Fusion Technology of Neural Networks and Fuzzy Systems: A Chronicled Progression from the Laboratory to Our Daily Lives *

Hideyuki Takagi
Kyushu Institute of Design
4-9-1, Shiobaru, Minami-ku, Fukuoka, 815-8540 Japan
Tel&Fax +81–92–553–4555, takagi@kyushu-id.ac.jp, http://www.kyushu-id.ac.jp/~takagi

Abstract

We chronicle the research on the fusion technology of neural networks and fuzzy systems (NN+FS), the models that have been proposed from this research, and the commercial products and industrial systems that have adopted these models. First, we review the NN+FS research activity during the early stages in Japan, the US, and Europe. Next, following the classification of NN+FS models, we show the ease of fusing these technologies based on the similarities of the data flow network structures and the non-linearity realization strategies of NNs and FSs. Then, we describe several models and applications of NN+FS. Finally, we introduce some important and recently developed NN+FS patents.

1 Introduction

During the late 1980s, the number of researchers and engineers interested in neural networks (NNs) and fuzzy logic (FL) increased, dramatically introducing the NN and FL technologies into several application fields. Both technologies are widely used and are considered fundamental engineering technologies. As the two technologies developed at same time, researchers and engineers studied the technologies' similarities and complementarities and began developing a fusion model.

Although the first paper that used both *NNs* and *fuzzy sets* as keywords can be found in 1974, research on fusing NNs and fuzzy systems (FSs) essentially began in 1988 and has dramatically increased since then. Within several years, NN+FS fusing technology was already being used in commercial products and industrial systems. The practicality of technology, introduced at the beginning of this paper, is supported by the number of real-word applications based on this technology. Since its introduction, the fusion technology has widely expanded into application fields requiring such tasks as control, operations research, retrieval, clustering, speech recognition, and others [15]. We show the spread of this trend from Japan to the US and Europe in next section. Figure 1 shows the increasing number of papers and conferences that using both the keywords *NN* and *FS* during the early 1990s.

Furthermore, genetic algorithms (GA) were introduced to the fusing technologies and several combinations of GA and FS and of GA and NN have been proposed since 1989. Consumer products that use these GA+NN and GA+FS cooperative models have also been put on the market.

In this paper, we describe the fusing technologies of Soft Computing from a historical and R&D perspective; we especially focus on NN+FS research models and application tasks in Japan where they were primarily researched and developed. For the past 10 years, fundamental patents on NN+FS have been established. We introduce some of these patents in section 5.

*This article appeared in International Journal of Applied Mathematics and Computer Science, vol.10, no.4, pp.647–673 (2000).

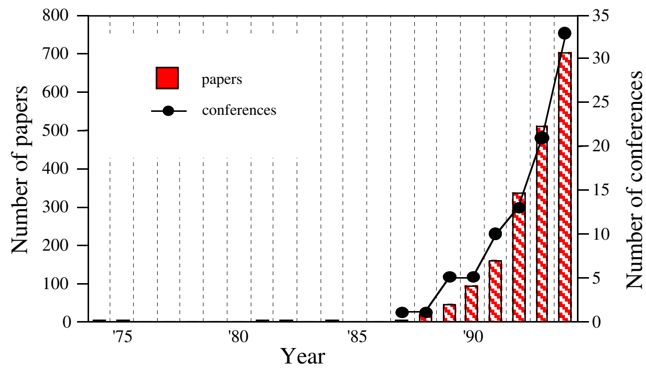


Figure 1: The number of NN+FS papers and conferences in early stage of NN+FS research.

2 Historical View of the R&D of NN+FS

2.1 NN+FS research in early stage

There is a significant difference between NN+FS research during the early stage and that after the late 1980s. Much of the research during the early stage introduced the concept of fuzziness to neurophysiology.

The first paper that used both *NN* and *fuzzy* as keywords was written by the Lee brothers in 1974. They proposed a neuron model of multi-inputs and multi-outputs [38, 39]. They generalized the McCulloch & Pitts neuron model whose output characteristic is a binary step function to deal with intermediate values. They also showed the possibilities of fuzzy automata and a λ-fuzzy language recognizer.

The study of the correlation to neurophysiology was further by Butnariu, who proposed a model of the eighth sensory and hearing-vestibular nerves, by using L-fuzzy automata [4], and Rocha et al., who employed tools such as fuzzy language, fuzzy entropy, and fuzzy automata in their attempt to analyze the nervous systems [55, 56, 57, 58]. Meanwhile, Chorayan presented its application as a method to analyze the neuron of the frog visual central analyzer and the crayfish sixth abdominal ganglion [5].

These researches introduced the fuzzy concept as a tool to the neural science field. Recently, there has been very little reported on the development of FL applications to neurophysiology. Instead, Shiue et al. presented an application of automata to a fuzzy learning machine that consisted of units of fuzzy neuron [62]. When the max–min operation is substituted by the sum of products operation, stochastic neural automata can be obtained.

2.2 NN+FS research in Japan

The second IFSA conference was held in Tokyo in 1987. Before this conference, FL control had already been applied to several industrial systems in Japan, for example, the auto-drive of the Senday subway system, the injection control of chemicals at the Sagamihara filtration plant, and the auto-operation of a dredger pump. The demonstration of a fuzzy reasoning chip at the IFSA conference as well as the presentation of these industrial applications showed the practicality of FSs to the conference participants and the representatives of Japanese industry. This conference triggered the first fuzzy boom.

In contrast, the second NN boom originated in the US during the 1980s was propagated to Japan. Two thousand participants attended the first IEEE conference on NNs in 1987. When its conference report arrived in Japan soon afterward, it triggered the Japanese NN boom. Numerous NN research was presented at several domestic Japanese conferences during the spring of 1988.

The timely overlap of the FS and NN booms provided the backdrop that sparked the interest in advancing the study of NN+FS technologies among the Japanese researchers.

Both technologies approximate non-linearity by combining membership functions, sigmoidal functions, or other base functions. Since NNs and FSs have different advantages in learning function and in explicit knowledge expression of fuzzy rules, respectively, they can complement each other. These points provide some of the technical background that stimulated interest in NN+FS research.

During the economic boom of the late 1980s, consumers demanded high-performance products despite increases in its cost. Manufacturers added new sensors to their products and developed many consumer products with a high degree of functionality. Consequently, as the number of input variables, such as the dimensional number of an input space, for a product increased, the number of fuzzy rules exponentially grew, increasing the complexity and development costs. In this scenario, the development an auto-designing FS became a necessity. This was the primary reason why the learning function of NNs attracted the attention of researchers in FL society.

The first NN+FS paper proposing auto-designing an FS using NNs was presented in May of 1988 [14, 63, 66] and became a national story. This research started an enthusiastic NN+FS R&D race which produced several papers during the summer of that year [64, 65]. The NN+FS technology was easily expected to become a practical technology because both NNs and FSs were independently practical technologies. Moreover, the subject of these NN+FS papers was aimed at auto-designing FSs using NNs, especially toward Japanese industry, which was particularly interested in the application of this technology. Japanese companies took initiative in the R&D of NN+FS technology from proposing its models to developing its systems and products. Many seminars and symposiums on NN+FS were held between 1988 – 1991, and the mass media frequently reported on its development.

Many consumer products using the fusing technology have been introduced to the market starting with neuro-fuzzy washing machines in 1991. The technology was applied–not to only consumer products but–also to industrial systems such as the rolling mill and chemical process of a pulp factory described in section 4.4. A series of new technical terms such as *neuro-fuzzy* and *neural&fuzzy* in advertisements quickly followed those of FL-based consumer products in 1990. These terms puzzled consumers and the industrial nonchalance toward naming technologies that exceeded the consumers' understanding was criticized. Today, such names are not endorsed even if they are used. NN+FS technology is pervasive and it is difficult to survey which products or systems use the technology reported in academic and business articles.

Japanese academic societies took the lead in promoting NN+FS technology by sponsoring several seminars and publishing several tutorials papers and special issues on NN+FS since 1988. *Soft Computing* began to take the place of NN+FS around 1993. It is difficult to find Japanese seminars or meetings on only NN+FS now. The proof of this trend can be found in the naming of an international conference biennially held in Iizuka City, Fukuoka, Japan. IIZUKA'88 was a workshop on fuzzy applications; IIZUKA'90 was the conference on FL and NNs; IIZUKA'94 was the conference on FL, NNs, and soft computing; and IIZUKA'96 was conference on soft computing.

2.3 NN+FS research in the US

Interest in FS increased before the interest in the NN+FS research increased in the US and Europe. In Japan, 1990 was called *the Year of Fuzzy.* Many consumer products adopting FL were already on the Japanese market. The use of the word *fuzzy* was widespread throughout Japan and came to be used in everyday conversation. In those days, *fuzzy* was social phenomena. Besides the novelty of the word, consumers chose the aforementioned high-performance products because of a booming economy and strong consumer buying power.

The success story of FS for Japanese businesses reached the US and Europe, and American and European researchers began to pay attention to the technology in 1991 and 1993, respectively. Since the interest in NNs had already been high in the US, the number of NN+FS papers increased soon.

Of course, there were pioneering works in the US before this NN+FS boom. Besides Lee brothers' paper, published in 1974, Kosko introduced the fuzzy concept to NN and the NN learning algorithm: FAM (fuzzy associate memory) that memorizes one fuzzy rule [35] and fuzzy cognitive map that extends the relational weight, $\{0,1\}$, of the cognitive map to $[0,1]$ and introduces Hebbian law as its learning algorithm [34].

The contribution of the Berkeley Initiative in Soft Computing (BISC) Program and its missionary, L.A. Zadeh, in expanding NN+FS research in the US was big. The BISC is a liaison program of the Computer Science Division at UC Berkeley that succeeded the Berkeley Computer Science Affiliates (BCSA) in 1991. It accepted registered researchers and research organizations around the world and provided an environment to exchange information on soft computing primarily through the BISC mailing list. L.A. Zadeh has propagated the wide concept of soft computing that is not bounded by only FL or NNs. One of the educational aims of the BISC seminar series is to host weekly lectures and to invite speakers from UC Berkeley or from around the world and to educate its students to accept FL, NNs, and other technologies without bias and reluctance. During the 1990s, some of these students started to present their NN+FS researches here and there.

The second reason why NN+FS research increased in the US in those days was that FL-only research was considered difficult to be widely accepted but the idea of combining FL with well-accepted NNs was thought to be easier. Although much attention was directed toward the applications of FL around 1991, this was mainly in industry, and skepticism and criticism of FL still remained in the US academic societies. The Japanese success story of FL business significantly influenced its acceptance in the US; however, it was combination of FSs with NNs that persuaded the acceptance by American researchers and academic societies.

2.4 NN+FS research in Europe

The start of wide acceptance of FL applications in Europe was later (1992) than in the US, although the first historically famous FL applications in the laboratory and industry were realized in Europe during the 1970s. This delay in Europe arose from the conservative action that hesitated to accept the new technologies until their undoubted effectiveness was proven, while in new technology was readily embraced.

The mass media reported the Japanese FSs success story in Europe, especially in Germany, since the end of 1990. The *HighTech* journal issued special issues on FL twice, computer journal, *MC*, featured FS on its cover page, and TV programs introduced several Japanese FS products. These reports increased the German interest in FL, and fuzzy seminars have been held in several cities in Germany since 1991.

During this time, many companies assigned a few their employees to the research FL to diversify their operations should FS significantly impact their business. However, the skepticism of FL was prevalent in many companies during this early stage and company researchers assigned to research FL spent much of their time and energy explaining its features and benefits and persuading their colleagues' acceptance.

The exception was Siemens AG. They introduced FL to their company on a large scale and became the leading European company of FL applications. They started an FL task force in Munich and developed several FL applications for different divisions. Following the task force, three NN, FS, and GA projects started simultaneously, and fusing technologies were applied to their internal applications. Several Siemens applications led other European companies to start introducing FL applications. This is the history of the beginning of NN+FS research in Europe.

The European Laboratory for Intelligent Techniques Engineering (ELITE) Foundation was established in 1992 and started sponsoring the EUFIT conference every year since 1993. The trend of European research could be observed by examining the EUFIT conferences. About 500 participants attended the EUFIT'93. Researchers and engineers who started FL research needed to obtain the latest information on FL research and FS applications from Japan, where both FL research and industrial business was the most active. As NN+FS research and applications were active in Japan at that time, it might be the reason why NN+FS was the representative keyword at EUFIT'94 and why GA was added to the keywords at EUFIT'95.

3 NN+FS models

3.1 Classification of NN+FS models

Hayashi et al. roughly categorized NN+FS models at their fusing level perspective [15]. Since it is visually easy to understand their categorization, we explain these models according Figure 2.

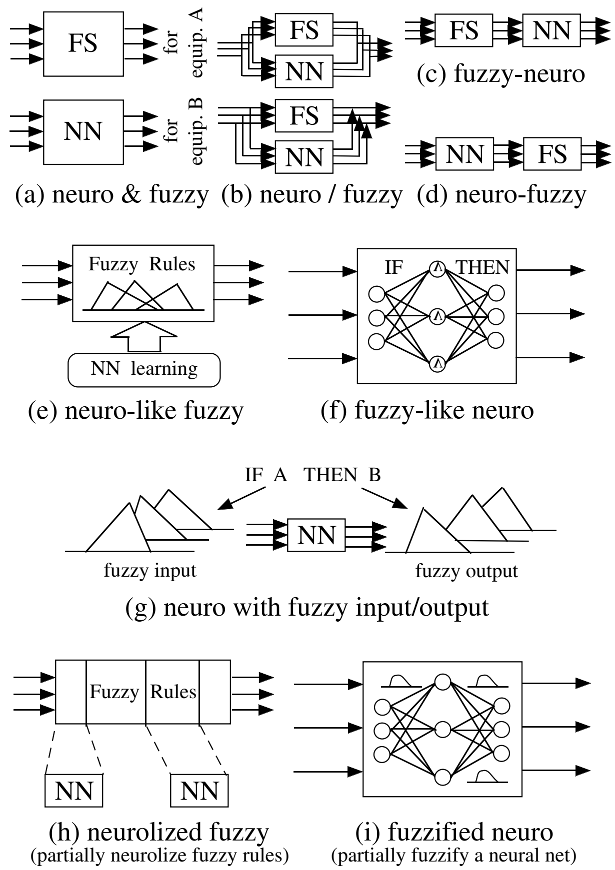


Figure 2: Hayashi & Umono's categorization of NN+FS models [15].

(a) is the case that an NN and FS are independently used in the same system. Air conditioners in section 4.4 (a) adopt this model.

(b) is the case that combines the output of an NN and FS or modifies the output of an NN or FS by the output of an FS or NN, respectively. Washing machines and microwave ovens in section 4.4 (b) adopt this model.

(c) and (d) are the case that an NN and FS are connected in cascade. Electric fans in section 4.4 (c) adopt this model. As this connection can be applicable to pre- or post-processing part of signal processing, this model was applied to several signal processing tasks, such as data analysis [49, 46], image recognition [70, 7, 18], image understanding [60], speech recognition [1] and others.

(e) is the case that designs or tunes an FS using an NN. Washing machines, rice cookers, vacuum cleaners, and photo copy machines in section 4.1 adopt this model. Introducing non-feed-forward or non-backpropagation algorithms [47, 52] and applying NN tuning to non-FS [40, 13, 50, 59] are categorized by this model, too.

(f) is the network expression of fuzzy reasoning [16, 74]. Although most research of this model uses the term NN, some are appropriately called network reasoning; a few researchers consciously avoid the term of NN [76].

(g) is the case that an NN configures fuzzy reasoning rules or conducts fuzzy reasons. Some are a feed-forward NN handling of fuzzy values [9, 10, 32, 67, 68, 19, 20, 21, 23], fuzzy associative memory based on BAM (bi-directional associative memory) [80, 81], and CMAC type of NN dealing with fuzzy rules [53]. An NN conducting fuzzy clustering may be categorized in this type of model, too [28, 72].

(h) is the case that certain parts of an FS are replaced by feed-forward NNs, such as the NN-driven fuzzy reasoning in section 4.1, or vector quantization NN [79]. A simplified model in which only the gains of membership functions are adjusted by NN is one of other models [43]. Besides its application to fuzzy reasoning, fuzzy regressive analysis is categorized by this model [25].

(i) is the fuzzyfication of an NN. Modifying the NN to handle fuzzy numbers of inputs, outputs, and fuzzy weights [26, 27], and fuzzyfying whole learning [17] are categorized in this case.

There are several categorizations of NN+FS with different perspectives [64, 65, 8, 71, 82].

3.2 Why it is easy to fuse NN and FS

Both NNs and FSs realize complex non-linearity by combining and interpolating multiple base functions. Due to these characteristics, both technologies are effective when application tasks are non-linear and difficult to clearly describe using mathematical equations or logic.

Membership functions correspond to the base functions for FS. An input space is fuzzily partitioned by the membership functions that are designed for each input variable. Each partitioned subspace corresponds to each fuzzy rule. The total characteristics of an FS are expressed by synthesizing all rules. A characteristic function of a neuron corresponds to a base function for NNs. The total characteristics of an NN are expressed as a synthesized result of the characteristics of all neurons, too.

The adjustment of both NNs and FSs is conducted by adjusting the characteristics of each element consisting of the whole, i.e. adjusting the shapes of membership functions for FSs, and adjusting the weights among neurons for NNs. We can see that NNs and FSs have same strategy to realize non-linearity, nevertheless, they have different historical backgrounds.

Furthermore, their basic structures are same. Figure 3 looks like an NN. To some, this may look like a neuro-fuzzy system. However, this is a pure FS. A membership function inputs an input value and outputs a membership value. As n membership values are obtained for n input variables, a t-norm operator merges these membership values and calculates a rule strength. The final system output is obtained by weighting consequent parts, y_i, with each rule strength and aggregating them. This general description of a normal FS is along with the flow of Figure 3.

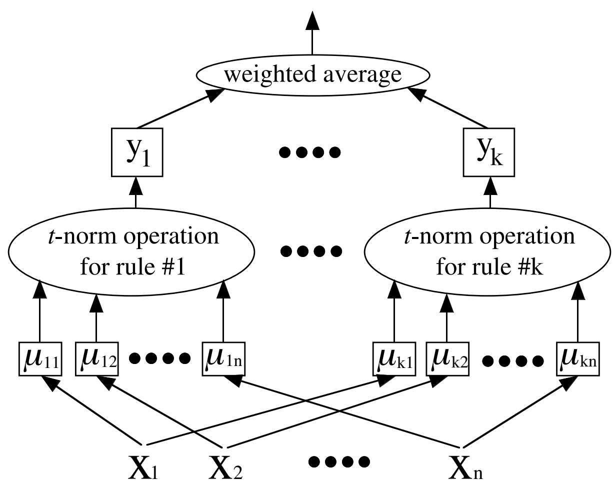


Figure 3: Network expression of a normal FS. The FS structure is essentially similar to that of an NN.

Once we understand that the structure of Figure 3 is that of a feed-forward NN, it is natural to think that backpropagation or other learning algorithms may be applicable to FS tuning and optimize membership functions, μ_{ij}, and subsequent outputs, y_i, based on an error minimum criteria between the outputs of a real FS and design specification. NN+FS models from (e) to (h) in section 3.1 are based on this idea.

The reason why NNs and FSs are fused is due to, not only their similarity, but also their complementarity of their different advantages. FSs deal with explicit logic by describing it in rules, which is difficult for NNs. Conversely, NNs have the learning capability that FSs do not have. When we want to use the both advantages, we choose NN+FS fused models. When we have complete explicit knowledge of the application tasks, a conventional knowledge-based system is the best. When the knowledge is not quantitative but qualitative, a fuzzy knowledge-based system is effective. But, it is needed for the FS to borrow techniques from other technical fields to systematically design membership functions which define the qualitative parts or adapt to a dynamic situation of application tasks. When we do not have knowledge about the application tasks, NNs can obtain implicit knowledge from data using their learning capability. However, even if we have partial knowledge of the tasks, NNs cannot use it directly. When we have partial explicit knowledge of given task and data which keep implicit task knowledge, any single use of AI, FS, or NN technology cannot use every information. In such a case, fusing technologies that complement the capabilities of logic description and learning from data each other is effective.

4 Research & Development

4.1 Designing a FS using NN

There was a strong interest in fusing NNs and FSs to practically automate the design of an FS using NN and, in response, many papers have been presented since 1988.

The first trigger to explicitly apply NNs to design an FS is NN-driven fuzzy reasoning [14, 63, 66] shown in Figure 4. The idea of this model is to make an NN design the entire shapes of membership functions. Unlike conventional FSs, the membership functions of this model are non-linear and are multi-dimensional. The outputs of the NN are the rule strengths of each rule, which are the combinations of membership values in antecedents.

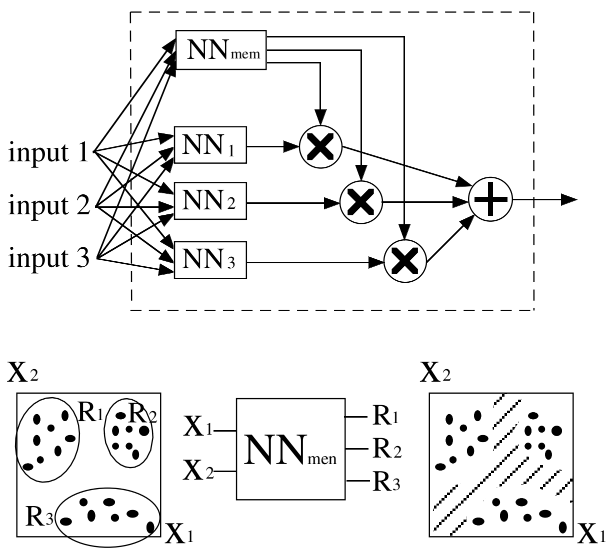


Figure 4: NN-driven Fuzzy Reasoning. Fuzzy partitioning of input space is conducted by NN_{mem} that corresponds to antecedent parts. The figure below illustrates how to train the NN_{mem}. First, the training data are roughly clustered, and then NN_{mem} is trained by the input vector and cluster numbers. The output of the trained NN_{mem} non-linearly fuzzy partitions the input space.

This model was used to control of the Hitachi rolling mill, and the system has run since 1991 [44]. The purpose of the rolling mill is to flatten the metal plate by controlling 20 rolls.

The surface shape of plate reel is detected by scanning (see Figure 5). The scanned shape is put into an NN. The NN categorizes the surface pattern and outputs the similarity between the input pattern and standard template patterns. Since fuzzy control rules are made for each standard surface pattern, the outputs of the NN correspond how the input surface pattern matches to each fuzzy rule, that is, the outputs correspond to rule strengths. In other words, the NN takes the role of antecedent parts of all FL rules. Using the aggregated final output of the FS, the 20 roles are controlled to flatten the plate along with the scanned line.

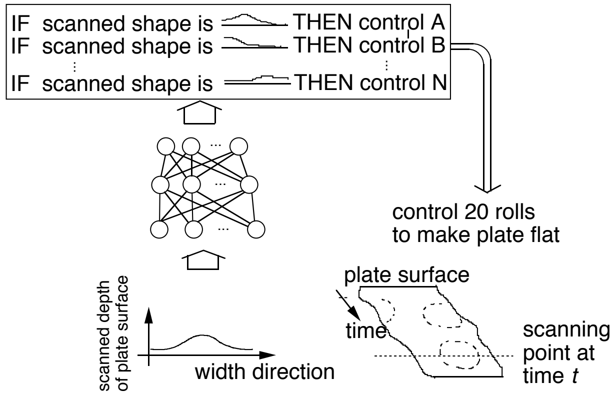


Figure 5: Rolling mill control by an FS and NN. The scanned pattern of plate surface is recognized by an NN. The output value of the each output unit of the NN is used as rule strengths for the corresponding fuzzy rule.

Following the NN-driven fuzzy reasoning, several NN+FS models have been proposed in Japan since the summer of 1988.

Furuya et al. proposed a Neuro Fuzzy System (NFS) that uses the matching levels between an input vector and learned patterns as membership values [9, 10]. Yamaguchi et al. proposed a method that forms membership functions using learning vector quantization (LVQ) [79]. They also proposed a learning fuzzy-NN to control the fuzziness in forming membership functions by introducing fuzzy reasoning into BAM [81].

Unlike these methods that design the shapes of membership functions by NN, Morita et al. proposed a simplified method that fixes the shape of membership functions and tunes only their gains by an NN [43].

Ishibuchi et al. carried out an experiment to construct various membership functions using NN [24], and derived a fuzzy language by which the form of membership function is transformed into that of instructor, and they further derived an interval-valued membership function of which values were broadened.

Watanabe et al. expressed FL rules by NN that is finely tuned through learning by allocation of one fuzzy variable to one neuron, and the weighting factor is so initialized that its sigmoid characteristics comes closer to the predetermined membership function [77].

Another important approach since 1989 was to parameterize the shape of membership functions as Figure 6 and optimize the parameters in the framework of NN learning. It adopted a triangular shape [48], a combination of sigmoidal functions [19, 20, 21], a Gaussian function [22, 75], or a bell shape [29], as the shape of membership functions and tunes the parameters of their shapes.

Sometimes we cannot previously obtain the training data for certain applications. Fuzzy reasoning networks that allow to be tuned by reinforcement learning were proposed for such a case [3, 45]. Several public domain and commercial software on NN+FS were announced. Some of such free software are NFCON-I [45] and FuNeGen 1.0 based on the Fune I model [11].

Among these NN+FS proposals, tuning the parameters of a membership function by an NN became an important approach and was applied to several consumer products. Note the difference between this approach and the NN-driven fuzzy reasoning; the former optimizes the shape parameters of the membership functions using an NN, and the latter optimizes whole shapes of membership functions using an NN.

The approach that tuned the shape parameters was applied to develop commercial products, and these first neuro-fuzzy consumer products were put on the market in 1991. These products include washing machines, vacuum cleaners, rice cookers, clothes dryers, dish washers, electric vacuums pots, inductive heating cookers, oven toasters, kerosene fan heaters, refrigerators, electric

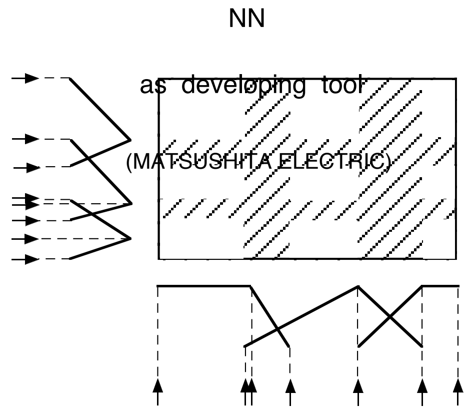


Figure 6: NN adjusts the shape parameters of membership functions.

fans, range-hoods, and photo copiers.

Figure 7 shows an FS that was used in copy machines [69]. The NN is used in development phase, and only the tuned FS is implemented in the final product.

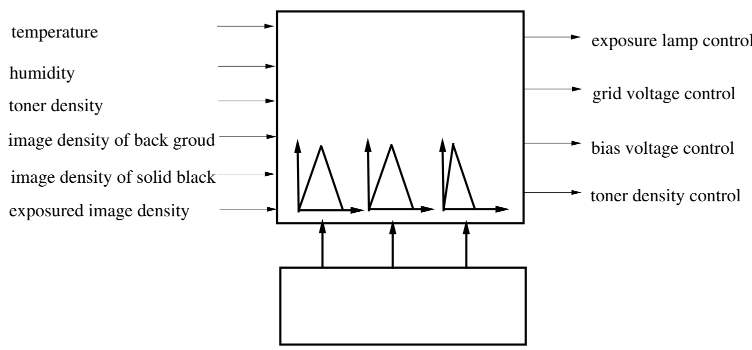


Figure 7: An NN adjusts the parameters of the membership functions for copy machines control. Only an optimized FS is implemented into copy machines, and the NN is used for design by the manufacturer.

4.2 Designing Fuzzy Systems using GA

Auto-designing FSs by NNs is equivalent to parameter optimization. We can then apply GA instead of the NN. The pioneering work of Karr in 1989 [31] was followed by several papers. Today, membership functions in antecedent parts, rule outputs of consequent parts, and the number of rules can be simultaneously optimized by the GA [36].

During the mid-1990s, Korean companies have actively applied this approach for their consumer products and process control.

Samsung's refrigerators, introduced to the market in 1994, adopt two FSs that are designed by GA [33]. The first FS estimates temperature distribution inside the refrigerator, and the second one inputs the estimated temperature distribution and determines the spout position of cool air. Both FSs are TSK models, and GA determines the parameters of the TSK models.

Samsung applied a similar approach to developing washing machines and introduced them to the market in 1995 [33]. Recent washing machines can swirl water very slowly to wash wool and/or lingerie which requires to be usually washed by hand. The motor of the washing machine is controlled by an FL controller whose parameters of I/O membership functions are determined by GA. The GA design is conducted in developing phase and only designed FSs are used in products.

LG Electric has applied GA for several products [61]. Their dishwashers, rice cookers, and microwave ovens use neuro-fuzzy models to estimate the number of dishes, to estimate amount of rice, and to cook well, respectively. These NN+FS systems were also designed by GA.

Other consumer products whose FSs were designed by GA include refrigerators, washing machines, and vacuum cleaners. GA was used to auto-tune the fuzzy rules of these models. GA was also used to acquire fuzzy process control rules for a plant that experiences long delays [61].

The basic framework of the NN and GA approaches to automatically design FSs was established by 1993, and these approaches have become quite practical.

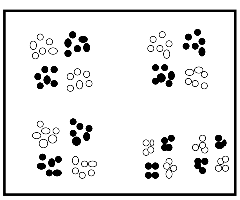
4.3 NN Learning and Configuration Based on Fuzzy Rule Base

One way to reduce the complexity of a given task and increase the performance of a solving system is to embed a priori knowledge of the task into the system. There are several ways to use a priori knowledge for NNs. For example, selecting training data, setting initial weights determined by the knowledge, limiting the searching space, and so on.

NARA is a structured NN that is constructed based on the IF–THEN fuzzy rule structure [68]. Small sub NNs that correspond to each consequent part and an NN that corresponds to whole antecedent parts of the fuzzy rules are combined and form the NARA. This is one way to realize the use of FSs for NNs.

The FL rules describe a priori knowledge of the given task, which is, for example, obtained by analyzing training data. Then, the complexity in each fuzzy partitioned input space is much less than that of the entire given task. Therefore, it becomes easier for the NARA to solve the given task.

Figure 8 shows a toy task and a comparison of performance between conventional NN and the NARA. Nevertheless both systems have same number of synaptic weights, and the NARA shows better performance because of embedding a priori knowledge of this task [68]. Furthermore, it is easier for the NARA to be internally modified to improve its performance because of explicit structure [68].



	training data	test data
conventional NN	50%	50%
NARA	85	83
improved NARA	94	89

Figure 8: Comparison of NARA and normal NN. The task is to classify B/W patterns in left 2-D space. In this case, we find that the input space is roughly separated into four parts. We may describe this feature with four FL rules. NARA is constructed according to this rule structure.

NARA has been used for an FAX ordering system. When retail electric shops order goods from a dealer of Matsushita Electric, they complete an order form by hand and send it by FAX. The FAX machine at the dealer site passes the FAX image to the NARA. The NARA recognizes the hand-written characters and sends character codes to the delivery center (see Figure 9).

This FAX ordering system is requested to have high recognition rates, because it is used for business customers. This is main reason why NARA is used, because of its high character recognition performance proven in a public contest; the NARA was one of three winners at the public character recognition contest sponsored by National laboratories under the Ministry of Posts and Telecommunications [73]. Today, the FAX-OCR part is on the market.

Methods that dynamically change the NN learning rate or other NN parameters by FL rules are other approaches to improve NN performance based on FL rule base [2, 78, 12].

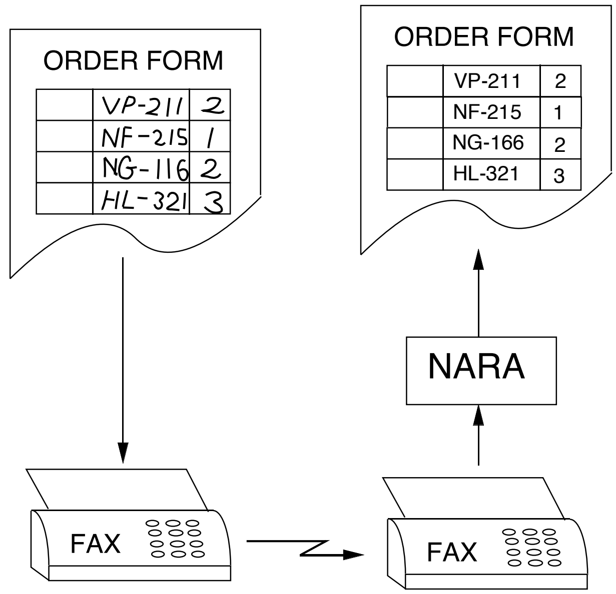


Figure 9: FAX OCR: hand-written character recognition system for a FAX ordering system.

4.4 The other combinations

There are many consumer products that use both NN and FS in several combinations, such as the independent use of FS and NN, correcting output mechanisms, and cascade combination [69], besides an NN development tool for FS previously mentioned in section 4.1.

(a) Independent use of an FS and NN is the case when one equipment uses FS and NN for different purposes, and they do not cooperate each other. For example, air conditioners of Matsushita Electric use an FS to control a compressor not to freeze in winter and use an NN to estimate PMV (Predictive Mean Vote) that is an index of comfort from six kinds of sensor information and is defined in ISO–7730 of the International Standard Organization [6].

(b) Correcting output mechanisms means that the output of an FS is corrected by the output of an NN. Although an opposite combination is possible, only the described order was realized in commercial products, so far. Washing machines of Hitachi, Sanyo, and Toshiba use this model.

Hitachi first made washing machines that use only an FS to determine the washing control parameters, e.g. washing time, water amount, and others. In their next version, they added a new sensor to increase the washing function that changes washing control according to a level of dirtiness. Here, they had two choices: making a new FS from beginning or combining the old FS for old sensors and a new system for the newly added sensor. To reduce development cost, they chose the latter approach and adopted an NN as the new system that dealt with the output of the new sensor. Oven ranges of Sanyo use the same combination system, also (see Figure 10.)

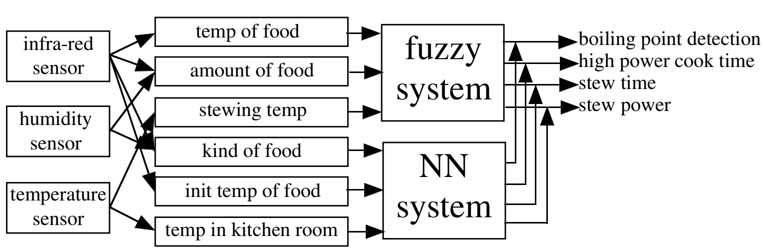


Figure 10: Sanyo's oven range whose NN fine-tunes the output of its FS from different view point.

The idea of this model includes something essential to solve complex tasks. Even if we cannot describe the detail of the complex tasks, sometimes we can describe the outline or skeleton of the tasks. We should use this a priori knowledge to increase the performance of the system solving the task. The skeletal knowledge can be described by fuzzy rules, and the remaining implicit parts can be solved by the learning function of NNs. In this case, since the essential parts are explicitly described and fixed in fuzzy rules, the NN finds a solution under the constraint described by the fuzzy rules. This idea is important to maintain safety when a user-trainable function is provided to consumers. Explicit logic maintains safety; under this control of the logic, NN changes the characteristics of products according to user's preference or lifestyle.

(c) Cascade combination is a combination system in which the output of first system of an FS or NN becomes the input to the second system of an NN or FS, representatively.

Electric fans of Sanyo use this system to detect where their users are physically located. This electric fan determines the angle to the location of its remote controller with three infrared sensors and changes the center direction of rotation of its neck; the user can always stay in the center of rotation of the fan regardless of his or her location (see Figure 11.)

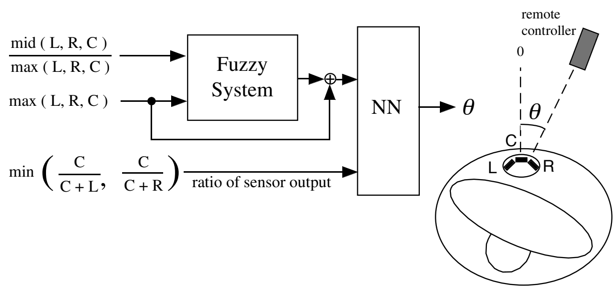


Figure 11: Sanyo's electric fan. L, R, and C are three infrared sensors in the pedestal of an electric fan. An FS estimates the distance between a remote controller and the pedestal, and an NN estimate angle between center front and the remote controller using the estimated distance and the three sensor outputs.

The difficulty of the angle estimation is that the sensor outputs depend not only angle, but also distance between sensors and the remote controller. This was why the statistical approach resulted $\pm 10°$ of estimated error in their first trial. The Sanyo's system first estimates the distance using an FS by inputting the outputs of the three sensors, then, an NN estimates the angle using the estimated distance and same sensor information. The final estimation error of products became $\pm 4°$.

Oven ranges of Toshiba use same combination. An NN first estimates initial temperature and number of pieces of bread from sensor information. Then, an FS inputs the outputs of the NN and other sensor information and determines the optimum cooking time and power.

Figure 12 shows another example of consumer products that combine an NN and FS sequentially. The room characteristics, such as wooden or concrete room, influences heating time. An NN estimate the room characteristics, and an FS determines the final control values using both NN output and sensing data.

There are also industrial applications for this cascade combination model. Toshiba applied this model to recover chemicals at a pulp factory [51]. The pulp industry uses expensive chemicals to make paper from chips. Liquid waste obtained in the final process mainly includes combustible organic components included in the chips and caustic potash soda. The purpose is to deoxidize the chemical and recover sulfureted sodium by burning the liquid waste.

The task is to control the temperature of the liquid waste and the amount of air, or oxygen, that are sent to a boiler to effectively recover the sulfureted sodium. The heated liquid waste is sprayed on a pile made from the contents of the liquid waste in the recovery boiler. As the pile burns and the supplied air is controlled, the chemicals deoxidizes when it passes through the pile. Thus, the sulfureted sodium is recovered. The size and shape of the pile determines the deoxidization time

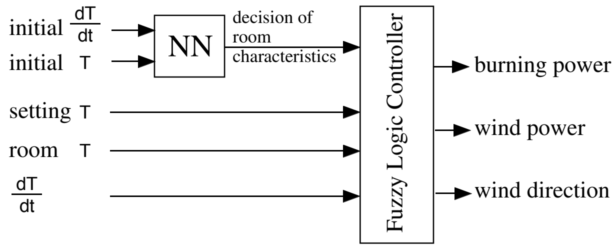


Figure 12: Samsung's fan heater. An NN estimates room characteristics, and FS inputs the decision of the NN as well as sensing values to finally decides control values.

and have an effect on the performance of recovering chemicals. It is necessary to monitor the shape of the pile and maintain a suitable shape by controlling temperature and air.

An NN is used to recognized the shape of the pile from its profile detected by CCD image and image processing. An FS is used to determine the control parameters for PID control by using sensing data from the recovery boiler and the pattern shape is recognized by the NN (see Figure 13).

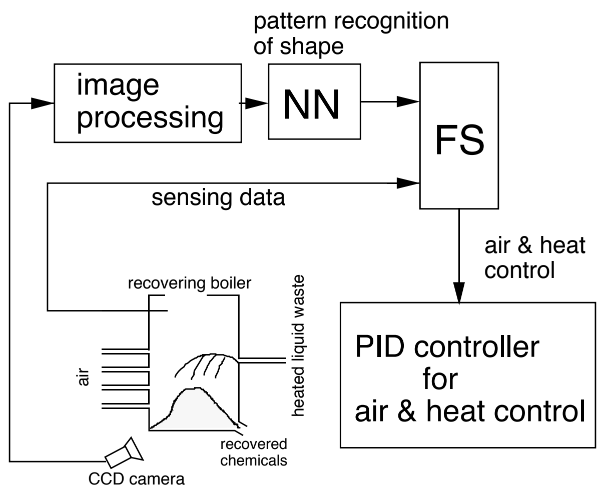


Figure 13: Chemicals recycling system at a pulp factory. NN identifies the shape of chemical pile from edge image, and an FS decides control values for air and heat control to recover chemicals effectively.

4.5 NN Learning and Configuration Based on GA

One important trend in consumer electric is to realize the function that adapts to the user environment or preference and customizes mass-produced products on user's side. NN learning function is a leading technology for this purpose. It has been applied in Japan to (1) kerosene fan heaters that learn and estimate when their owners use the heater during the day and (2) refrigerators that learn when their owners frequently open and close the door and pre-cool frozen food before they start opening or closing the refrigerator door.

LG Electric realized a user-trainable NN trained by GA for air conditioners [61]. RCE network [54] in their air conditioners inputs room temperature, outdoor temperature, time, and user-set temperature, and outputs control value to keep the user-set temperature. Suppose a user indicates that he or she wishes to change the control to adapt his or her preference. Then, GA changes the characteristics of the NN by changing number of neurons and weights (see Figure 14).

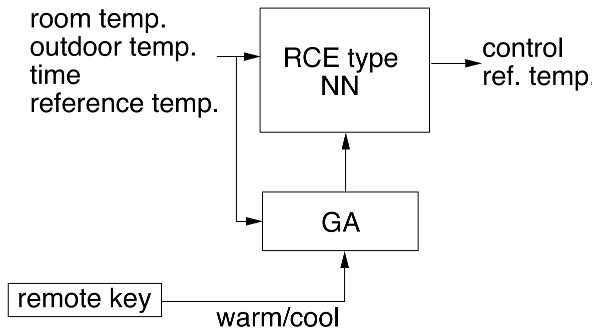


Figure 14: Temperature control by a RCE neural network controller designed by GA on the user's side.

4.6 NN-Based Fitness Function for GA

The strategy of a GA search is multi-point searching, and the GA searches for the optimum point based on the individuals. All individuals are applied to an application task and evaluated by a fitness function for the search in next generation. However, this method cannot be used for on-line processes. Once an individual is applied to the process, the condition of the process has been changed. So, the best individual, i.e. the best on-line control value, must be selected without applying individuals to the actual process. This is why GA is difficult to use for on-line processes.

One solution is to simulate the given process and embed it into a fitness function (See Figure 15). Suppose we wish to apply GA to a hydroponics system. Generally, it is very difficult to mathematically model plant growth. However, as it is easy to observe the input/output data of the given process, an NN may be a good simulator.

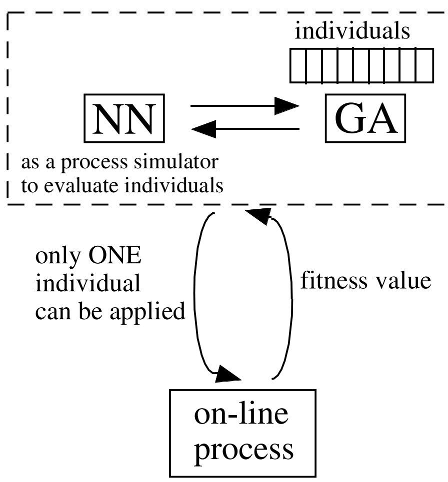


Figure 15: GA with an NN fitness function that emulates the on-line process.

GA whose fitness function uses an NN as a simulator of plant growth was used for a hydroponics system [42] (see Figure 16). The hydroponics system controls the temporal pattern of water drainage and supply to the target plant to maximize its photosynthetic rate. The simulation NN is trained using the temporal pattern as input data and CO_2 as output data. The CO_2 is a proxy for the photosynthetic rate of the plant. Temporal patterns of water drainage and supply that are generated by the GA are applied to the pseudo-plant, the trained NN, and evaluated how much CO_2 is created. The best temporal pattern is selected by the simulation and applied to the actual plant.

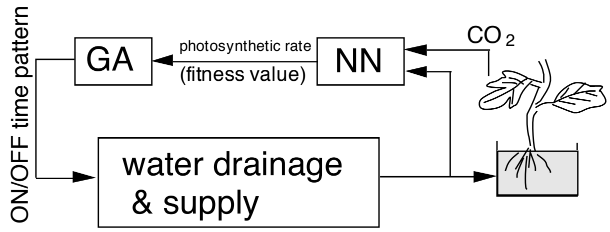


Figure 16: Water control for a hydroponics system.

A similar approach is used for ethyl alcohol production [30]. Bioprocess takes the plant's place of the aforementioned hydroponics system. Recurrent type of NN learns the fermentation characteristics, and GA decides the optimum control parameters to maximize the alcohol fermentation.

Although an industrial application has not yet been developed, computerized color recipe prediction system using GA and NNs [41] is an actual application expected soon. The NNs in this model are used as parts of a fitness function to increase the performance of GA.

The color recipe prediction system inputs the spectrum of given color and estimates the mixture rate of color pigments. Mizutani et al. applied GA to estimate the ratios. To improve the precision of its estimate, they constructed a complex fitness function consisting two NNs and a rule-based system [41].

They actually made paint with the estimated mixture ratios and evaluated the performance of their system. The average color difference between given color and the created paint colors that conventional NN model and their GA+NN model made were 2.98 and 0.71, respectively. Since it is considered that the human threshold level of distinguishing color difference is 0.7, it can be said that the system performance approaches a professional level.

4.7 GA Whose Performance Is Controlled by Fuzzy Rule Base

A dynamic parametric GA in Figure 18 that fuzzy rule base changes GA parameters adaptively was proposed in 1993 [37], and some papers followed. However, the research on this application has been few. There is no report of this type application in industry, so far except lab level research.

5 Fundamental patents

Since we have not surveyed NN+FS patents, a review of all of the patents cannot be shown in this paper. However, we examine the claims of several important NN+FS patents applied just before and since our first presentation of NN-driven fuzzy reasoning in 1988.

The first patent claims any FS whose antecedent and/or consequent parts consist of an NN [83, 85]. Five of the models, (b), (e), (f), (g), and (h) among eight NN+FS models in section 3.1 are of this type. Well-used ANFIS in Matlab fuzzy toolbox is the model (e). There may be discussion as to whether this model, (b), is covered by this patent. As the NN outputs correct the FS outputs of the model (b), it is equivalent to the FS whose consequent part includes the NN. This is why the model (b) is within the technical philosophy of this patent.

The second patent claims any FS that is designed using NNs [84, 85]. The model (d) in section 3.1 corresponds to this claim. The technical philosophy of this patent is to automatically design FSs with NNs, and embedding the NNs in the final system is not considered important. Since the NN+FS model that has the biggest contribution to business is the model (d), this patent is very practical.

The third patent claims the model of NN-driven fuzzy reasoning [84, 86] and is included in the first patent.

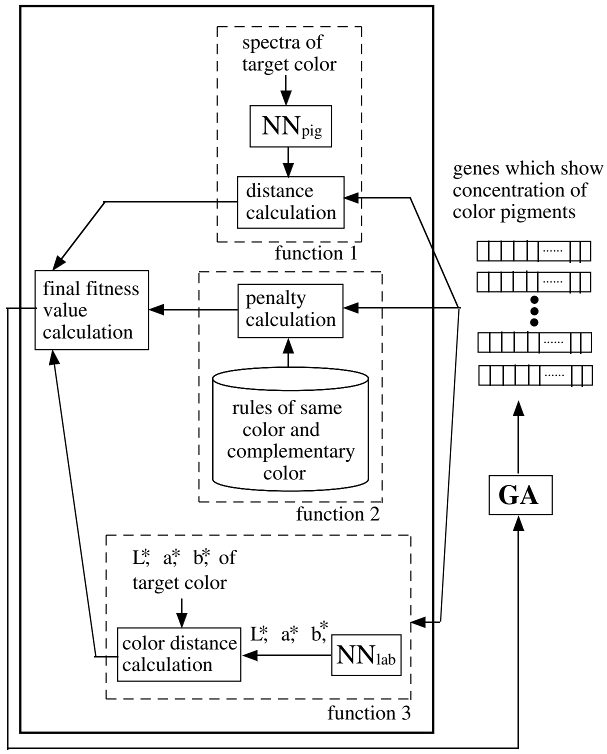


Figure 17: GA fitness function consisting two NNs and one rule base for a computerized color recipe prediction system.

The first and second patents were registered in Japan and in the US, and are in the estimation phase in Europe (UK, France, and Germany). The third patent was registered in Japan, the US, the EC, and China.

6 Conclusion

In this paper, we introduce the exciting conditions surrounding R&D on NN+FS, especially during the early stage. NN+FS research rapidly spread from Japan to the rest of the world, and NN+FS technology has since been widely used in commercial products and industrial systems. Once considered a commercial advantage, this technology has become a mainstay feature in product development. The implementation of this simple use of NN+FS is so common that it is no longer considered newsworthy anymore. Today, interest to widen cooperative technology in Soft Computing is much stronger than in NN+FS, and is used in everyday products. When a new technology that replaces NN or FS is developed, a new fervor, much like the one about NN+FS research during the 1990s, may return and further improve the performance of commonplace products and industrial systems. As researchers, we are expected to develop such technologies that will make a mark in the history of computational intelligence.

References

[1] Amano, A., Aritsuka, T., Hataoka, N., and Ichikawa, A. (1989): *A study on application of neural networks and fuzzy logic to Consonant Recognition Based on Pair Discrimination Rules*, Trans. of IEICE Part D-II, vol.J72-D-II, no.8, pp.1200–1206 (in Japanese).

[2] Arabshahi, P., Choi, J. J., Marks, R. J., and Caudell, T. P. (March, 1992): *Fuzzy control of backpropagation*, IEEE Int. Conf. on Fuzzy Systems, pp.967–972, San Diego, CA, USA.

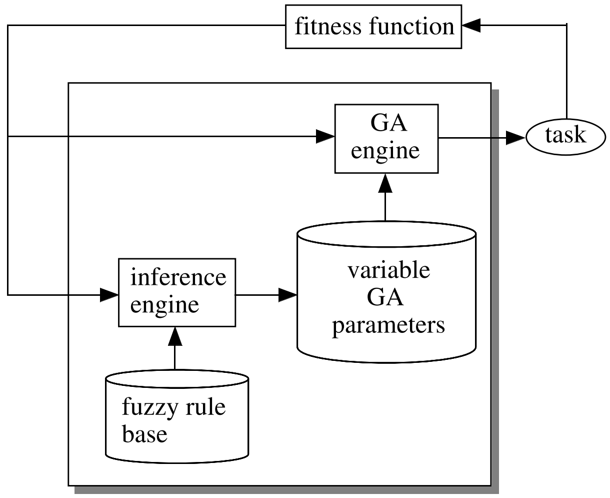


Figure 18: FS observes the change of fitness values and dynamically changes GA parameters such as population size, crossover rate, and mutation rate.

[3] Berenji, H.R. and Khedkar, P. (1992): *Learning and tuning fuzzy logic controllers through reinforcements*, IEEE Trans. on Neural Networks, vol.3, no.5, pp.724–740.

[4] Butnariu, D. (August, 1977): *L-fuzzy automata. Description of a neural model*, Proc. Int. Congress on Cybernetics Systems, vol.3rd, no.2, Bucharest, Romania, pp.119–124.

[5] Chorayan, O.G. (1982): *Identifying elements of the probabilistic neuronal ensembles from the standpoint of fuzzy sets theory*, Fuzzy Sets & Systems, vol.8, no.2, pp.141–147.

[6] Fanger, P.O. (1970): *Thermal Comfort — Analysis and Application in Environmental Engineering*, New York: McGraw-Hill.

[7] Fukuda, T., Ito, S., and Arai, F.. (1993): *Recognition of human face using fuzzy interface and neural network*, Trans. of Japan Society of Mechanical Engineers, vol.59, no.558, Series C, pp.508–514 (in Japanese.)

[8] Furuhashi, T. (1993): *Representation of fuzzy rules using neural networks*, J. of Japan Society for Fuzzy Theory and Systems, vol.5, no.2, pp.204–217 (in Japanese).

[9] Furuya, T., Kokubu, A. and Sakamoto, T. (1988): *NFS: Neuro fuzzy inference system*, Int. Workshop on Fuzzy System Applications (IIZUKA-88), Iizuka, Fukuoka, Japan, pp.219–230.

[10] Furuya, T., Kokubu, A. and Sakamoto, T. (1989): *NFS: Neuro fuzzy inference system*, Trans. of Information Processing Society of Japan, vol.30, no.6, pp.795–798 (in Japanese).

[11] Halgamuge, S.K. and Glesner, M. (1994): *Neural networks in designing fuzzy systems for real world applications*, Fuzzy Sets and Systems, vol.65, no.1, pp.1–12.

[12] Halgamuge, S. K., Mari, A. and Glesner, M. (1994): *Fast perceptron learning by fuzzy controlled dynamic adaptation of neural parameters*, In R. Kruse et al. (Eds.), *Fuzzy Systems in Computer Science"* Ch. 3.4, Wiesbaden. Germany: Verlag Vieweg, pp.129–140.

[13] Hayashi, I., Naito, E., and Wakami, N. (Nov., 1991): *A proposal of fuzzy connective with learning function and its application to fuzzy information retrieval*, Int. Fuzzy Engineering Symposium (IFES'91), pp.446–455 Yokohama, Kanagawa, Japan.

[14] Hayashi, I. and Takagi, H. (May, 1988) *Formulation of fuzzy reasoning by neural network*, 4th Fuzzy System Symposium of IFSA Japan, Tokyo, Japan, pp.55–60 (in Japanese).

[15] Hayash, I. and Umano, M. (1993): *Perspectives and trends of fuzzy-neural networks*, J. of Japan Society for Fuzzy Theory and Systems, vol.5, no.2, pp.178–190 (in Japanese).

[16] Hayashi, Y. and Nakai, M. (1990): *Automated extraction of fuzzy IF-THEN rules using neural networks*, Trans. of IEE Japan, vol.110-C, no.3, pp.198–206 (in Japanese).

[17] Hayashi, Y., Buckley, J.J., and Czogala, E. (July, 1992): *Direct fuzzyfication of neural network and fuzzyfied delta rule*, 2nd Int. Conf. on Fuzzy Logic & Neural Networks (IIZUKA'92), pp.73–76, Iizuka, Fukuoka, Japan.

[18] Hirota, K., Yamauchi, K. and Murakami, J. (1992): *Image recognition based on fuzzy·neuro technology by using binocular stereoscopic vision*, 8th Fuzzy System Symposium, pp.657–660 (in Japanese).

[19] Horikawa, A., Furuhashi, T., Okuma, S., and Uchikawa, Y. (Nov., 1990): *Composition methods of fuzzy neural networks*, 16th Annual Conf. of IEEE Industrial Electronics Society, vol.2, pp.1253–1258, Pacific Grove, CA, USA.

[20] Horikawa, S., Furuhashi, T., Okuma, S., and Uchikawa, Y. (1991): *A Learning Fuzzy Controller Using a Neural Network*, Trans. of the SICE, vol.27, no.2, pp.208–215 (in Japanese).

[21] Horikawa, S., Furuhashi, T. and Uchikawa, Y. (1992): *Composition methods and learning algorithms of fuzzy neural networks*, J. of Japan Society for Fuzzy Theory and Systems, vol.4, no.5, pp.906–928 (in Japanese).

[22] Ichihashi, H. and Watanabe, T. (1990): *Learning control by fuzzy models using a simplified fuzzy reasoning*, J. of Japan Society for Fuzzy Theory and Systems, vol.2, no.3, pp.429–437 (in Japanese).

[23] Imasaki, N., Kiji, J. and Endo, T. (1992): *A fuzzy rule structured neural networks*, J. of Japan Society for Fuzzy Theory and Systems, vol.4, no.5, pp.985–995 (in Japanese).

[24] Ishibuchi, H. and Tanaka, H. (1989): *Fuzzy regression analysis on fuzzy group: linear programming approach and neural networks approach*, SICE Kansai branch Symposium, Osaka, Jaoan, pp.13–18 (in Japanese).

[25] Ishibuchi, H. and Tanaka, H. (1992): *Fuzzy regression analysis using neural networks and its application*, Trans. of the SICE, vol.28, no.1, pp.116–124 (in Japanese).

[26] Ishibuchi, H., Fujioka, R., and Tanaka, H. (1992): *Classification of fuzzy vectors by neural networks*, Trans. of the Inst. of Systems, Control and Information Eng., vol.5, no.5, pp.198–206 (in Japanese).

[27] Ishibuchi, H., Okada, H. and Tanaka, H. (1992): *Learning of neural networks with fuzzy weights*, 8th Fuzzy System Symposium, Hiroshima, Japan, pp.185–188 (in Japanese).

[28] Izumida, M., Murakami, K. and Aibara, T. (1988): *Fuzzy clustering on neural network*, IEICE Techn. Rept. PRU88-71, vol.88, no.279, pp.1–8 (in Japanese).

[29] Jang, J-S. R. (1993): *Self-learning fuzzy controllers based on temporal back propagation*, IEEE Trans. on Neural Networks, vol.4, no.1 pp.156–159.

[30] Karim, M. and Rivera, S.L. (June, 1992): *Use of recurrent neural networks for bioprocess identification in on-line optimization by micro-genetic algorithms*, 1992 American Control Conference, pp.1931–1932, Chicago, IL, USA.

[31] Karr, C.L., Meredith, D.L., and Stanley, D.A., (March, 1989): *Control of liquid level via learning classifier system*, the SPIE - The Int. Society for Optical Engineering, vol.1095, pt.1, pp.78–85.

[32] Kawamura., A., Watanabe, N., Okada, H., and Asakawa, K. (1992): *A prototype of neuro-fuzzy cooperation system*, IEEE Int. Conf. on Fuzzy Systems (FUZZ-IEEE'92), pp.1275–1282, San Diego, CA, USA.

[33] Kim, J.W., Ahn, L.S., and Yi, Y.S. (1995): *Industrial applications of intelligent control at Samsung Electric Co., – in the home appliance devision –*, In W. Chiang and J. Lee (Eds.) *Fuzzy Logic for the Applications to Complex Systems*, Taipei, Taiwan: Singepore: World Scientific Publishing, pp.478–482.

[34] Kosko, B. (1986): *Fuzzy cognitive maps*, Int. J. Man-Machine Studies, vol.24, no.1, pp.65–75.

[35] Kosko, B. (1991): *Neural networks and Fuzzy systems*, NJ, USA: Prentice Hall.

[36] Lee, A.M and Takagi, H. (March, 1993): *Integrating design stage of fuzzy systems using genetic algorithms*, IEEE 2nd Int. Conf. Fuzzy Systems (FUZZ-IEEE'93), vol.1, pp.1293–1296, San Francisco, CA.

[37] Lee, A.M. and Takagi, H. (1993): *Dynamic control of genetic algorithms using fuzzy logic techniques*, 5th Int. Conf. on Genetic Algorithms (ICGA'93), Urbana-Champainm IL, USA, pp.76–83.

[38] Lee, S.C. and Lee, E.T. (1974): *Fuzzy sets and neural networks*, J. Cybern, vol.4, no.2, pp.83–103.

[39] Lee, S.C. and Lee, E.T. (1975): *Fuzzy neural networks*, Mathematical Biosciences, vol.23, no.1-2, pp.151–177.

[40] Maeda, M. and Murakami, S. (1987): *A fuzzy decision making method for multi-objective problems with a preference structure representation by fuzzy connectives*, Trans. of SICE, vol.23, no.5, pp.517–524 (in Japanese).

[41] Mizutani, E., Takagi, H., and Auslander, M. (April, 1995): *A cooperative system based on soft computing methods to realize higher precision of computer color recipe prediction*, Proc. of SPIE - The Int. Society for Optical Engineering, vol.2492, Pt.1, pp.303–314.

[42] Morimoto, T., Takeuchi, T., and Hashimoto, Y. (Oct. 1993): *Growth optimization of plant by means of the hybrid system of genetic algorithm and neural network*, Int. Joint Conf. on Neural Networks (IJCNN-93-Nagoya), vol.3, pp.2979–2982, Nagoya, Japan.

[43] Morita, A., Imai, Y. and Takegaki, M. (1988): *A method to refine fuzzy knowledge model of neural network type*, SICE'88 27th meeting, Chiba, Japan, JS33-3, pp.347–348 (in Japanese).

[44] Nakajima, M., Okada, T., Hattori, S., and Morooka, Y. (1993): *Application of pattern recognition and control technique to shape control of the rolling mill*, Hitachi Review, vol.75, no.2, pp.9–12 (in Japanese).

[45] Nauck, D. and Kruse, R. (Mar./April, 1993): *A fuzzy neural network learning fuzzy control rules and membership functions by fuzzy error backpropagation*, 1993 IEEE Int. Conf. on Neural Networks, vol.2. pp.1022–1027, San Francisco, CA, USA.

[46] Nishio C. and Nakanishi, S. (1992): *Sales promotion planning support system by neural networks*, 8th Fuzzy System Symposium, Hiroshima, Japan, pp.153–156 (in Japanese).

[47] Nomura, H., Hayashi, I. and Wakami, N. (1989): *Self-tuning method of fuzzy reasoning by Hopfield neural network*, 5th Fuzzy System Symposium, Kobe, Japan, pp.177–182 (in Japanese).

[48] Nomura, H., Hayashi. I., and Wakami, N. (1992): *A self-tuning method of fuzzy reasoning by delta rule and its application to a moving obstacle avoidance*, J. of Japan Society for Fuzzy Theory and Systems, vol.4, no.2, pp.379–388 (in Japanese).

[49] Okamoto, T., Takagi, T. and Nakanishi, S. (1991) : *Time series modeling and prediction by neural networks and fuzzy set theory*, 7th Fuzzy System Symposium, Nagoya, Japan, pp.459–462 (in Japanese).

[50] Ogawa, Y., Morita, T., and Kobayashi, K. (1991): *A fuzzy document retrieval system using the keyword connection matrix and a learning method*, Fuzzy Sets and Systems, vol.39, no.2, pp.163–179.

[51] Ozaki, T. (1994): *Recovering boiler control at a pulp factory*, Neuro-Fuzzy-AI Handbook, Tokyo, Japan: Ohm Publisher, pp.1209–1217 (in Japanese).

[52] Ozawa, J., Hayashi, I. and Wakami, N. (1991): *A proposal of examination method of input-output data using CMAC-algorithm for fuzzy modeling*, 7th Fuzzy System Symposium, Nagoya, Japan, pp.435–438 (in Japanese).

[53] Ozawa, J., Hayashi, I., and Wakami, N. (March, 1992): *Formulation of CMAC-fuzzy system*, IEEE Int. Conf. on Fuzzy Systems (FUZZ-IEEE'92), pp.1179–1186, San Diego, CA, USA.

[54] Reilly, D.L., Cooper, L.N., and Elbaum, C. (1982): *A neural model for category learning*, Biological Cybernetics, vol.45, no.1, pp.35–41.

[55] Rocha, A.F., Francozo, E., Handler, M.I., and Balduino, M.A. (1980): *Neural languages*, Fuzzy Sets & Systems, vol.3, no.1, pp.11–35.

[56] Rocha, A.F. (1981): *Neural fuzzy point processes*, Fuzzy Sets & Systems, vol.5, no.2, pp.127–140.

[57] Rocha, A.F. (1982): *Basic properties of neural circuits*, Fuzzy Sets & Systems, vol.7, no.2, pp.109–121.

[58] Rocha, A.F., and Bassani, J.W.M. (1982): *Information theory applied to the study of neural codes*, General survey of systems methodology. Proc. 26th annual meet. soc. gener. systems reasearch with the American association for advanc. science, Washington D.C., vol.2, pp.528–533.

[59] Sakawa, M., Sawada, K. and Douki, Y. (1992): *Fuzzy multiobjective combinatorial optimization through neural networks*, 8th Fuzzy System Symposium, Hiroshima, Japan, pp.145–148 (in Japanese).

[60] Sawaragi, T., Shibata, K., Katai, O., Iwai, S., and Tsukada, K. (March, 1992): *Integrated cognitive architecture for image understanding using fuzzy clustering and structured neural network*, IEEE Int'l Conf. on Fuzzy Systems San Diego, CA, USA, pp.21–28.

[61] Shin, M.S., Lee, K.L., Lim, T.L., and Wang, B.H. (1995): *Applications of evolutionary computations at LG Electrics*, in *Fuzzy Logic for the Applications to Complex Systems,* ed. by W. Chiang and J. Lee, World Scientific Publishing, pp.483–488.

[62] Shiue, L.C., and Grondin, R.O. (1987): *On designing fuzzy learning neural-automata*, IEEE Int'l Conf. on Neural Networks (ICNN'87), San Diego, CA, USA, vol.2, pp.299–307.

[63] Takagi, H. and Hayashi, I. (Aug., 1988): *Artificial_neural_network-driven fuzzy reasoning*, Int. Workshop on Fuzzy System Applications (IIZUKA'88), pp.217–218, IIZUKA, Japan.

[64] Takagi, H. (1990): *Ambiguity and knowledge information processing*, Proc. of System, Control and Information, vol.34, no.5, pp.263–273 (in Japanese).

[65] Takagi, H. (July, 1990): *Fusion technology of fuzzy theory and neural networks — Survey and future directions —*, Int. Conf. on Fuzzy Logic & Neural Networks (IIZUKA'90), vol.1, pp.13–26, Iizuka, Japan.

[66] Takagi, H. and Hayashi, I. (1991): *NN-driven fuzzy reasoning*, Int. J. of Approximate Reasoning (Special Issues of IIZUKA'88), vol.5, no.3, pp.191–212.

[67] Takagi, H., Kouda, T., and Kojima, Y. (1991): *Neural-networks designed on approximate reasoning architecture*, J. of SOFT, vol.3, no.1, pp.133–141 (in Japanese).

[68] Takagi, H., Suzuki, N., Kouda, T., Kojima, Y. (1992): *Neural networks designed on approximate reasoning architecture*, IEEE Trans. on Neural Networks, vol. 3, no. 5, pp.752–760.

[69] Takagi, H. (1995): *Survey of fuzzy logic applications in image processing equipment*, In J. Yen et a;. (Eds.) *Industrial Applications of Fuzzy Control and Intelligent Systems*, Ch.4, pp.69–92, IEEE Press, Piscataway, NJ, USA.

[70] Takahashi, H. and Minami, H. (1989): *Subjective evaluation modeling using fuzzy logic and a neural networks*, 3rd IFSA Congress. pp.520–523 Seattle, WA, USA.

[71] Takahashi, H. (1993): *Features and applications of "The fuzzy-neuro system"*, J. of Japan Society for Fuzzy Theory and Systems, vol.5, no.2, pp.233–244 (in Japanese).

[72] Tan, Y. and Ejima, T. (1989): *Fuzzy partition model: A network with multipartitioning units and its basic properties*, IEICE Techn. Rept. PRU89-45, pp.39–46 (in Japanese).

[73] Toyota, A (1992): *News flush*, J. of Inst. of Elect., Info., and Commun. Engineers, vol.72, no.7, p.812 (in Japanese).

[74] Uehara, K. and Fujise, M. (July, 1990): *Learning of fuzzy-inference criteria with artificial neural network*, Int. Conf. on Fuzzy Logic & Neural Networks (IIZUKA'90), vol.1, pp.193–198, Iizuka, Fukuoka, Japan.

[75] Wang, L.-X. and Mendel J.M. (March, 1992): *Back-propagation fuzzy system as nonlinear dynamic system identifier*, IEEE Int. Conf. on Fuzzy Systems, pp.1409–1418, San Diego, CA, USA.

[76] Wang, P-Z. and Zhang, D. (1992): *The netlike inference process and stability analysis*, Int. J. of Intelligent Systems, vol.7, no.4, pp.361–372.

[77] Watanabe, N., Kawamura, A., Masuokam R., Owada, Y., and Asakawa, K. (1990): *Fuzzy control using neural network*, 40th of meeting IPSJ, Tokyo, Japan, 3C-1 0, pp.148–149 (in Japanese).

[78] Xu, H. Y., Wang, G. Z. and Baird, C. B. (June, 1992): *A fuzzy neural networks technique with fast back propagation learning*, Int. J. Conf. on Neural Networks (IJCNN'92), Baltimore, MA, USA, pp.214–219.

[79] Yamaguchi, T., Tanabe, M. and Murakami, J. (1989): *Fuzzy control using LVQ unsupervised learning*, SICE Joint Symposium of 15th System Symposium and 10th Knowledge Engineering Symposium, Sapporo, Japan, pp.179–184 (in Japanese).

[80] Yamaguchi, T., Imasaki, N. and Haruki, K. (1990): *A reasoning and learning method for fuzzy rules with associative memory*, Trans. of IEE Japan, vol.110-C, no.3, pp.207–215 (in Japanese).

[81] Yamaguchi, T., Imasaki,N. and Haruki, K. (June, 1990): *Fuzzy rule realization on associative memory system*, IEEE and INNS Int. Joint Conf. on Neural Networks (IJCNN'90), Washington D.C., USA, vol.II, pp.720–723.

[82] Yamaguchi, T. (1993): *Fuzzy associative memory system*, J. of Japan Society for Fuzzy Theory and Systems, vol.5, no.2, pp.245–260 (in Japanese).

[83] Japan patent No. 2553725 (Aug., 1996).

[84] Japan patent No. 2813567 (Aug., 1998).

[85] US patent No. RE36,823 (Aug., 2000)

[86] US patent No. 5168548 (Dec., 1992).
US patent No. 5255344 (Oct., 1993).