characterized by admission examinations to clear the patients' diagnosis, whereas special curative treatments and checking examinations occur in phases with a higher iteration number. Thus, we refine the model for treatment phases in a patients' clinical pathway, representing it as an extended GERT-network [19][20], see Fig. 1(b). The respective course of medical procedures a patient has to pass and the diagnostic treatments to be scheduled are calculated based on our empirical data.

This concept of treatment phases allows us to consider variable and uncertain clinical pathways. It includes uncertainty from random effects in the progress of patient treatments which results in inherent dynamics and stochastic load for a hospital. For example, hospitals may experience capacity bottlenecks if casually a high amount of current inpatients turns out to need a higher intensity of care as usual.

3.5 Simulation Runs and Target Measures

The simulation runs were set up for 22 days and started with a hospital that was pre-filled with patients. The first 12 days were assumed as a tuning phase of the system and therefore were not evaluated. As target measures we evaluate the system performance, the resource occupation of the ancillary units, and the waiting times of the patients. We compare the results of the MAS approach from Paulussen et al. to the status quo in hospital patient scheduling. Because hospital patient scheduling is an ad hoc approach [5], patients usually do not get a prefixed time-slot for treatment, but rather wait in a queue (or in the ward unit) until they are requested by the ancillary unit. Thus, no scheduling data allowing a more detailed comparison is available. Therefore, we model the status quo of patient scheduling as a first-come first-serve (FCFS) scheduling strategy that well describes the situation in

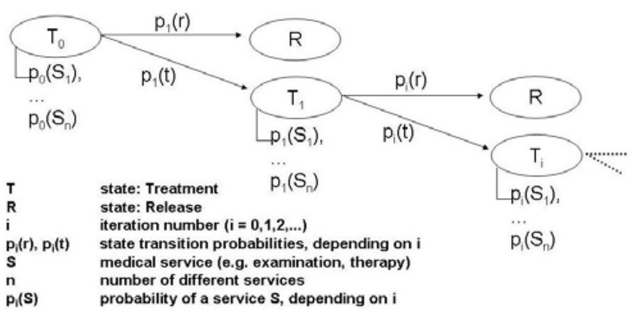

Fig. 1(a). Patient treatment as Markov process.

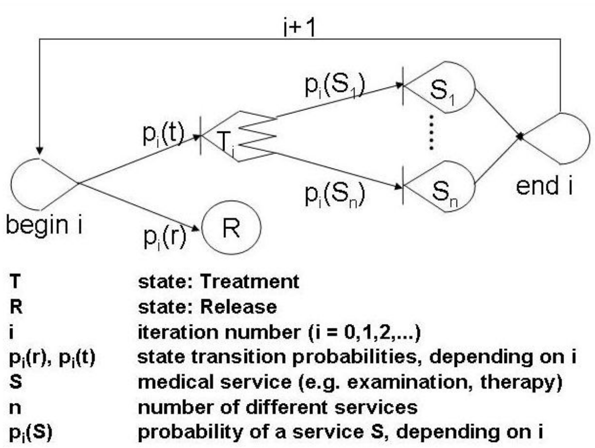

Fig. 1(b). GERT model of patient treatment phases.
Fig. 1. Modeling clinical pathways in the real-world scenario.

real world.

4. Experimental Results

4.1 System Performance

In all test runs, the MAS proposed by Paulussen et al. showed satisfactory performance for real-time demands. This holds even for high load situations in hospital wide ward rounds. On average the negotiations took 1.500 ms with a standard deviation 5 ms. The maximal negotiation time that occurred was 31.009 ms.

4.2 Patient Waiting Times

To evaluate the average patient waiting time, we calculate the time between the assignment of a medical task and its execution. Fig. 2 compares for ten test runs the average patient waiting times for the FCFS scheduling with the auction-based coordination approach from Paulussen et al. The auction-based approach permanently results in lower waiting times than the FCFS heuristic. On average it outperforms the status quo approach resulting in patient waiting times that are about 40% lower. The reduction is a result of the interaction between resource and patient agents which in opposite to the status quo scheduling allows a more efficient inter-unit coordination.

Furthermore, we examine the distribution of patient waiting times over different health states. Fig. 3 shows how the deviation of average waiting times (in percent) depends on the health state of a patient. With an FCFS scheduling heuristic, the patient waiting times are independent from the patients health state and are similar to the average waiting times. In contrast, for the auction-based approach (including health state dependent cost functions) the waiting time decreases with lower health state of the patient. Therefore, patients with severe illness are treated earlier in comparison to patient with a high health state.

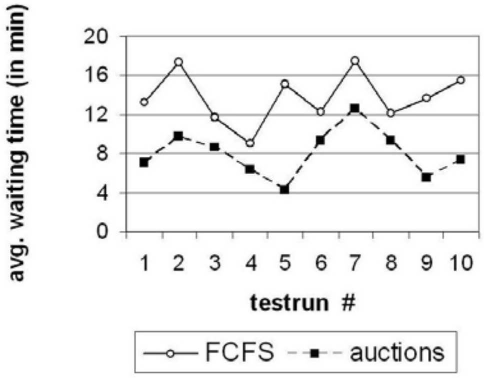

Fig. 2. Average patient waiting times.

Fig. 3. Deviation of average waiting time over health state.

4.3 Resource Occupation

The resource occupation is the opening time of the ancillary units. Each day, a resource opens in the morning and closes when all due medical tasks have been finished. Thus, a resource occupation of 1 means that a resource stays open for 9h. The goal of a hospital is to perform all scheduled medical treatments and to minimize the opening time of a resource. Fig. 4 compares for 10 test runs the average resource occupation of the auction-based MAS from Paulussen et al. to the FCFS planning heuristic. On average, the auction-based coordination shows a 13% lower resource occupation. When using this approach the ancillary units finish the due tasks earlier than in the status quo approach. This is due to lower resource idle times during the day and therefore leads to earlier closing times and less overtime for the ancillary units.

5. Summary and Conclusions

This paper evaluates the performance of a multi-agent system (MAS) [21][22] for patient scheduling in a real-world scenario. For the evaluation, we performed a field study and developed, based on the empirical data gathered in the field study, a realistic simulation model of a medium-sized hospital. When comparing the MAS approach proposed by Paulussen et al.

[21][22] to the status quo of patient scheduling in hospitals, we observed a good performance of the market-based MAS approach for the real-world scenario. First, the MAS allows real-time scheduling and satisfies real-time demands even in high-load situations which typically occur in hospital-wide ward rounds in the morning. Second, the quality of existing schedules can be improved by the decentralized marked-based coordination. In comparison to the current status quo in hospital patient scheduling, the MAS approach reduces patient waiting times and resource occupation. Furthermore, by considering the health state of the patients for scheduling, patients with severe diseases have on average lower waiting times in comparison to patients with a better health state. Thus, it balances patient waiting times according to health state of the patients and may contribute to a higher satisfaction of the patients.

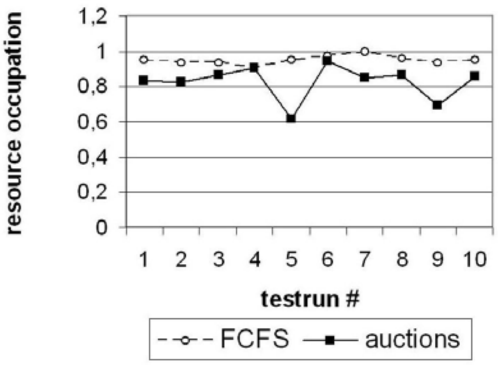

Fig. 4. Resource occupation times.

Along with the test scenario, a concept of variable clinical pathways was developed. For the future, we are planning to incorporate the concept of variable clinical pathways directly into the patient agents. This concept can serve as a knowledge base for a prognosis on the further course of a patients' individual treatment and thus, improve schedule quality in uncertain environments.

Acknowledgments

We would like to thank the Deutsche Forschungsgemeinschaft for funding this work under the project title "Agentenbasierte Planung und Koordination funktionsübergreifender Aktivitäten in medizinischen Behandlungspfaden (MedPAge)" as part of SPP 1083.

References

[1] M Brahimi and D J Worthington, Queuing models for out-patient appointment systems a case study. Journal of the Operational Research Society, Vol. 42 1991, No. 9, pp. 733-746.
[2] L Braubach, A Pokahr and W Lamersdorf, Medpage: Rationale Agenten zur Patientensteuerung. Künstliche Intelligenz, Vol. 18 2004, No. 2, pp. 33-36.
[3] R W Brennan and W O, A simulation test-bed to evaluate multi-agent control of manufacturing systems. In J A Joines, R R Barton, K Kang and P A Fishwick, eds., Proceedings of the 2000 Winter Simulation Conference 2000, pp. 1747-1756.
[4] K Decker and J Li, Coordinated hospital patient scheduling. Proceedings of ICMAS'98, 1998, pp. 104-111.
[5] K Decker and J Li, Coordinating mutually exclusive resources using GPGP. Autonomous Agents and Multi-Agent Systems, Vol. 3, No. 2, 2000, pp. 133-157.
[6] C Derman, Finite State Markovian Decision Processes, Acad. Press, New York, 1970.
[7] R Feinen, Patientenbezogene Organisation von Behandlungsprozessen. S Eichhorn and B Schmidt-Rettig, eds., Profitcenter und Prozessorientierung: Optimierung von Budget, Arbeitsprozessen und Qualität, Vol. 8, Kohlhammer, Stuttgart, 1999, pp. 188-199.
[8] W Fischer, Patientenklassifikationssysteme zur Bildung von Behandlungsfallgruppen im stationären Bereich -Prinzipien und Beispiele, BSV+ZIM Verlag, Bern and Wolfertswill, 1997.
[9] L Gierl, Simultane Patientendurchlaufplanung für mehrere Leistungsstellen. M Meyer, ed., Krankenhausplanung, Fischer, Stuttgart, 1979.
[10]C P Gomes, Artificial intelligence and operations research: Challenges and opportunities in planning and scheduling. The Knowledge Engineering Review, Vol. 15, No. 1, 2000, pp. 1-10.
[11]A Greulich, G Thiele and M Thiex-Keye, Prozessmanagement im Krankenhaus, Schriftenreihe zum Managementhandbuch Krankenhaus, Vol. 8, Decker, Heidelberg, 1997.
[12]S Hanks, M E Pollack and P R Cohen, Benchmarks, testbeds, controlled experimentation, and the design of agent architectures. AI Magazine Vol. 14, No. 4, 1993, pp. 17-42.
[13]A Howe and E Dahlman, A critical assessment of benchmark comparison in planning. Journal of Artificial Intelligence Research, Vol. 17, 2002, pp. 1-33.
[14]J Huang, N R Jennings and J Fox, An agent architecture for distributed medical care. In M Wooldridge and N R Jennings, eds., Intelligent Agents: Theories, Architectures, and Languages, LNAI, Springer-Verlag: Heidelberg, Germany, Vol. 890, 1995, pp. 219-232.
[15]N R Jennings, K P Sycara and M Wooldridge, A roadmap of agent research and development. Autonomous Agents and Multi-Agent Systems, Vol. 1, No. 1, 1998, pp. 7-38.
[16]A Kumar, P S Ow and M J Prietula, Organizational simulation and information systems design: An operations level example. Management Science, Vol. 39, No. 2, 1993, pp. 218-240.
[17]L Lee, H Nwana, D Ndumu and P De Wilde, The stability, scalability and performance of multi-agent systems. BT Technology Journal, Vol. 16, No. 3, 1998, pp. 94-103.
[18]J-S Liu and K P Sycara, Multiagent coordination in tightly coupled task scheduling. Readings in Agents, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1998, pp. 164-171.
[19]K Neumann and U Steinhardt, GERT-Networks and the Time-Oriented Evaluation of Projects, Springer, Berlin and New York, 1979.
[20]K Neumann, Scheduling of projects with stochastic evolution structure. In J Weglarz, ed., Project Scheduling: Recent Models, Algorithms and Applications, Kluwer Academic Publishers, 1999, pp. 309-332.
[21]T O Paulussen, A Zöller, A Heinzl, L Braubach, A Pokahr and W Lamersdorf, Agent-based patient scheduling in hospitals. In S Kirn, O Herzog, P C Lockemann and O

Spaniol, eds., Multiagent Engineering Theory and Applications in Enterprises, Springer Verlag, Heidelberg, Germany, 2006, pp. 255-276.

[22]T O Paulussen, N R Jennings, K S Decker and A Heinzl, Distributed patient scheduling in hospitals. In G Gottlob and T Walsh, eds., Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence (IJCAI'03), 2003, pp. 1224-1232.

[23]T O Paulussen, A Zöller, A Heinzl, A Pokahr, L Braubach and W Lamersdorf, Dynamic patient scheduling in hospitals. In M Bichler, C Holtmann, S Kirn, J Müller and C Weinhardt, eds., Coordination and Agent Technology in Value Networks, GITO Berlin, March 2004, pp. 34-45.

[24]G Pedroni and P Zweifel, Wie misst man Gesundheit?, 1990.

[25]A Pokahr, L Braubach and W Lamersdorf, Jadex: A BDI reasoning engine. In R Bordini, M Dastani and A Dix, J Seghrouchni, eds., Multi-Agent Programming, Kluwer, 2005, pp. 149-174.

[26]S Poslad and P Charlton, Standardizing agent interoperability: The FIPA approach. In M Luck, V Marík, O Stepánková and R Trappl, eds., Lecture Notes in Computer Science, Multi-Agent Systems and Applications, 9th ECCAI Advanced Course ACAI 2001 and Agent Link's 3rd European Agent Systems Summer School, EASSS 2001, Prague, Czech Republic, 2-13 July 2001, Selected Tutorial Papers, Springer, Vol. 2086, 2001, pp. 98-117.

[27]M L Puterman, Markov Decision Processes-Discrete Stochastic Dynamic Programming, Wiley, New York u.a., 1994.

[28]A S Rao and M P Georgeff, BDI agents: From theory to practice. Proc. of 1st International Conference on Multi-Agent Systems (ICMAS'95), 1995, pp. 313-319.

[29]L W Robinson and R R Chen, Scheduling doctors' appointments: Optimal and empirically-based heuristic policies. IIE Transactions, Vol. 35, 2003, pp. 295-307.

[30]J Schlüchtermann, Patientensteuerung, Verlag Josef Eul, Bergisch Gladbach, 1990.

[31]B Scholz-Reiter and M Freitag, Nichtlineare Dynamik und PPS. PPS Management, Vol. 3, 2001, pp. 34-37.

[32]A Singh and Jain S Meeran, A state-of-the-art review of job-shop scheduling techniques. Tech. report, Department of Applied Physics, University of Dundee, Scotland, 1998.

[33]G W Torrance, Utility approach to measuring health-related quality of life. Journal of Chronic Diseases, Vol. 6, 1987, pp. 493-500.

[34]G Wöhe, Einführung in die allgemeine Betriebswirtschaftslehre, Vahlen, München, 2005.

[35]M Wooldridge and P Ciancarini, Agent-oriented software engineering: The state of the art. In P Ciancarini and M Wooldridge, eds., Lecture Notes in Computer Science, First Int. Workshop on Agent-Oriented Software Engineering, Springer-Verlag, Berlin, Vol. 1957, 2000, pp. 1-28.