Lecture Notes in Artificial Intelligence 1415

Subseries of Lecture Notes in Computer Science
Edited by J. G. Carbonell and J. Siekmann

Lecture Notes in Computer Science
Edited by G. Goos, J. Hartmanis and J. van Leeuwen

Springer
Berlin
Heidelberg
New York
Barcelona
Budapest
Hong Kong
London
Milan
Paris
Santa Clara
Singapore
Tokyo

José Mira   Angel Pasqual del Pobil
Moonis Ali   (Eds.)

Methodology and Tools in Knowledge-Based Systems

11th International Conference
on Industrial and Engineering Applications
of Artificial Intelligence and Expert Systems
IEA-98-AIE
Benicàssim, Castellón, Spain, June 1-4, 1998
Proceedings, Volume I

Springer

Series Editors
Jaime G. Carbonell, Carnegie Mellon University, Pittsburgh, PA, USA
Jörg Siekmann, University of Saarland, Saarbrücken, Germany

Volume Editors

Angel Pasqual del Pobil
Department of Informatics, Jaume-I University
Campus Penyeta Roja, E-12071 Castellón, Spain
E-mail: pobil@inf.uji.es

José Mira
Departamento de InteligenciaArtificial, Facultad de Ciencias
Universidad Nacinoal de Educación a Distancia
Sende del Rey, s/n, E-28040 Madrid, Spain
E-mail: jmira@dia.uned.es

Moonis Ali
Department of Computer Science, SouthwestTexas State University
San Marcos, TX 78666-4616, USA
E-mail: ma04@swt.edu

Cataloging-in-Publication Data applied for

Die Deutsche Bibliothek - CIP-Einheitsaufnahme

International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems <11, 1998, Benicasim; Castellón de la Plana>:
11th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems : Benicàssim, Castellón, Spain, June 1 - 4, 1998 ; proceedings / IEA-98-AIE. - Berlin ; Heidelberg ; New York ; Barcelona ; Budapest ; Hong Kong ; London ; Milan ; Paris ; Santa Clara ; Singapore ; Tokyo : Springer
Vol. 1. Methodology and tools in knowledge based systems / José Mira ... (ed.). - 1998
(Lecture notes in computer science ; Vol. 1415 : Lecture notes in artificial intelligence)
ISBN 3-540-64582-9

CR Subject Classification (1991): I.2, J.2, J.6

ISBN 3-540-64582-9 Springer-Verlag Berlin Heidelberg New York

Typesetting: Camera ready by author
SPIN 10637249 06/3142 – 5 4 3 2 1 0 Printed on acid-free paper

Preface

Nowadays, it is generally accepted that the aim of Applied Artificial Intelligence is to render computational a large portion of non-analytical human knowledge. To attain this end, we need first to build knowledge-level models of analysis and synthesis tasks in scientific and technical domains, such as those performed daily by human experts in fields such as medical diagnosis, design in civil or telecommunication engineering, architecture, flexible manufacturing, or tutoring. Then, these models have to be transformed in such a way that their entities and relations can be linked to the primitives of a programming language and, finally, produce a program and continue with the usual phases in software engineering (validation, evaluation, and maintenance).

This purpose, that seems to be clear, has suffered since its origins in 1956, from a lack of methodology and foundations. That is, there has been an excessive hurry to develop applications (*expert* systems) without the technical and methodological support available to other engineering disciplines —those dealing with matter or energy— having been established. This is the reason why the advancement of Knowledge Engineering has not been as robust as expected.

Fortunately, interest in methodology and foundations has grown in recent years, commencing by Clancey and Chandrasekaran's proposals about generic tasks aiming at capturing recurrent abstractions in human knowledge modeling. Then, efforts have been made to build libraries of problem-solving methods to develop these tasks by decomposing them up to primitive level and completing these tasks and methods with ontologies and domain knowledge models together with a set of assumptions about implicit representations for each method and about the method's assumptions which are implicit in each domain model. These three basic concepts —tasks, method, and domain—, along with the underlying pursuit of designing reusable components, have characterized most of methodological developments around KADS, CommonKADS, and PROTÉGÉ, for instance.

The scope and topics included in the Call for Papers of the *Eleventh International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems* (IEA/AIE-98) were compiled within this spirit of concern about sound foundations and methodology, as well as with the explicit acknowledgment of the necessity of developing efficient procedures to make the models operational. As a result of this call, 291 contributed and invited papers were submitted from 41 countries; the program committee selected 187 among them, after conscientiously considering the reviews provided by at least two referees per paper. We believe that the significant increase in the number of submitted papers, with respect to recent conferences, is a symptom of a maturing interest within the AI community towards fundamental issues relevant to well-founded and robust applications in the real world.

We are pleased to present, as program chairs and editors of these two volumes, a final version of the accepted papers incorporating the reviewers' comments. We have arranged their contents basically following the topic list included in the Call for Papers, adding some additional topics which received special attention as a result of being the subject of invited sessions. The first volume entitled *Methodology and Tools in Knowledge-Based Systems*, is divided into four main parts and includes the contributions having a basic and methodological nature, along with those concerning knowledge modeling, formal tools, and generic tasks of analysis in applied AI. There are sections on fuzzy knowledge representation and inference, qualitative reasoning, evolutionary computing, and multiagent systems, among others.

One of the most frequent deficiencies in the majority of methodological developments lies in ignoring the conclusive step about how to render the models operational with the final result of an implemented system. We believe that this fact accounts for a considerable lack of credibility towards AI among researches on the outside, who feel that it has failed in that it has not made enough inroads into real-world applications. Consequently, AI researchers are sometimes seen as *just blowing smoke*. It is still common to find journal articles that do not support claims on rigorous experimental evidence or that only show solutions to toy problems by way of validation.

In the second volume, with the title *Tasks and Methods in Applied Artificial Intelligence*, we have included the contributions dealing with aspects that are more directly relevant to application development. These contributions are grouped into five parts: generic tasks of synthesis and modification, machine learning, applied AI and Knowledge-Based Systems in specific domains, and validation and evaluation criteria.

The editors are also aware of the grand challenges for AI concerning artificial behavior for agents that have to deal with the real world through perception and motor actions. Nowadays, there is an enormous lack of balance between existing AI systems in some aspects of their competence. Whereas in some formal microworlds AI systems have reached the highest human level of competence —the recent success of chess-playing systems being a paradigmatic example—, or there are knowledge-based systems exhibiting human expert competence in narrow technical domains such as medical diagnosis, etc., few systems exist surpassing the competence of a cockroach, for instance, in moving around pursuing a goal in an unstructured world. This enormous distance between pure abstract intellectual tasks at one end, and those that involve sensorimotor interaction with the physical world at the other, calls for an emphasis on research on robotic agents.

Since the current state of affairs is partly due to the Turing vision of a disembodied, abstract, symbol-processing intelligence, new proposals —such as those put forward by Harnad or Brooks— are worth consideration. Robotic capacities including the ability to see, grasp, manipulate, or move have been added to an extended version of the Turing test. The symbol grounding problem has been approached by the physical grounding hypothesis: grounding a system's representations in the physical world via sensory devices with the result of emergent functionalities. Taking the biological paradigm seriously implies building on top of an integrated and distributed sensorimotor system, since the coordination of our movement is done mainly in an unconscious way, relying on perception without central processors coming into play. Neural networks have proven to be an adequate paradigm for approaching this kind of problem as well as others at the subsymbolic level. We believe that the connectionist and symbolic perspectives to AI should be taken as mutually supporting approaches to the same problems, rather than as competitive areas, as is often the case. Hybrid systems integrating both perspectives appear to be the right track to follow.

This emphasis on perception and robotics has obtained a satisfactory response in terms of the number of submitted papers, as compared with previous conferences.

Consequently, a section on perception is included in Volume I, and in Volume II more than 20 papers can be found in sections devoted to perceptual robotics, robot motion planning, and neurofuzzy approaches to robot control.

The papers included in this volume were presented at IEA/AIE-98 which was held in Benicàssim, Castellón, Spain on June 1-4, 1998. The event was sponsored by the *International Society of Applied Intelligence* —which promotes this conference series—, Universidad Jaume I de Castellón —the hosting institution— and Universidad Nacional de Educación a Distancia, in cooperation with several international associations such as AAAI, ACM/SIGART, ECCAI, and AEPIA, among others. Support for this event has been provided by Fundació Caixa Castelló-Bancaixa, Ministerio de Educación y Ciencia, Fundació Universitat Empresa of the Universidad Jaume I, and Silicon Graphics Computer Systems.

We would like to express our sincere gratitude to the members of the organizing and program committees, to the reviewers, and to the organizers of invited sessions for their invaluable effort in helping with the preparation of this event. Thanks also to the invited speakers, Michael Brady and Bob J. Wielinga, with particular gratitude to Roberto Moreno-Díaz, for their original papers given as plenary lectures and appearing in this book. Thanks also to Moonis Ali, president of ISAI and IEA/AIE-98 general chair, for his constant support. The collaboration of the Technical Committee on Robot Motion and Path Planning of the *IEEE Robotics and Automation Society* deserves a special mention, as well as Toshio Fukuda, president of this society, for his help in the review process. Also, thanks to Springer-Verlag and particularly to Alfred Hofmann for an already long and fruitful collaboration with us. We sincerely thank all authors for making the conference and this book possible with their contributions and participation.

Finally, the editors would like to dedicate this book to the memory of Núria Piera, who promoted research on qualitative reasoning across Spain and Europe and could not see by herself the success of her last organized session, since she had to move to her definitive dwelling.

The theme for the 1998 conference was *New Methodologies, Knowledge Modeling and Hybrid Techniques*. Our focus has been on methodological aspects in the development of KBS's, knowledge modeling, and hybrid techniques that integrate the symbolic and connectionist perspectives in AI applications. The global assessment of the contributions contained in these two volumes is reasonably positive. They give a representative sample of the current state of the art in the field of Applied Artificial Intelligence and Knowledge Engineering and they clearly illustrate which problems have already been solved or are on the way to being solved and which still present a challenge in the serious enterprise of making Applied Artificial Intelligence a science and an engineering discipline as unequivocal and robust as physics or matter and energy engineering. We hope that these volumes will contribute to a better understanding of these problems and to expedite the way to their solution for the well-being of humankind with the advent of the third millennium.

Angel Pasqual del Pobil
José Mira Mira

March 1998

Table of Contents, Vol. I

1 Methodological Aspects

2 Knowledge Modeling

3 Formal Tools

Fuzzy Knowledge Representation and Inference

Uncertainty, Causal Reasoning and Bayesian Networks

Qualitative Reasoning

Neural Networks

Evolutionary Computing

Object-Oriented Formulations

Hybridization Techniques

Context-Sensitive Reasoning

Multi-Agent Systems

Divisible Problems

4 Generic Tasks of Analysis

Perception

Natural Language Understanding

System Identification

Monitoring

Fault Diagnosis

Predictive Models

Table of Contents, Vol. II

1 Synthesis Tasks

Spatial, Temporal and Spatio-Temporal Planning and Scheduling

Motion Planning for Robots

System Configuration

2 Modification Tasks

Knowledge-Based Control Systems

Dynamic Systems Supervision

Intelligent Supervision

Predictive Control Systems

Perceptual Robotics

Fuzzy and Neurofuzzy Approaches to Robot Control

Program Reuse

3 Machine Learning

Machine Learning Applications: Tools and Methods

Inductive and Deductive Strategies

Case-Based Reasoning

Learning Advances in Neural Networks

4 Applied Artificial Intelligence and Knowledge-Based Systems in Specific Domains

Multimedia

Human-Computer Interaction

Decision Support Systems

Tutoring Systems

Connectionist and Hybrid AI Approaches to Manufacturing

Modeling and Simulation of Ecological/Environmental Systems

Reports on Real Applications with Significant Findings

5 Validation and Evaluation Criteria

Simulation

A Fuzzy-neural Multiagent System for Optimisation of a Roll-mill Application*

Carlos A. Iglesias[†**], José C. González[‡] and Juan R. Velasco[‡]

[†]Dep. de TSC e Ing. Telemática, E.T.S.I. Telecomunicación
Universidad de Valladolid, E–47011 Valladolid, Spain
{cif@tel.uva.es}
[‡]Dep. de Ingeniería de Sistemas Telemáticos, E.T.S.I. Telecomunicación
Universidad Politécnica de Madrid, E–28040 Madrid, Spain
{jcg,juanra}@gsi.dit.upm.es

Abstract. This article presents an industrial application of hybrid system: the development of a fuzzy-neural prototype for optimising a roll-mill. The prototype has been developed following an agent-oriented methodology called *MAS-CommonKADS*. This prototype has the original characteristic of being agent-oriented, i.e. each learning technique has been encapsulated into an agent. This multiagent architecture for hybridisation provides flexibility for testing different hybrid configurations. Moreover, the intelligence of the agents allows them to select the best possible hybrid configuration dynamically, for some applications whereas no best configuration for all the situations has been determined.

1 Introduction

This article presents a generic philosophy for developing hybrid systems by the usage of agent technology for encapsulating different symbolic and connectionist modules. In addition, we follow a methodological approach to hybrid systems, applying an extension of *CommonKADS* for multiagent systems development that takes into account hybridisation issues in the design model. This paper shows the benefits of this approach in an industrial application, the optimisation of the steel band production of a roll-mill.

The outline of this paper is as follows. In section 2 the rationale of using a multiagent system for hybridisation and the methodological approach that has been followed is discussed. Section 3 describes the proposed architecture for a multiagent fuzzy-neural model. Section 4 shows how this generic architecture has been applied to an industrial application and the experimental results. Finally, some conclusions are drawn in section 5.

* This research is funded in part by the Commission of the European Community under the ESPRIT Basic Research Project *MIX: Modular Integration of Connectionist and Symbolic Processing in Knowledge Based Systems*, ESPRIT-9119, and by the Spanish Government under the CICYT projects TIC91-0107 and TIC94-0139.

** This research was partly done while the first author was visiting the Dep. Ingeniería de Sistemas Telemáticos (Universidad Politécnica de Madrid).

2 An agent-oriented methodological approach

The approach to hybrid systems development in the MIX project [GVI95] has consisted on encapsulating the learning components into agents. This provides a *component-based* approach to hybrid system design, since the interfaces between the components are standardised. The designer of the hybrid system can select the components (agents) to combine. The intelligence of the agents allow that the agents themselves can select their configuration.

The development of the fuzzy-neural prototype has followed the agent-oriented methodology *MAS-CommonKADS* [IGGV97]. This methodology allows the analysis and design of application from an agent-oriented perspective.

This methodology extends the knowledge engineering methodology *CommonKADS* [SWV94] with techniques from object-oriented and protocol engineering methodologies. The methodology consists of the development of seven models: *Agent Model*, that describes the characteristics of each agent; *Task Model*, that describes the tasks that the agents carry out; *Expertise Model*, that describes the knowledge needed by the agents to achieve their goals; *Organisation Model*, that describes the structural relationships between agents (software agents and/or human agents); *Coordination Model*, that describes the dynamic relationships between software agents; *Communication Model*, that describes the dynamic relationships between human agents and their respective personal assistant software agents; and *Design Model*, that refines the previous models and determines the most suitable agent architecture for each agent, and the requirements of the agent network.

The analysis phase [IGVE96,IGV96] has consisted of the development of a generic task model for fuzzy-neural hybridisation, the identification and description of the agents for carrying out these tasks in the *agent model*, and the description of the interactions and services between the agents in the *coordination model*. The tasks that require knowledge have been described in an *expertise model*.

The design of these knowledge tasks has taken into account the possibility of integrating hybrid techniques. The problem of integrating machine learning issues in *CommonKADS* has been tackled in [VdVA92,HLA95]. The approach has consisted of identifying the different learning methods that can be associated with an inference source [Hil95] and identifying learning goals [VdVA92] (e.g. tuning fuzzy sets) and defining learning tasks to achieve these goals and assign these tasks to learning agents. When there is several ways of performing a learning task, and there is no best known method, we have identified several strategies:

- An implementation of all the possibilities (or the most promising alternatives) into one intelligent agent, and the intelligent agent selects at run-time the most suitable method for performing the task. We have to define explicit criteria for selecting a method. In other case, these criteria must be learnt at run-time.

- Definition of one agent per method, and definition of an additional agent which selects the best result. That is, all the methods are executed in parallel, and the best one is selected in each execution (*stand-alone model*).
- A refinement of the previous method, called the *intelligent stand-alone model.* The intelligent stand-alone model is a compound of an agent *manager* and a set of learning agents which are included into a public group. The *manager* selects the best system or knowledge base that has been learnt. It can also select the different regions on which the learning agents should work. Depending on the results obtained by the different learning agents, the *manager* can learn which agents work better under certain circumstances. So, this *manager* performs meta-processing for determining the regions (macro-states) on which the learning agents will work.

The main benefits of the *intelligent stand-alone model* are that the *manager* is able to perform a meta-processing according to predefined beliefs (e.g. knowledge about different regions of the problem and knowledge) and dynamically learn and update its beliefs. The manager can also try different strategies to achieve its goals, as getting a new definition of the regions, not working anymore with regions, working with some knowledge base, not working anymore with some learning agent, etc.). This approach changes the traditional approach of manual testing, providing high flexibility for defining experiments.

In addition, the *learning agents* are able of evaluating the results they obtain. An interesting and useful feature is that the *learning agents* are able of changing their own parameters to achieve their goal (getting a better performance). For instance, our C4.5 agent (section 3) is able to modify the confidence level parameter with the strategy *generate-and-test.*

3 Architecture of the generic multiagent fuzzy-neural model

The main characteristics of this model are:

- Two learning tasks are identified: improving fuzzy sets and learning fuzzy rules. For each learning task, an *intelligent stand-alone model* is used, consisting of a manager and a set of learning agents.
- The combination of the fuzzy sets and the fuzzy rules is carried out by another *intelligent stand-alone model.* The manager of this configuration is called the *FZ_Engine* agent and the learning agents are the managers of the previously defined learning tasks (i.e. improving fuzzy sets and learning fuzzy rules) as shown in figure 1.
- The agent *FZ_Engine* carried out a partition of the complete problem space into a set of regions. The identification of regions can be carried out by neural nets and by fuzzy clustering algorithms.
- The agent *FS_Manager* is the manager of the fuzzy set improvement task. These fuzzy sets can be extracted by neural net agents (e.g. *NN_Get_FS_n* and fuzzy clustering agents (e.g. *FZ_Get_FS_n*) , in on-line or off-line mode.

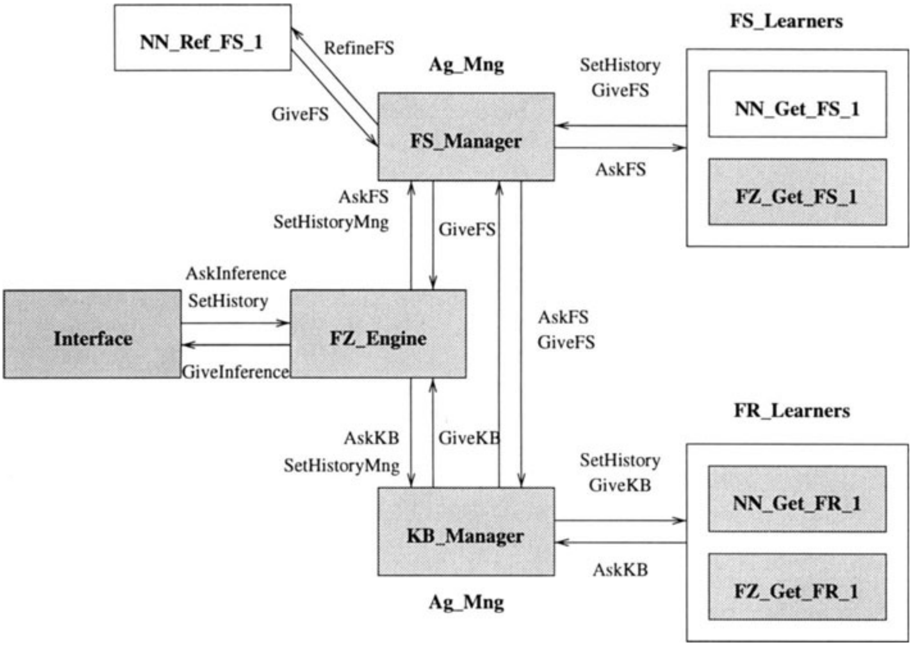

Fig. 1. Event flow diagram of the generic model. Shadowed boxes show the implemented prototype.

- The agent *KB_Manager* is the manager of the fuzzy rule extraction task. These rules are acquired inductively by neural nets agents (e.g. *NN_Get_FR_n*) or symbolic induction (e.g. *FZ_Get_FR_n*) in each region of the problem space, in on-line or off-line mode.

The hybrid systems that deal with fuzzy sets and fuzzy knowledge bases are, in their turn, built from an association/cooperative model: the so-called *intelligent stand-alone model.* The agents defined for this fuzzy-neural model will be reused for other multi-hybrid models (fuzzy logic, case based reasoning and neural networks) [ALLM96] in the different applications of the MIX project.

4 The roll-mill application

The Roll-mill application [Ale95,PA96] has been selected to test the generic hybrid model. In such process, the output of the intelligent system is the suggested rolling force to optimise the steel band production. Twelve parameters are used as the input of the force predictor: three discrete and nine continuous.

A data file with 11054 vectors (each vector with twelve inputs and the best output for them) has been used to train the system. 8866 vectors (80.2% data) constituted the train set, and the rest (2188), the test set. The value to optimise is the relative error as described in [Ale95].

Three experiments have been carried out: a pure connectionist system with unclassified data (section 4.1), a pure connectionist system with clusters (section 4.2) and a multiagent fuzzy-neural prototype (section 4.3).

4.1 Pure Connectionist System with Unclassified Data

A simple neural net was trained with the complete training set. For this purpose, a fully connected feed-forward Multi-Layer Perceptron with four layers (12*12*6*1) was used. Learning was achieved by using the back propagation algorithm (1000 epochs and a learning rate of 0.2). The average error resulting from this experiment was 12.1%.

4.2 Pure Connectionist System With Clusters

Three of the 12 input values, N_PASS, LUBRICATION and ROLL_FINISH, were discrete, the first two having 3, and the last one having 21 different values. So, it was probable that these values had a significant meaning for the process. To find out about this, the learning and test data sets were divided into six classes with different combinations of the two first discrete variables (there were no samples with N_PASS=0.5). Then, a net with the same architecture was trained for each of these sets and the nets were tested with the test sets of the corresponding class. The average error obtained was 12.3% (slightly higher).

By examining the data graphically, it could be seen that two of the input variables, WIDTH and THICKNESS, showed a higher correlation with the output variable than the rest. So, these two were selected as the basis for clusterisation. By using the ISODATA algorithm, the bi-dimensional problem space was divided into 7 classes as shown in table 1, together with the sizes of the corresponding data sets.

CLASS	THICKNESS	WIDTH	SIZE	
			Train	test
1	[0.00, 0.50]	[0.00, 0.30]	1310	334
2	[0.00, 0.30]	[0.30, 0.55]	1971	458
3	[0.00, 0.50]	[0.55, 0.75]	2044	522
4	[0.00, 0.50]	[0.75, 1.00]	1214	276
5	[0.50, 1.00]	[0.50, 1.00]	971	253
6	[0.50, 1.00]	[0.00, 0.50]	748	190
7	[0.30, 0.50]	[0.30, 0.55]	608	155

Table 1. Data clusterization

A net with the same architecture as before was trained for each of these classes and tested with the test data of the class. Results are shown in table 2

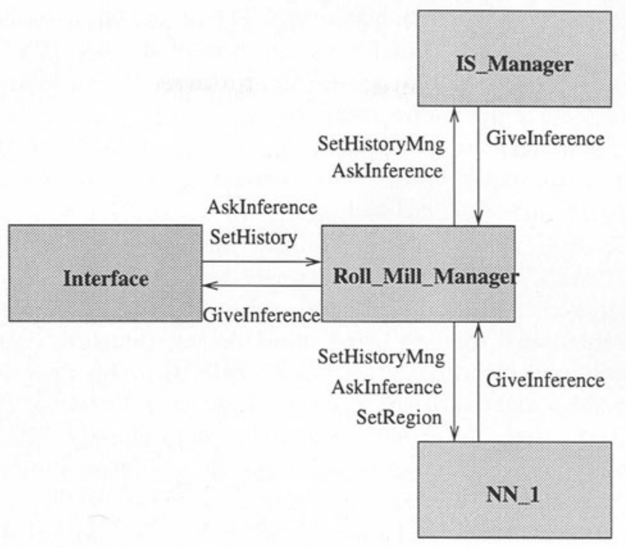

Fig. 2. Event flow diagram of the roll-mill model. The *IS_agent* is developed in fig. 1.

(column 2). The table shows that only the results for class 6 were better than with the initial network (column 3), but the average for these classes was worst.

The Stuttgart Neural Nets Simulator (SNNS)[1] was used for the implementation of pure connectionist systems.

4.3 Multiagent fuzzy-neural prototype

The generic architecture describe in section 3 has been instantiated for the roll-mill application. The prototype only considers some of the agents of the generic architecture, marked with shadowed boxes (figure 1). This model has been extended with another intelligent stand-alone configuration shown in figure 2. This configuration consists of an agent manager, a connectionist agent with clusters (*NN_1*), described in section 4.2, and an *intelligent stand-alone model IS_Manager* despicted in figure 1 and described below. The *MAST* multiagent platform[2] has been used for the implementation of the multiagent system.

The *FZ_Get_FS_1* agent uses the ISODATA fuzzy clustering algorithm to define trapezoidal fuzzy sets for every continuous variable of the system. This iterative algorithm classifies data according to their distance to a characteristic point of the cluster called centroid, based on the minimisation of the cluster volume and maximisation of its density. Centroids has been selected as the most

[1] The *SNNS* Stuttgart Neural Nets Simulator has been developed by the University of Stuttgart, at `http://vasarely.informatik.uni-stuttgart.de/snns/snns.html`

[2] The *MAST* multiagent platform has been developed by the Technical University of Madrid and it is available at `http://www.gsi.dit.upm.es/~mast`

representative peaks of the data histogram. The algorithm is based on the minimisation of the sets volume and maximisation of density [VV94,GG89]. For instance, the fuzzy sets obtained by the algorithm for the variables width of the band and rolling force are shown in figure 3.

The *FZ_engine* agent uses the fuzzy engine FuzzyCLIPS 6.02A[3]with standard max-min as compositional rule of inference and COG (Centre of gravity algorithm) algorithm for defuzzification.

The *FS_Get_FS_1* agent extracts the fuzzy rule using the C4.5 algorithm with pre-processed inputs and outputs as described in [IGV96]. For rule extraction C4.5 algorithm has been used as follows . First, historic numeric data file has to be transformed in a pseudo-fuzzy data file, by changing every number in the file by the label of the fuzzy set with bigger membership value. This data file feeds the C4.5 algorithm to create an Induction Decision Tree and a set of rules. C4.5 algorithm, that has been designed to classify data, can be used to create prediction systems (as in this case) if predicted values are discrete. In this case, predicted values are the fuzzy sets of the predicted variable. We have generated two knowledge bases of fuzzy rules. Each knowledge base has a different confidence factor. This factor sets the pruning level in the decision tree (intermediate step to create rules) and rules generation. The used factors are 25 (default) and 50.

The *NN_Get_FR_1* agent uses a Kohonen self-organising map, following the method for rule extraction described in [Kos92,Hun93], though the results have been worse in this application than the obtained with the *FZ_Get_FR_1* agent.

The table 2 shows the results obtained for the neural nets and fuzzy knowledge bases generated. The fuzzy KBs have been tested using the specific (S) generated fuzzy KB for every cluster and the general (G) fuzzy KB generated for the whole data. In addition, the results are shown for the values 25 and 50 of the confidence factor of C4.5.The last column remarks the best value of the error rate for each cluster, with the global mean error for these cases.

Data	Neural Nets		Fuzzy KBs				**Best**
	Spec.	Gen.	S-25%	S-50%	G-25%	G-50%	
C1	14.1	13.8	9.48	10.19	8.60	**8.43**	8.43
C2	14.6	14.1	13.43	5.90	5.82	**5.12**	5.12
C3	12.1	11.1	**5.93**	8.41	8.25	7.03	5.93
C4	14.7	13.2	7.31	12.67	9.79	**6.50**	6.50
C5	11.1	10.8	7.49	5.47	5.86	**5.81**	5.81
C6	**8.7**	9.6	9.93	11.36	13.32	10.35	8.70
C7	11.1	10.0	13.64	13.10	11.48	**7.03**	7.03
Global	12.77	12.16	9.29	8.94	8.38	6.92	**6.49**

Table 2. Average error over clusterised and global data

[3] FuzzyCLIPS has been developed by the Knowledge Systems Laboratory, National Research Council, Canada, available at `http://ai.iit.nrc.ca/fuzzy/fuzzy.html`

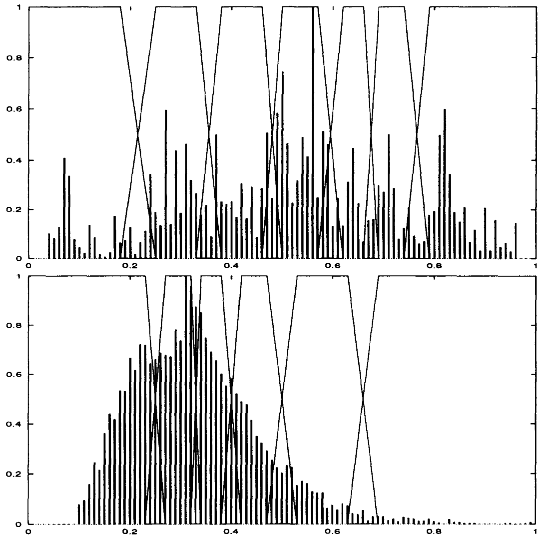

Fig. 3. *Width of the band* and *Rolling force* fuzzy set distributions

5 Conclusions

This article has shown a distributed approach to hybridisation that takes advantage of the intelligence of the components for determining the hybrid configuration at run-time. The application of a methodology contributes to the reuse these configuration and the developed agents.

The main results obtained in this work from a conceptual point of view are:

- The adoption of a methodology for developing hybrid systems for an agent-oriented perspective.
- The definition of a reusable configuraton called the *intelligent stand-alone model* for building hybrid systems.
- The design and implementation of a hybrid fuzzy-neural model which is a first attempt towards an association model. This model has the additional advantage of being open, and it can be easily combined with other paradigms.
- The design and implementation of a reusable agent class Selector for intelligent stand-alone models.
- A reusable implementation of services for learning agents.
- The implementation of reusable fuzzy and neural agents. These agents have been reused in a fuzzy logic, case based reasoning and neural networks hybrid model [ALLM96].

From the analysis of performance of this experience, these results can be remarked :

- We have obtained better results than with the pure neural approach. The error rate has been reduced to 50%.
- Our premise of getting better results dividing the data space in regions seems to be promising. More extensive work has to be done, however, to determine a better splitting of the problem space into regions by using both symbolic and connectionist methods.
- More extensive work is needed in the neural part for extracting fuzzy sets and rules.
- Fusion models need to be validated and integrated in our model.

Acknowledgements

We would like to thank all the members and collaborators of the MIX project. They have contributed to a very fruitful and friendly atmosphere for researching.

References

[Ale95] Frederic Alexandre. Report on definition and specification of industrial applications: Steel rollmill application. Technical Report MIX/WP3/INRIA/1.0, INRIA Lorraine, March 1995.

[ALLM96] F. Alexandre, A. Labbi, Y. Lallement, and M. Malek. Evaluation report on the case-based reasoning / neural networks hybrid architecture. Technical Report Deliverable D8, IMAG - INRIA, 1996.

[GG89] I. Gath and A. B. Geva. Unsupervised optimal fuzzy clustering. *Trans. on Pattern Analisys and Machine Intelligence*, PAMI-11(7):773–781, July 1989.

[GVI95] José C. González, Juan R. Velasco, and Carlos A. Iglesias. A distributed platform for symbolic–connectionist interoperation. In Ron Sun and Frederic Alexandre, editors, *Proceedings of the IJCAI'95 Workshop on Connectionist-Symbolic Integration: From Unified to Hybrid Approaches*, pages 94–99, Montreal, Canada, August 1995. International Joint Conference on Artificial Intelligence. (An extended version of this paper has appeared in *Connectionist Symbolic Integration*, Lawrence Erlbaum Associates, 1996.).

[Hil95] Melanie Hilario. KADS: A framework for neurosymbolic integration? Technical Report UNIGE-AI-95-04, Computer Science Department. University of Geneva, 1995.

[HLA95] M. Hilario, Y. Lallement, and F. Alexandre. Neurosymbolic Integration: Unified versus Hybrid Approaches. In M. Verleysen, editor, *Proceedings European Symposium on Artificial Neural Networks*, Brussels (Belgium), 1995.

[Hun93] C. Hung. Building a neuro-fuzzy learning control system. *AI Expert*, 8(11):40–49, 1993.

[IGGV97] Carlos A. Iglesias, Mercedes Garijo, José C. González, and Juan R. Velasco. Analysis and design of multiagent systems using MAS-CommonKADS. In AAAI'97 Workshop on Agent Theories, Architectures and Languages, Providence, RI, July 1997. ATAL. (An extended version of this paper will be published in *INTELLIGENT AGENTS IV: Agent Theories, Architectures, and Languages*, Springer Verlag, 1998,.

[IGV96] Carlos A. Iglesias, José C. González, and Juan R. Velasco. Prototype of a fuzzy-neural hybrid model. evaluation report. Technical Report MIX/WP2/UPM/3.0, Dep. Ingeniería de Sistemas Telemáticos, E.T.S.I. Telecomunicación, Universidad Politécnica de Madrid, January 1996.

[IGVE96] Carlos A. Iglesias, José C. González, Juan R. Velasco, and Klaus Eder. Prototype of a fuzzy-neural hybrid model. Implementation report. Technical Report MIX/WP2/UPM/2.0, Dep. Ingeniería de Sistemas Telemáticos, E.T.S.I. Telecomunicación, Universidad Politécnica de Madrid, January 1996.

[Kos92] Bart Kosko. *Neural Networks and Fuzzy Systems. A Dynamical Systems Approach to Machine Intelligence.* Prentice-Hall, Englewood Cliffs, 1992.

[NK95] Detlef Nauck and Rudolf Kruse. NEFCLASS - a neuro fuzzy approach for the classification of data. In *Symposium on Applied Computing during 1995 ACM Computing Week*, Nashville, March 1995.

[PA96] Nicolas Pican and Frédéric Alexandre. Artificial neural networks for the presetting of a steel temper mill. *IEEE Expert*, pages 22–27, February 1996.

[SWV94] A. Th. Schreiber, B. J. Wielinga, and J. M. Akkermans W. Van de Velde. CommonKADS: A comprehensive methodology for KBS development. Deliverable DM1.2a KADS-II/M1/RR/UvA/70/1.1, University of Amsterdam, Netherlands Energy Research Foundation ECN and Free University of Brussels, 1994.

[VdVA92] Walter Van de Velde and Agar Aamodt. Machine learning issues in CommonKADS. Technical report ESPRIT Project 5248 KADS-II/TII.4.3/TR/VUBS/002/3.0, Free University Brussels, January 1992.

[VV94] Juan R. Velasco and F. Enrique Ventero. Some applications of fuzzy clustering to fuzzy control systems. In *Proceedings of the Joint Conference on Information Sciences*, pages 363–366, Pinehurst, NC, USA, November 1994.