Adaptive Interfaces and Soft Computing

Esko Juuso and Kauko Leiviskä
Department of Process Engineering
University of Oulu
P.O. Box 444 , FIN-90571 Oulu
FINLAND
e-mail: (Kauko. Leiviska or Esko. Juuso) @oulu.fi

Abstract: Adaptive interfaces are tools that combine traditional simulation and intelligent methods thereby enhancing the capabilities of simulation systems. An integrated set of compatible intelligent tools is a good goal for these environments, and the use of qualitative knowledge increases in simulation applications.
Adaptive interfaces present the results of simulation also to the users who are not so closely involved in simulation practices and methods. Hypermedia shows a good promise in both user interfaces and documentation

Esko Juuso was born in Ylitornio, Finland, 1951. He received the Diploma Engineering degree in Technical Physics from the University of Oulu in 1979.
He has been Research Engineer and Process Computer Analyst at Outokumpu during 1980 - 1985. He has been Assistant and Head Assistant at the Control Engineering Laboratory, University of Oulu since 1986. During 1990 - 1992 he has been visiting researcher at UMIST, Manchester, England. He is a member of Esprit Working Groups SiE - Simulation in Europe since 1993, and FALCON - Fuzzy Algorithms for Control since 1994. He is member of Board of the Scandinavian Simulation Society (SIMS) since 1995 and a member of various conference committees. A list of more than 80 publications of which he is (co)author is available. Recently his work has concentrated on modelling and control of industrial processes, intelligent control methods, fault diagnosis, production scheduling, and managerial decision making in uncertain environment.

Kauko Leiviskä was born in Pyhäntä, Finland, 1950. He received the Diploma Engineering degree in Process Engineering in 1975, the Licentiate of Technology degree in Control Engineering in 1976 and the Doctor of Technology degree in Control Engineering in 1982 from the University of Oulu. He has been Professor of Control Engineering and Head of Control Engineering Laboratory at the same University since 1988. He has been the member of IFAC-Systems Engineering Committee 1988-1992 and the member of several conference committees. He has also been the member of board in Finnish NMO for IFAC in 1988-1991. A list of about 200 publications of which he is (co)author is available. Recently his work has concentrated on modelling and control of industrial processes, intelligent control methods, production scheduling and millwide control. He has also been consulting industry on control engineering and millwide control applications.

1. Introduction

Modern simulation systems are combinations of numeric and symbolic computation and hybrid systems as shown in Figure 1 are repeatedly met [13]. In such a system the exchange of data between different parts is difficult, although necessary. Coupling simulation software with software components of different nature, however, is unavoidable in real applications, and therefore the exchange of data should be ensured.

Adaptive interfaces are one solution for this problem, because they are combining adaptive and intelligent methods with traditional computer simulation. Artificial Intelligence (AI) has concentrated on building reasoning and problem -solving mechanisms which correspond to human way of doing those things. Simulation is an attempt to study and understand the operation of real physical systems by running computer models of these systems. Artificial intelligence is working at symbolic level, and simulation at numeric level. Recently, progress in soft computing (or computational intelligence) has provided many methodologies for solving these problems. Actually, soft computing can be considered as a collection of methods for qualitative simulation (in broad sense).

Several soft computing methodologies are available. Rule-based expert systems are suitable for codifying metaknowledge of a system but they are not easy to extend to complex systems. For applications, mapping quality to quantity can be done by fuzzy set systems. Behavioural models can be based on neural networks, if the system is small enough. Genetic algorithms and related evolutionary strategies have recently been used in many narrow applications.

Hypermedia lends itself easily to user interfaces of complex systems. It is also an excellent way of linking on-line documentation to automation systems. These applications are described in [23]. In the same way, it could also facilitate the interpretation of simulation

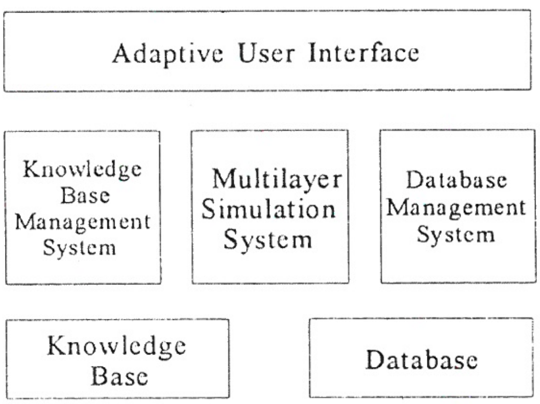

Figure 1. The Combined Simulation and Expert System

systems. Hypermedia systems require an information model to be efficient.

2. What Is Soft Computing

2.1 Rule-based Expert Systems

In rule-based expert systems, the domain knowledge is represented as sets of rules that are checked against a collection of facts or knowledge about the current situation [31]. They rely on expert's knowledge and inference mechanisms that in a way simulate human reasoning. Rule-based programming is commonly used in the development of expert systems. However, this paradigm leads to serious problems in practical applications. The first problem is inheritant to its knowledge acquisition principle; when really massive rule bases are met, their updating and adaptation becomes impossible without linking the rule-based systems to more efficient modelling methods. Using declarative Prolog language (or extensions of it) at the relation level can reduce the amount of rules to some extent [11]. However, these simulations are still too slow for complex systems.

The other problem is in the uncertainty processing which is needed for real world problems. Uncertainty is an unavoidable part of complex simulation environments: information is partly uncertain or subjective; scope of the applicability of models is usually limited; many interactions with other application areas should be taken into account; different time horizons are used in different application areas, etc. Therefore, methods for uncertainty processing could be very beneficial in many applications, and especially in connecting applications (Figure 2 [12]).

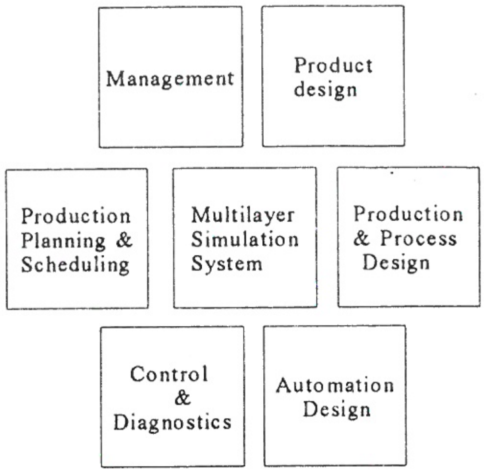

Figure 2. Simulation Applications in Manufacturing Industry [12].

2.2 Fuzzy Set Systems

Introduction of fuzzy systems could benefit expert systems. Fuzzy sets provide a unified framework for taking into account the gradual or flexible nature of variables, and the representation of incomplete information [6]. Fuzzy control has been one of the most active and fruitful areas of application of fuzzy sets theory. A good introduction to rule-based fuzzy systems can be found in a survey paper of Lee's [22].

A wide range of fuzzy development and simulation environments is available. Most of the systems are based on fuzzy rules, Mamdani- [5] or Sugeno-Tagaki type [28]. Typically, advanced shells have graphical user interface running under MS Windows or MS DOS. For some of them, source code generators to programming languages, usually to C code, are available. Also some microcontrollers, usually belonging to the bigger product family, are supported. Commercial development environments are well suited to small problems but building complex systems is quite laborious. On-line tuning is not possible, and adaptability is not taken into account in final running versions This was the main reason for developing a new fuzzy controller FuzzyCon which is a Windows application implemented in Visual Basic 3.0. FuzzyCon is a fuzzy controller specially designed for adaptive tuning in process industry [17].

In the modelling framework, the fuzzy sets approach provides a flexible way to real world applications through defining the meaning for linguistic variables. Rules are generated in the same way as for conventional knowledge-based systems. The only difference is that usually the number of rules is kept considerably smaller because of the fuzzy representation.

As mentioned before, acquiring the knowledge from the human operator is a tedious and time-consuming task for complex systems at least if the rule-based procedure is used. By using fuzzy approach the available prior knowledge can be combined with information obtained from numerical measurements. Even if no prior knowledge exists, the rules and membership functions can be extracted from the data [1]. Fuzzy c-means (FCM) clustering is an example of data clustering techniques which are used for lumping together data points that populate some multidimensional space [3]. Various methods have been developed for fuzzy clustering [12, 19, 32, 33].

2.3 Neural Networks

Neural networks have been inspired by biological nervous systems and mathematical theories for learning [8]. They are characterized by their learning ability and parallel distributed structure. Neural networks can be considered as black box modelling methods. The way NNs perform their internal function is hard to visualise, and therefore NN models are difficult to connect to other models of the system. Another drawback is that sometimes extremely long training periods are required.

Various methods have been developed for neural computing. Backpropagation is probably the most popular method in the supervised learning [26]. Self-organizing maps are well-known networks in the unsupervised learning [20]. Self-organizing maps learn to categorise input vectors by moving the neurons to respond to density differences in the input space. Neural computing can be implemented on the basis of matrix manipulations, and therefore the execution can be vectorised and parallellised. Learning algorithms have a close relation to optimisation methods, e.g. a backpropagation method corresponds to gradient descent. A huge number of programming and development tools is available both commercially and in public domain.

Difficulties in explanation and generalisation suggest that connecting NNs to other modelling techniques is vitally important as far as complex systems are concerned. A solution might be a neurofuzzy approach which makes the model more understandable. ANFIS method (Adaptive-Network-based Fuzzy Inference Systems) is a well -known neurofuzzy method which is suitable for the tuning of membership functions [10]. Neural computing also provides a suitable identification method for working point adaptation. Another possibility would be to use knowledge-based networks that carry with them the original expert system structure [21]. Nodes in the network correspond to premises or conclusions and connections represent rules. So it can be said that the knowledge base of the expert systems and the network structure are transferable.

Clustering type techniques as self-organizing maps can be used for extracting features from data, and then these features are generalised by fuzzy methods, by neurofuzzy methods or by linguistic equations. Various neural learning methods [4, 7, 9], orthogonal least squares [30] or inductive learning [25] can be used in connection with data.

2.4 Genetic Algorithms

Genetic algorithms (GA) have for long time been considered as belonging to general systems theoreticians interested in extremely complex optimisation studies [27]. Recently, some narrow applications have been developed. GAs can be considered as an experimenting tool which produces a satisfactory solution which is not necessarily optimal. GAs are useful when this can be accepted, e.g. model is not known, search space is very large, and data are noisy. The algorithm (selection, crossover, mutation) is a reasonable way for processing a population of alternatives. The crucial thing is that fitness functions measure the right thing, i.e. they should be application dependent.

Genetic algorithms can assist other methods of soft computing by optimising structures. For expert systems, optimising the rule base is a possible application. However, genetic algorithms are suitable for very complex systems which on the other hand, are fairly problematic for expert system. Network structures of neural networks could be another application area, but the execution will always be very slow since both GAs and NNs require a lot of computation. Development of linguistic equations from data is also a possible application.

2.5 Linguistic Equations

Different approaches and modelling methods can be combined through the Linguistic Equations Framework presented in [15]. The framework provides a unified method for developing and tuning adaptive expert systems. A wide variety of applications is based on this methodology, although, some of these applications are implemented by conventional techniques. The development system is implemented in Matlab environment, and the resulting fuzzy systems can be transferred to FuzzyCon for further on-line tuning.

In this methodology, membership functions are tuned by simulation experiments with multilayer simulation systems [14]. The simulation system can also be (partly) replaced by experts or by experiments with real systems. Both analytical and heuristic knowledge can be used simultaneously. As many rules as possible are replaced by linguistic relations. However, some of them are needed, and the system provides a flexible environment for combining these rules with more efficient modelling methods. In the linguistic equations approach, the relations are developed gradually: only a small part of the problem is taken into account at a time. The system is adaptive since the meaning of the linguistic values depends on the working point of the process. More details about the tuning of fuzzy controllers can be found in [18].

3. Simulation and Soft Computing

A simulation process may be divided e.g. into the following general phases [29]: definition of simulation scope, model construction, solution of model equations, model validation and model use.

Simulation requires many types of expertise, and getting to the appropriate level takes time. First of all, the application area must be known. Users are increasingly constructing their models from modules available in the unit libraries without any help from simulation specialists. Intelligent methods provided by soft computing are useful in simulation - from modelling to analysis of the results. Especially non-expert users need these adaptive intelligent interfaces [13].

3.1 Definition of Simulation Scope

The definition of the simulation scope restricts the simulation according to the level of description, processes or systems included, target accuracy and model types or solution methods utilized.

This phase is very case dependent and intelligent methods could be of help in many ways. An assisting system can propose which area should be selected for a study, which kinds of models are required, and which results should be gathered, etc. If the system has a learning ability, it can warn users if a simulation run designed will either probably not work or produce any additional information.

Intelligent monitoring systems are checking if the requirements of the simulation run are met. For stochastic models, the system checks if all the necessary data are gathered. For continuous time simulation, the system checks if the proper integration method is used. Expert systems have been developed for selecting suitable step sizes and integration algorithms. These systems change, if necessary, the method during the run. These systems are aimed at handling considerable changes in working point conditions. They do not compete with multiple step size integration methods designed for dynamic step size control.

Fuzzy set systems can improve operation of expert systems at higher level since fuzzy partitioning can be used as a basis for analyzing working areas during the simulation experiments. Fuzzy set systems also provide more flexible tools for designing simulation experiments since the interactions between variables and/or parameters can be utilised as well. Neural networks can be used as automatic learning and classification tools during the simulation experiments. Genetic algorithms are useful provided that the search space is very large, and structural variations are essential. Linguistic equations separate the knowledge bases and databases. They also produce very compact simulation experiment designs since the prior knowledge available can be taken into account as interactions. Explanations can also be generated for complex systems since the corresponding rules or relations can be constructed for a limited part of the system. Usually, equations are crisp, and all the fuzziness is taken into account by membership functions.

3.2 Model Construction

The model construction starts from the division of the system to be modelled into two parts: the model and the environment.

In the simulation of technical systems a hierarchical structure of submodels and system components is usually met. The division makes it possible to build parts of the final model as separate tasks with a final task of integrating the submodels into one. The submodels are built of variables and relations between the variables.

Expert systems can help in model construction in cases which do not include a considerable amount of uncertainties. As mentioned above, simulation models are usually constructed from components provided by the simulation package. Construction can be difficult for non-expert users but a conventional expert system can help users to select the most suitable modules and to specify the parameters correctly.

Expert systems help also non-expert users in parameter estimation, and in testing and fitting of input data. Some expert systems have been developed for assisting in the use of large software packages, e.g. in the selection of suitable methods in statistical analysis packages.

In the case of uncertainties, building sophisticated systems on the basis of conventional expert systems is not feasible, and therefore, a more flexible method for uncertainty processing is needed. Soft computing provides many additional methods in this area. Fuzzy set systems provide a flexible and well - explained link to the real system, e.g. when uncertainties in the input data must be taken into account (see Figure 3). Neural networks can be used as an automatic learning tool if their restrictions are kept in mind. Neurofuzzy systems combine fuzziness with the black box neural methods. Linguistic equations systems have a very compact structure which makes the adaptation to different working points easier.

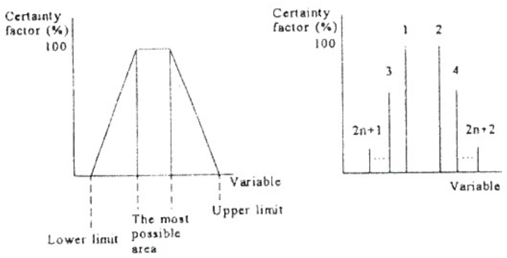

Figure 3. Fuzziness in Simulation Experiment Design [14]

3.3 Solution of Model Equations

The model built will ultimately turn up as a set of mathematical equations that are solved by the computer. This process is to a great extent facilitated by the use of different programming languages and simulation environments. In the selection of a specific programming language it is again important to consider the objective of the simulation study, because different simulation tools have different features.

Result analysis, optimization and selection of the key changes are important areas where a lot can be obtained by adaptive and intelligent methods. The goal of these systems is to give advice to users on the basis of simulation results. Advice should include instruction, how to change the model and its parameters in such a way that the goal is reached as well as possible. Since fuzzy systems have a gradual connection to the quantitative world, selection of the best model has a more concrete meaning. Sensitivity analysis can be produced by fuzzy set approach in a very straightforward manner (see Figure 3). Generating the change recommendations can be done by fuzzy reasoning. Membership functions can be considered as a tool for interpreting the user's questions and goals to the simulation program. Deep knowledge about both the structure and operation of the model and the simulation techniques is needed. Fuzzy set systems can also be a basis for analyzing working areas during the simulation experiments. Linguistic equations can bring experimental design, monitoring and control tasks together since the compactness of the system makes all comparisons very fast. A limited set of tunable parameters also help in adapting to new working points.

3.4 Model Validation

The validation of a simulation is the most important phase of any simulation study. In that phase one gets assured that the model is a true representation of the system in consideration. In order to make sure of the reliable use of the model, all the variables must have a certain area of confidence. Extrapolation outside the confidence region is always dangerous. Usually the model is based on short-term experiments and it describes well this situation, but later perhaps after the process changes, it cannot be relied on at all.

An intelligent system can assist in searching errors and running automatical tests. Intelligent systems can also analyse erroneous results and search potential sources for those errors. In addition , a higher level qualitative model to be used in the validation process, can be constructed. Fuzzy set systems are natural extensions in this area since a comparison to qualitative models makes model validation, critical testing and error analysis more consistent. Linguistic equations approach can be used in expanding the validation to large complex systems which cannot be handled by other methods.

3.5 Model Use

Once the model has been built and validated, it is used as a counterfeit real system. This means that one could experiment with a system that has not been built yet. One could also go through sequences of events which are not possible with a real system e.g. for safety or cost reasons.

In the use of simulation results one should be aware of the inherent limitations of their applicability. Very often the simulation results will not be used as such, but they will be used as input into a decision process. In such cases it is even more important to convey the qualitative implications of the simulation in an easily understandable form to the decision maker. This stresses on the interface of the model and interactive interfaces for information input , processing and reporting are favoured. The model documentation is also important and the self-documenting features of computerised models are recommended.

Simulation runs produce a vast amount of results, from which the essential part should be picked up. Expert systems can filter the results to the users by explaining the results and how they have been achieved. They can also produce new knowledge, e.g. rules, structures, classification, sensitivity and parameter analysis, from simulation data. Expert systems enhance the capabilities of simulation as a decision support tool, e.g. estimate how well the goals are achieved, taking into account possible contradictory constraints, monitoring the run to find causal relationships, analysing causal relations found in order to construct the best causal model, and show how sensitive the resulting parameters are to the manipulated parameters. They can also propose changes to the model on account of the sensitivity analysis, automatically re-run the simulation after changes and interpret user's questions and goals to the program.

Fuzzy set systems are very powerful tools in explaining the results to the users. Neural networks can be used as automatic learning and classification tools during the simulation experiments. A vast amount of data provides a good platform for the application of neural networks. Linguistic equations approach is a new efficient method for producing new knowledge.

4. Hypermedia in Simulation

Hypermedia provides a unified training environment by integrating various kinds of materials available on different media [23]. In addition to the hypertext itself, these systems can include computer simulation, animation, spreadsheet applications, CAD programs, etc. Hypermedia is also an excellent way to link on-line documentation to automation systems. These applications are described in [23]. Objectives of the process and how they are achieved are the key idea in the operator support system.

The efficient use of hypermedia information systems requires information modelling. Multilevel flow modelling (MFD) [24] is useful for a continuous process, which includes process devices, process functions and process goals. Currently the representation is extended to cover process control, alarms, and interlocking situations. Also the division of workload between process computers and human operators is sketched in the current version of the modelling paradigm. An example of how the MFM-model is used in conveying the model structure to display hierarchies and hypertext documentation is shown in Figure 4 [23]. The experiences are based on the power plant simulator that was connected to an on-line automation system to highlight the problems in connecting knowledge from process and automation design.

MFM-modelling could also provide a working environment for large (and heterogeneous) simulation systems. This way of modelling can explain the position of individual simulation models to non-professionals, which will enhance understanding of simulation and simulation results. The hierarchical models also provide means to integrate simulation tools with other planning (or design) tools. The resulting simulation environments could be attractive and easy to understand.

Hypermedia can link the multiparadigm-multilanguage models together. A lot of development work must be done in order to make the system practically applicable,

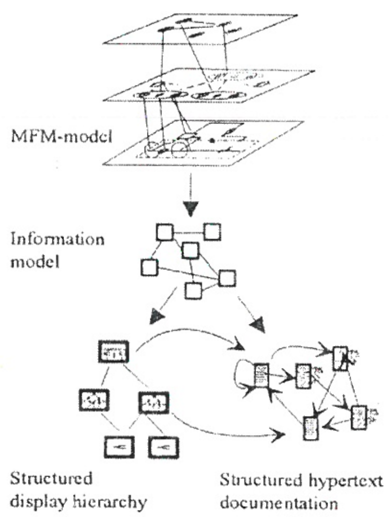

Figure 4. Using An Information Model As A Basis for User Interface Design [23]

although most of the ingredients of the system have been tested in small scale applications. The overall system could contain the following parts [12]: (1) the simulator and the intelligent interface of the simulator can be considered as a multilayer simulator; (2) a hypermedia user guide gives advice on how to use an intelligent interface in such a way that all the user defined variables and parameters are consistent, i.e. the results of the simulation will be reasonable; (3) a hypermedia training material teaches non-specialist users different ways of running the process.

5. Conclusions

Adaptive and intelligent methods can enhance properties of simulation interfaces and usability of the simulators. Different methodologies of soft computing can find their own role in the overall system, and all the methodologies, including expert systems, should be adapted to appropriate levels in simulation applications. Expertise on different domains could be acquired into wider use through soft computing.

REFERENCES

1. BABUSKA, R. and JUUSO, E.K., **Constructing Fuzzy Models From Prior Knowledge and Data**, Open Working Group Meeting of FALCON (Fuzzy Algorithms for Control), Aachen, September 1, 1995, 26 p.

2. BABUSKA, R. and VERBRUGGEN, H.B., **A New Identification Method for Linguistic Fuzzy Models**, Proceedings FUZZ-IEEE/IFES'95, Yokohama, Japan, March 1995, pp. 905-912.

3. CHIU, S., **Fuzzy Model Identification Based On Clustering Estimation**, JOURNAL OF INTELLIGENT & FUZZY SYSTEMS, 2(3), September 1994.

4. VALENTE DE OLIVEIRA, J., **Neuron Inspired Rules for Fuzzy Relational Structures**, FUZZY SETS AND SYSTEMS, 57(1), July 1993, pp. 41-55.

5. DRIANKOV, D., HELLENDOORN, H. and REINFRANK, M., **An Introduction To Fuzzy Control**, SPRINGER, Berlin, 1993.

6. DUBOIS, D. and PRADE, H., **Fuzzy Sets: A Survey of Engineering Applications**, European Symposium on Computer Aided Process Engineering - 2, ESCAPE - 2, Toulouse, France, October 5 - 7, 1992, COMPUTERS & CHEMICAL ENGINEERING, Vol. 17 S, Pergamon, Oxford, UK, 1993 pp.373-380.

7. GLORENNEC, P. Y., **Learning Algorithms for Neurofuzzy Networks**, FUZZY CONTROL SYSTEMS, CPC Press, Boca-Raton, FL, 1994, pp. 3-18.

8. HUANG, S.H. and ZHANG H.-C., **Neural-expert Hybrid Approach for Intelligent Manufacturing: A Survey**, COMPUTERS IN INDUSTRY, 26, 1995, pp.107-126.

9. JANG, J.-S.R., **Self-learning Fuzzy Controller Based On Temporal Back-propagation**, IEEE TRANSACTIONS ON NEURAL NETWORKS, 3(5), September 1992, pp.714-723.

10. JANG, J.-S.R., **ANFIS: Adaptive-Network-based Fuzzy Inference Systems**, IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS, 23(3), 1993, pp.665-685.

11. JUUSO, E.K., **Linguistic Equation Framework for Adaptive Expert Systems**, Proceedings of the 1992 European Simulation Multiconference, York, UK, June 1-3, 1992, pp. 99-103, San Diego, USA, 1992, SCS International.

12. JUUSO, E.K., **Adaptive Interfaces in Synthetic Environments**, 2nd Workshop of the SiE Working Group (SiE-WG) on Improving the Modelling and Simulation Process, Brussels, June 29-30, 1995, 8 p.

13. JUUSO, E.K., **Adaptive Interfaces Based On Soft Computing**, 2nd Workshop of the SiE Working Group (SiE-WG) on Improving the Modelling and Simulation Process, Brussels, June 29-30, 1995, 10 p.

14. JUUSO, E.K. and LEIVISKÄ, K., **An Intelligent User Interface for Combined Simulation and Expert System**, Proceedings of the 1990 European Simulation Multiconference, Nuremberg, June 10-13, 1990, pp. 300-305.

15. JUUSO, E.K. and LEIVISKÄ, K., **Expert Systems Combined With A Multilayer Simulation System**, in MATHEMATICAL and INTELLIGENT Models in System Simulation, BALTZER SCIENTIFIC PUBLISHING, 1991, pp.325-330.

16. JUUSO, E.K. and LEIVISKÄ, K., **Adaptive Expert Systems for**

Metallurgical Processes, Proceedings of the IFAC Workshop, Espoo, Finland, August 26-28, 1991, IFAC Workshop Series, 1992, No. 2, PERGAMON, Oxford, UK, 1992, pp.119-124.

17. JUUSO, E.K. and LEIVISKÄ, K., **A Development Environment for Industrial Applications of Fuzzy Control**, Proceedings of 3rd European Congress on Intelligent Techniques and Soft Computing -EUFIT'95, Aachen, August 28-31, 1995, Vol. 2, pp. 796-803, Aachen, 1995, Mainz.

18. JUUSO, E.K., MYLLYNEVA, J. and LEIVISKÄ, K., **Adaptive Tuning of Fuzzy Logic Controllers**, Preprints of the 2nd IFAC Workshop on Computer Software Structures Integrating AI/KBS Systems in Process Control, Lund, Sweden, August 10-12, 1994, pp. 82-87.

19. KAYMAK, U. and BABUSKA, R., **Compatible Cluster Merging for Fuzzy Modeling,** Proceedings FUZZ-IEEE/IFES'95, Yokohama, Japan, March 1995, pp. 897-904.

20. KOHONEN, T., **Self-Organization and Associative Memory**, 2nd edition, SPRINGER, Berlin, 1987.

21. LACHER, R.C., HRUSKA, S.I. and KUNCICKY, D.C., **Backpropagation Learning in Expert Networks**, IEEE TRANS. NEURAL NETWORKS 3, 1, 1992, pp.62-72.

22. LEE, C.C., **Fuzzy Logic in Control Systems: Fuzzy Logic Controller-Part I and Part II**, IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS, 20, 1990, pp.404-435.

23. LEIVISKÄ, K., **Hypermedia in Simulation and Control**, 2nd Workshop of the SiE Working Group (SiE-WG) on Improving the Modelling and Simulation Process, Brussels, June 29-30, 1995, 6 p.

24. LIND, M., **Modelling Goals and Functions of Complex Industrial Plants**, APPLIED ARTIFICIAL INTELLIGENCE, 8, 1994, pp. 259-283.

25. ROSS, T. J., **Fuzzy Logic With Engineering Applications**, MCGRAW-HILL, New York, 1995.

26. RUMMELHART, D.E., HINTON, G.E. and WILLIAMS, R.J., **Learning Internal Representations By Error Propagation**, Parallel Data Processing, MIT PRESS, Cambridge MA, 1986, pp. 318-362.

27. SZCZERBICKA, H., **Are Genetic Algorithms A Panacea for Optimization of Simulation Models?**, Proceedings of the Conference on Modelling and Simulation, San Diego CA, 1994, pp. 38-46, SCS International.

28. TAKAGI, T. and SUGENO, M., **Fuzzy Identification of Systems and Its Application To Modeling and Control**, IEEE TRANSACTIONS OF SYSTEMS, MAN AND CYBERNETICS, 15(1), 1985, pp.116-132.

29. WAHLSTRÖM, B. and LEIVISKÄ, K., **On the Use of Simulation in the Engineering Sciences**, IMACS TRANSACTIONS ON SCIENTIFIC COMPUTATION 85, Vol. III, ELSEVIER SCIENCE PUBLISHERS B.V., Amsterdam, 1986, pp. 3-10.

30. WANG, L.-X., **Adaptive Fuzzy Systems and Control, Design and Stability Analysis**, PRENTICE HALL, New Jersey, 1994.

31. WATERMAN, D.A., **A Guide To Expert Systems**, ADDISON-WESLEY PUBLISHING COMPANY, 1986.

32. YOSHINARI, Y., PEDRYCZ, W. and HIROTA, K., **Construction of Fuzzy Models Through Clustering Techniques**, FUZZY SETS AND SYSTEMS, 54, 1993, pp. 157-165.

33. ZHAO, J., WERTZ, V. and GOREZ, R., **A Fuzzy Clustering Method for the Identification of Fuzzy Models for Dynamical Systems**, 9th IEEE International Symposium on Intelligent Control, Columbus, OH, August 1994.