In C. Voudouris, G. Owusu, R. Dorne & D. Lesaint (ed.), Service Chain Management, Springer, 2008, 263-272

19. Multi-agent based systems for staff empowerment

19.1 Introduction

Human resources are the main resources in a service industry. Success or failure of a service operation is often determined by its personnel management policy. A successful management policy would provide job satisfaction to employees, which will lead to higher morale, productivity and service quality.

Staff empowerment is a management concept. The idea is to give employees autonomy in doing their jobs. The aim is to improve job satisfaction by allowing employees to have control over their operations.

For staff empowerment to work, appropriate management arrangements are required. The philosophy behind staff empowerment recognizes personal needs by individuals. To ensure that the collective behaviour of the staff achieves the company's goals, the management must define the authority and responsibilities for each individual or department/division. The assumption behind staff empowerment is that by defining negotiation protocol and assessment criteria properly, the market mechanism will ensure that the staff gains autonomy while the company achieves its goals. In other words, everybody wins.

This Chapter presents a framework for designing a mechanism to implement empowerment in BT's workforce scheduling. It identifies the computational techniques for tackling the problem. BT's problem is used for illustration. However, the framework and techniques are general and therefore could be used for other job-staff allocation activities, where staff empowerment is to be employed.

19.2 BT's Workforce Scheduling Problem

The importance of problem formulation is often underestimated [2][7]. Modeling formalizes the company's considerations; i.e. what are considered important by the company. It also defines the company' objectives, i.e. what the company wants to achieve [24]. In this section, we shall briefly describe BT's scheduling problem. Detailed description of the problem can be found in [27][32]. Formal definition of the problem can be found in [25].

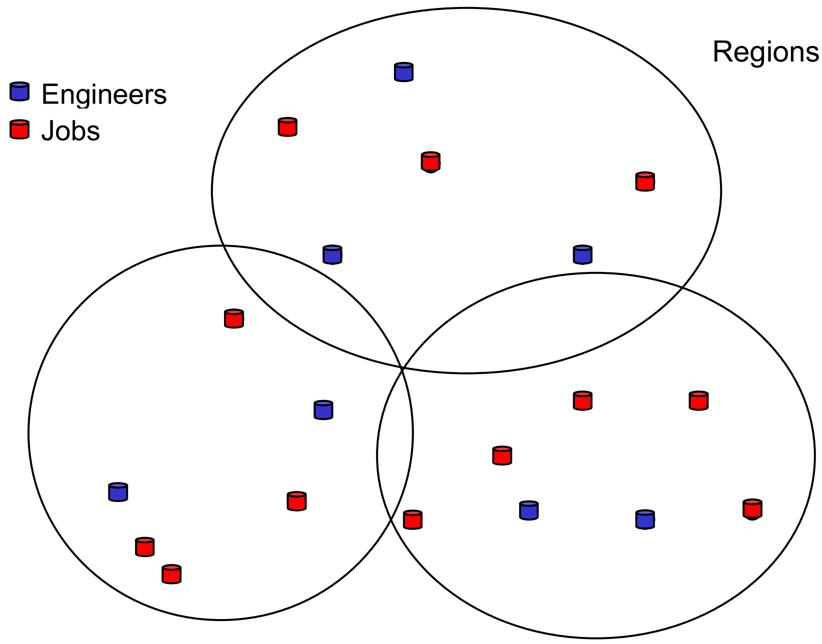

Figure 20.1 A generic workforce scheduling problem, where the task is to assign engineers to jobs, satisfying a variety of constraints and optimizing a set of company objectives

BT has to serve a large number of customers every day. Their needs vary from telephone repairs to network design and installation. Engineers also have to be sent for repair, maintenance and installation of BT's networks. These jobs are geographically distributed. Each job demands engineers of certain skills. Some jobs will take longer time to complete than others – the duration of a job can be estimated through past experience on similar jobs. Some jobs can only be done at certain times. Logically, each job can be assigned a value (how important it is to the company).

To serve the jobs, BT has a large number of engineers. The engineers are also geographically distributed (some of them may start their working days from home). Each engineer may be more skillful on certain types of jobs, but less skillful in others. Therefore, for each engineer, one can logically assign a “preference” to each skill, indicating how efficient they are in serving jobs which require such skill. It is in BT and the customers’ interest that the engineers are sent to jobs which they specialize in; i.e. maximizing the “preference” values in job assignments.

The task is to assign these engineers to the jobs, subject to their availability (e.g. some engineers may be willing to work overtime, but others may not). Whenever possible, BT would like to minimize the distance traveled by the engineers. Figure 20.1provides a visual description of the workforce scheduling problem.

19.3 How to achieve the goals?

BT’s problem as formulated above is a complex problem. The manager’s job is to look after the company’s interest, which is a *multi-objective optimization problem* [4]. Each regional controller has its own scheduling problem, which is technically a constraint satisfaction problem [23]; in particular, the problem can be seen as a *constrained multi-vehicle routing problem* [18]. Although BT’s problem is used as an example, the approach presented here is general – it can be applied to similar situations.

The workforce scheduling problem described above can be tackled through central planning [26]. However, field engineers can often create delays if they want to. To protect their own interest, controllers may not help other regions freely. One of the best ways to reduce game playing is through improvement of morale. Staff empowerment is an effective management strategy to maintain morale in the workforce.

To implement staff empowerment, the regional controllers are given permission to look after their own interests. This means the problem will be formulated as a *distributed scheduling problem* in which the individual agents have their individual goals (as opposed to having shared goals, as in [16]).

To tackle this distributed scheduling problem, our approach is to model the regional controller's activities with agents. We define a *buyer agent* and a *seller agent* for each controller. The buyer agent handles the jobs that the controller has. Its task is to "buy" services to complete the jobs. The seller agent handles the engineers. Its task is to "sell" services to complete the jobs.

We define a *management agent* that looks after the company's interest. Ultimately, the management agent will be a program that interacts with the human manager who is in charge of the overall operation. We give the management agent the duty of handling the multi-objective optimization problem. Details of this will be described in the next section.

19.4 Handling multi-objectives

One approach to multi-objective optimization is to define mathematically the relative importance of the multiple objectives. This turns the problem into a single-objective optimization problem. One major drawback of this approach is that human managers are often reluctant to define the relative importance of the multiple objectives in abstract, mathematical terms (either due to sheer difficulties or due to their unwillingness to commit themselves). It is, however, relatively easier for one to express one's preference when one is given a few schedules. Therefore, we define the goal of the management agent as to generate a Pareto set of schedules for the human manager to choose.

Our approach is to give the management agent the task of finding a Pareto set of schedules, which will be presented to the human agent for selection. The Pareto set is generated by iterations. In each iteration the management agent provides the buyer and seller agents the weights for each of their objectives. Thus the buyers and seller agents each have a single-objective scheduling problem to solve. They interact with each other (to be explained later) to generate a schedule. The weights-definition and scheduling-generation process repeats until enough number of schedules have been generated. This is shown in Figure 20.2.

This approach allows us to neatly separate multi-objective optimization from the rest of the problem. It also reflects the management structure.

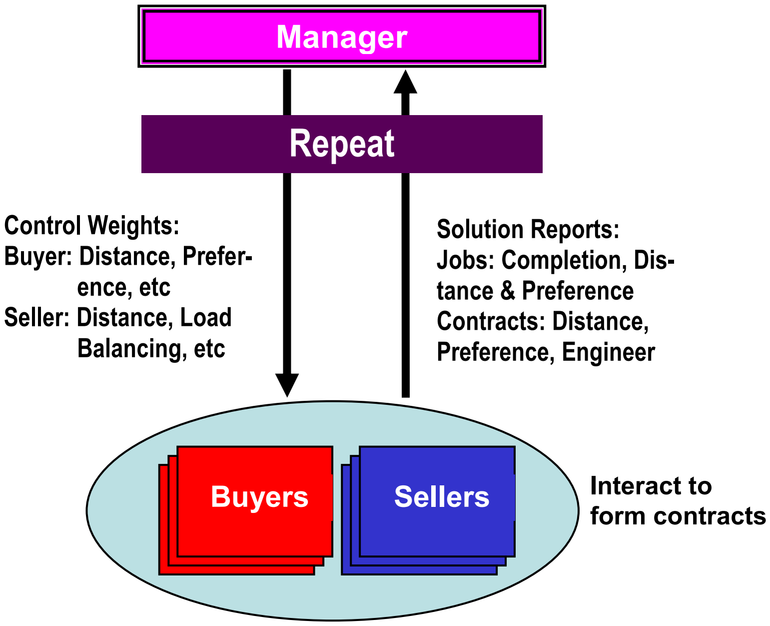

Figure 20.2 A multi-agent based architecture for workforce scheduling

19.5 How to generate a Pareto set of schedules?

Given the architecture defined above, the key question is how the management agent should adjust the weights for the buyer and seller agents.

Before we can answer this question, we need to decide on the metric for measuring the quality of a Pareto set. We combine two metrics in our approach: *range* and *evenness* of distribution. Formal definitions of these metrics can be found in the literature, e.g. see [15].

For illustration, let us assume that we have two objectives; this can be represented by two functions, f_1 and f_2 to maximize. Figure 20.3 shows two Pareto sets: the circles and the squares. The circle set has a wider *range* than the square set, because one of its members has a f_1 value higher than any of the members in the square set; the same applies to f_2. Members in the circle set also more *evenly distributed* than members of the square set. Therefore, the circle set is preferred to the square set, according to the metrics that we adopt. It provides the human manager a set of solutions to suit different preferences of f_1 and f_2. In other words, the human manager has better chance of finding a satisfactory solution in the circle set than in the square set.

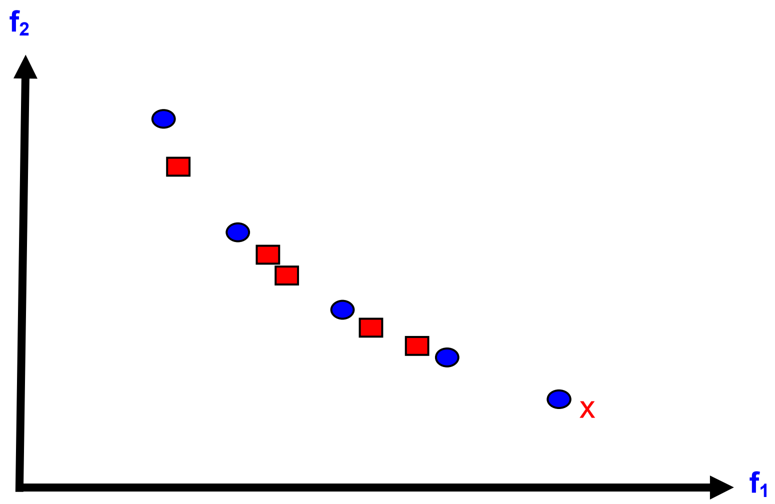

Figure 20.3 The function space showing the quality of Pareto sets: f1 and f2 are two functions to maximize. The set of circle solutions is preferred to the set of square solutions because (a) they range wider and (b) they are more evenly distributed. The mark "x" indicates a desirable member to be added to the set of red solutions.

Having decided on the metric to adopt, we have some guidance on how the management agent should set the weights for the buyer and seller agents. For example, good solution to add to the set of square solutions in Figure 20.3 is where the cross indicates. This solution will extend the range of the square Pareto set.

For the management agent, knowing the target position (such as the position marked by the cross in Figure 20.3) in its objective space is only the first step towards setting the weights for the buyer and seller agents. The only control that the management agent has is in setting the weights. It has no direct control over what schedule the buyers and sellers will generate. In attempt to generate a schedule in the target position, the management agent must discover the *mapping* from the weights space to the management agent's objective space. Given that the workforce scheduling problem does not change radically over time, learning of this mapping is possible. Many machine learning tools can be used for this purpose [14]. This is the subject of our on-going research [10].

19.6 Self-interested multi-agent scheduling

So far we have not explained how the buyer and seller agents interact to generate a schedule. The buyer and seller agents are given their objective functions by the management – the function against which their performance will be assessed. Each agent is free to adopt any methods to generate a schedule to suit their own preference, which (as recognized by staff empowerment) may take into consideration their own agenda which is beyond the management's control.

The problem is basically a self-interested distributed scheduling problem. One of the better methods for tackling this class of problems is the *Contract Net*, which was developed in the early 1980s [3][6][21]. The basic principle behind Contract Net is that the buyers broadcast their jobs, and the sellers bid for services provision. The bids vary in quality, from the buyers' point of view. For example, one bid might involve a longer traveling distance, but a better fit of skill. Since the buyer agent has been given the weights to balance between traveling distance and fit of skill (as well as other objectives), it will be able to evaluate different bids. A buyer would make an offer to the best bid. Seller agent has the final say whether to take up the offer. As the seller may get multiple offers for the same engineer, it does not necessarily accept an offer.

The contract net provides a protocol for generating schedules. Unfortunately, it does not necessarily generate the most efficient schedule. Basically, it samples one schedule only, by assigning one job to the locally best engineer ("best" as agreed between a buyer and a seller) at a time. Figure 20.4 shows a scenario involving three regions. In this scenario, Region 1 has one spare engineer. Region 3 has one job that is not served by any engineer. If the company's overall objective is to finish as many jobs as possible, while minimizing the traveling distance is of secondary importance, then it would have preferred to assign Engineer 1 to Job 1, Engineer 2 to Job 2, and Engineer 3 to Job 3. This revised schedule would require more traveling, but complete one more job. Unfortunately, this improved schedule will probably not be found by a standard contract net if each region controller were to give priority to its own engineers.

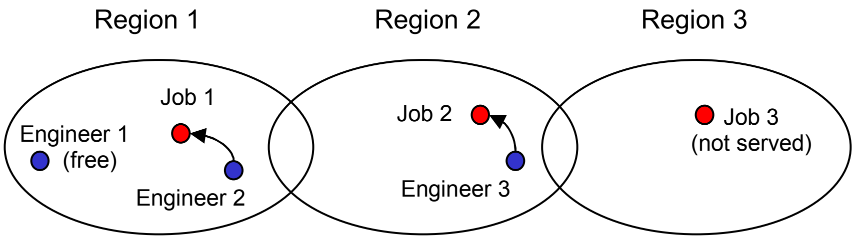

Figure 20.4 A schedule that could be improved if job-completion is the dominating criteria for solution quality

19.7 RECONNET – local search over schedules

An obvious improvement to the contract net protocol is to hill-climb in the space of schedules. A large number of local search methods in the literature could help hill-climbing in the space of schedules (e.g. see [11]). The complication in this problem is that local search requires releasing of contracts. Contracts cannot be released unilaterally; otherwise the situation would be chaotic: an engineer could be sent to a job which has been cancelled by the service buyer, and a job could be waiting to be served by an engineer whose controller has cancelled the contract. This is a self-interested distributed system. Both buyers and sellers must agree before a contract could be released. To facilitate local search, a contract release mechanism must be designed. The new protocol is called RECONNET (which stands for REtractrable CONtract NET).

RECONET introduces a contract release mechanism to standard Contract Net. When a buyer has a job that needs to be served, it could ask for bids that may involve the release of an existing contract. For example, in Figure 20.4, the buyer in Region 3 (call it Buyer 3) may ask the seller in Region 2 (call it Seller 2) to make a bid for Job 3. Seller 2 may make a bid to Buyer 3, on condition that the buyer in Region 2 (call it Buyer 2) is willing to release the contract (of buying Engineer 3's service). Buyer 3 may then offer Buyer 2 a compensation for releasing its contract. Buyer 2 may attempt to secure an alternative contract for Job 2 before it decides to take up Buyer 3's offer. The contract release protocol is shown in Figure 20.5.

Contract release is driven by a market mechanism. To do Job 3, Engineer must travel further than serving Job 2. Therefore, Seller 2 will ask Buyer 3 for a price. Suppose Seller 2 asks for a price of £20, but Job 3 is worth £100 to Buyer 3, then Buyer 3 would be able to offer Buyer 2 anything up to £80 for releasing the original contract. Suppose Buyer 3 offers Buyer 2 £60 to release the contract. Buyer 2 could then use part of it to buy alternative services for Job 2. Since the benefit to a buyer decreases as the chain lengthens, infinite loop is not a threat to this contract release protocol.

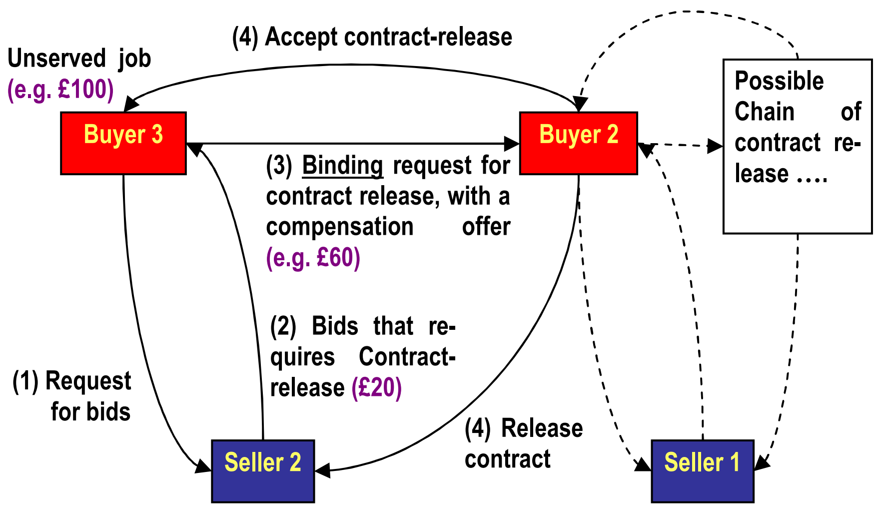

Figure 20.5 A scenario involving Contract Release

19.8 Dynamic scheduling

Workforce scheduling is a dynamic problem. New jobs may arrive at any time. Delays may occur, possibly due to complications in services or traffic congestions. This means schedules have to be revised constantly, which adds a new dimension to this complex problem.

To react to new jobs and unexpected delays, rescheduling time is crucial. Like most local search algorithms, the more time the algorithm is given, the more chance it has in finding better solutions. The size of the Pareto set generated by the management agent is also relevant.

The size of the problem varies over regions and time. A problem of realistic size would involve 150 to 300 jobs per region. In a prototype that implements RECONNET, a schedule for 7 regions of realistic size took 5 to 15 minutes to generate. (We assumed that each service buyer will buy from 7 sellers – bids from far away sellers are unlikely to be practical, due to traveling cost and traveling time.) There is plenty of room for speed-up in an operational system. It is reasonable to assume a speed-up of at least an order of magnitude over the prototype system. If the management agent needs to generate a Pareto set of 10 schedules, it will take 5 to 15 minutes to reschedule, which is quite acceptable.

When the situation is changed, rescheduling from scratch may not be the best policy. Local search is well suited for schedule-repair. RECONNET is a framework which could support a wide range of local search methods, including Tabu Search [8][9] and Guided Local Search [13][28].

19.9 Research Frontier

In RECONNET, a buyer offers compensation to another buyer for contract release. There is no reason why this compensation cannot be negotiable. Bargaining is a well studies topic in game theory. Automated bargaining could play a part in this research [12].

Each seller agent has to solve a scheduling problem for every objective function defined by the management agent. This is not a standard constraint satisfaction problem [23], hence demands specialized scheduling techniques. For example, assignments have to be agreed by buyer agents. Besides, assignments cannot be undone unilaterally. Exactly how this scheduling problem should be solved is subject of our on-going study.

The contract release protocol in RECONNET may not be optimal. Research in distributed artificial intelligence may be consulted. In [1][19][20], agents are allowed to cancel contracts unilaterally. This is viable when all agents are cooperative. Unfortunately, this is not an assumption in our model. We acknowledge the possibility that agents may manipulate the biddings to maximize their own benefit. This, plus the dynamic nature, makes contract de-committing strategies such as those proposed in [17][19][20] non-applicable for the problem defined in this paper. The practical needs in this problem demands new techniques.

We pointed out earlier that the nature of the problem does not change radically over time. Therefore, it is possible to take advantage of information gathered over time in designing the system. A simple simulator has been implemented to investigate the various degrees of delays and their impacts. Deeper investigation is being planned in the current project.

19.10 Concluding Summary

To summarize, we have used BT's problem to illustrate how workforce scheduling can be tackled with staff empowerment. The problem is formalized as a multi-objective, dynamic distributed optimization problem. We have divided this complex problem into sub-problems. This allows us to deal with them separately. It also allows us to bring in established research, such as multi-objective optimization, constraint handling, distributed scheduling, contract net, machine learning and agent-based modeling.

The key to the success of a multi-agent system is in how the authority and negotiation protocol is defined. In our application, the company's interest is looked after by giving the management agent the authority to define the criteria for assessing the schedules. The controllers are given freedom in how they schedule their engineers to serve the jobs – they know how their performance will be assessed. The system relies on the market's "invisible hand" to produce schedules that balances the different agents' needs.

The multi-agent platform implemented enables the management to test different designs before they are implemented. It allows one to evaluate the effectiveness of different market mechanisms and ask what-if questions. This allows the management to identify mechanisms and conditions under which desirable results (as defined by the management) could be achieved.

Bibliography

[1] Aknine, S., Pinson, S. & Shakun, M. F. (2004). An extended multi-agent negotiation protocol. Autonomous Agents and Multi-Agent System, 8(1):5–45

[2] Borrett, J.E. & Tsang, E.P.K. (2001). A context for constraint satisfaction problem formulation selection, Constraints, Vol.6, No.4, 2001, 299-327

[3] Davis, R., and Smith, R.G. (1988). "Negotiation as a metaphor for distributed problem solving". In A. Bond, and L. Gasser, L. (eds.), Readings in distributed artificial intelligence, Morgan Kaufmann, 333-356.

[4] Deb, K., Multi-objective optimization using evolutionary algorithms, John Wiley & Sons, 2001

[5] Faltings, B. and Macho-Gonzalez, S. (2003). “Open constraint optimisation”. In F. Rossi, (eds.), Proceedings, Principles and Practice of Constraint Programming (CP 2003), Lecture Notes in Computer Science 2833, Springer, 303-317.
[6] FIPA (Foundation for Intelligent Physical Agents). (2002). FIPA Contract Net Interaction Protocol Specification. http://www.fipa.org/specs/fipa00029/SC00029H.html
[7] Freuder, E.C. (1999). Modeling: the final frontier, The First International Conference on The Practical Application of Constraint Technologies and Logic Programming (PACLP), London, April 1999, 15-21
[8] Glover, F.W. and Laguna, M. (1997). Tabu Search. Kluwer Academic Publishers
[9] Glover, F. & Kochenberger, G.A. (ed.) (2003). Handbook of metaheuristics, Kluwer
[10] Gosling, T., Jin. & Tsang, E.P.K. (2006). Games, supply chains and automatic strategy discovery using evolutionary computation, in J-P. Rennard (Eds.), Handbook of research on nature inspired computing for economics and management, Chapter XXXVIII, 572-588
[11] Hoos, H. & Tsang, E.P.K. (2006). Local search for constraint satisfaction, Chapter 5, F. Rossi, P. van Beek & T. Walsh (ed.), Handbook of Constraint Programming, Elsevier, 245-277
[12] Jin, N. & Tsang, E.P.K. (2006). Co-adaptive Strategies for Sequential Bargaining Problems with Discount Factors and Outside Options, Proceedings, Congress on Evolutionary Computation (CEC), 7913-7920
[13] Mills, P. (2002). Extended Guided Local Search. PhD Thesis. Department of Computer Science, University of Essex.
[14] Mitchell, T.M. (1997). Machine learning, McGraw-Hill International Editions.
[15] Okabe, T. (2004). Evolutionary multi-objective optimization. Shaker Verlag Aachen.
[16] Prosser, P. (1990). Distributed asynchronous scheduling. PhD Thesis. Department of Computer Science, University of Strathclyde.
[17] Raiffa, H. (1982). The Art and Science of Negotiation. Harvard University Press, Combridge, Mass.
[18] F. Rossi, P. van Beek & T. Walsh (ed.). (2006). Handbook of Constraint Programming, Elsevier
[19] Sandholm, T. and Lesser, V. (2002). Leveled-commitment contracting: A backtracking instrument for multiagent systems. AI Magazine, 23(3):89–100.
[20] Sen, S. and Durfee, E. (1998). A formal study of distributed meeting scheduling. Group Decision and Negotiation, 7:265–289.
[21] Smith, R.G. (1980). “The contract net protocol: high-level communication and control in a distributed problem solver”. IEEE Transactions on Computers, Vol.C-29, No.12, 1104-1113.

[22] Thomas, K. W. and Velthouse, B. A. (1990) Cognitive Elements of Empowerment: An 'Interpretive' Model of Intrinsic Task Motivation. Academy of Management Review, Vol 15, No. 4, 666-681

[23] Tsang, E.P.K. (1993). Foundations of Constraint Satisfaction. Academic Press

[24] Tsang, E.P.K. (2002). Constraint satisfaction in business process modelling. Technical Report CSM-359, University of Essex.

[25] Tsang, E.P.K., Gosling,T., Virginas, B., Voudouris, C. & Owusu, G. (2005). Retractable Contract Network for Distributed Scheduling, in Proceedings, 2nd Multidisciplinary International Conference on Scheduling: Theory & Applications (MISTA), New York, July 2005, 485-500

[26] Ursu, M., Virginas, B. and Voudouris, C. (2004), "Distributed resource allocation via local choices: general model and a basic solution". Proceedings, 8th International Conference on Knowledge-Based Intelligent Information & Engineering Systems (KES2004). Wellington, New Zeeland.

[27] Virginas, B., Ursu, M., Owusu, G. and Voudouris, C. (2005). "Intelligent workforce allocation within an agent-based paradigm: central and distributed decision powers". The IASTED International Conference on Artificial Intelligence and Applications (AIA 2005). The 23rd IASTED International Multi-Conference on Applied Informatics, Austria.

[28] Voudouris, C. and Tsang, E.P.K. (2003). "Guided local search". In Glover, F. (eds.) Handbook of metaheuristics. Chapter 7, Kluwer, 185-218.

[29] Wilkinson, A. (1998). Empowerment: theory and practice. Personnel Review. Vol. 27(1): 40-56

[30] Shen, W. and Norrie, D.H., Agent-Based Systems for Intelligent Manufacturing: A State of the Art Survey, in International Journal of Knoweldge and Information Systems, 1:2, 1999, 129-156

[31] Ehrgott, M. and Gandebleux, X. (ed.), Multiple Criteria Optimization: State of the Art Annotated Bibliographic Survey, International Series in Operations Research and Management Science, Vol. 52, Kluwer Academic Publishers, Boston, 2002

[32] Ursu, M.F., Virginas, B., Owusu, G., and Voudouris, C, Distributed Resource Allocation via Local Choices: A Case Study of Workforce Allocation, International Journal of Knowledge-Based and Intelligent Engineering Systems, IOS Press, Netherlands, Volume 9, Number 4, 2005, 293-301