Int. J. Appl. Math. Comput. Sci., 2000, Vol.10, No.4, 647–673

FUSION TECHNOLOGY OF NEURAL NETWORKS AND FUZZY SYSTEMS: A CHRONICLED PROGRESSION FROM THE LABORATORY TO OUR DAILY LIVES

HIDEYUKI TAKAGI*

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 of their development 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.

Keywords: cooperative models, neural networks, fuzzy systems, genetic algorithms, real world applications, overview

1. Introduction

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

Although the first paper that used both *NNs* and *fuzzy sets* as keywords dates back to 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 the technology, introduced at the beginning of this paper, is supported by a number of real-world applications. Since its introduction, the fusion technology has widely expanded into application fields requiring tasks such as control, operations research, retrieval, clustering, speech recognition, and many others (Hayashi and Umano, 1993). We show the spread of this trend from Japan to the US and Europe in the next section. Figure 1 shows the increasing number of papers and conferences using the keywords *NN* and *FS* during the early 1990s.

* Kyushu Institute of Design, 4–9–1, Shiobaru, Minami-ku, Fukuoka, 815–8540 Japan, e-mail: takagi@kyushu-id.ac.jp, http://www.kyushu-id.ac.jp/~takagi

which produced several papers during the summer of that year (Takagi, 1990a; 1990b). 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 for the Japanese industry which was particularly interested in the application of this technology. Japanese companies took the initiative in the R&D of NN+FS technology by proposing its models to developing its systems and products. Many seminars and symposiums on NN+FS were held between 1988 and 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 only to consumer products, but also to industrial systems such as the rolling mill and the chemical process of a pulp factory described in Section 4.4. A series of new technical terms such as *neuro-fuzzy* and *neural and fuzzy* in advertisements quickly followed those of FL-based consumer products in 1990. These terms puzzled consumers and the industrial nonchalance towards naming technologies that exceeded the consumers' understanding was criticized. Today, such names are not endorsed even if they are used. The 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 the NN+FS technology by sponsoring several seminars and publishing several tutorial 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 a conference on FL and NNs; IIZUKA'94 was a conference on FL, NNs, and soft computing; and IIZUKA'96 was a conference on soft computing.

2.3. NN+FS Research in the US

The interest in FSs 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 a social phenomenon. 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 FSs for Japanese businesses reached the US and Europe, and American and European researchers began paying 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 the Lee brothers' paper, published in 1974, Kosko introduced the fuzzy concept to NNs and the NN learning algorithm: FAM (fuzzy associate memory) that memorizes one fuzzy rule (Kosko, 1991) and a fuzzy cognitive map that extends the relational weight, {0,1}, of the cognitive map to [0,1] and introduces the Hebbian law as its learning algorithm (Kosko, 1986).

The contribution of the Berkeley Initiative in Soft Computing (BISC) Program and its missionary, L.A. Zadeh, in expanding the NN+FS research in the US was outstanding. 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 research works here and there.

The other reason why the NN+FS research increased in the US in those days was that the 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 the combination of FSs with NNs that persuaded the acceptance by American researchers and academic societies.

2.4. NN+FS Research in Europe

The wide acceptance of FL applications in Europe started 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 resulted from the conservative action that hindered the acceptance of the new technologies until their undoubted effectiveness was proven.

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, another computer journal, *MC*, featured FSs 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 that time, many companies assigned a few employees to the research on FL to diversify their operations should FS significantly impact their business. However, the skepticism about 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.

An 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 the European research could be observed by examining the EUFIT conferences. About 500 participants attended the EUFIT'93. Researchers and engineers who started the FL research needed to obtain the latest information on it 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 a 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 (Hayashi and Umano, 1993). Since it is visually easy to understand their categorization, we explain these models according to Fig. 2:

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

(c) and (d) are the case when an NN and an FS are connected in cascade. Electric fans in Section 4.4(c) adopt this model. As this connection can be applicable to the pre- or post-processing part of signal processing, this model was applied to several signal processing tasks, such as data analysis (Nishio and Nakanishi, 1992; Okomato *et al.*, 1991), image recognition (Fukuda, 1993; Hirota *et al.*, 1992; Takahashi and Minami, 1989), image understanding (Sawaragi *et al.*, 1992), speech recognition (Amano *et al.*, 1989) and many others.

(e) is the case when an FS is designed or tuned 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 (Nomura *et al.*, 1989; Ozawa *et al.*, 1991) and applying NN tuning to non-FS (Hayashi *et al.*, 1991; Maeda and Murakami, 1987; Ogawa *et al.*, 1991; Sakawa *et al.*, 1992) are categorized by this model, too.

(f) is a network expression of fuzzy reasoning (Hayashi and Nakai, 1990; Uehara and Fujise, 1992). Although most research on this model uses the term NN, some appropriately call it network reasoning; a few researchers consciously avoid the term of NN (Wang and Zhang, 1992).

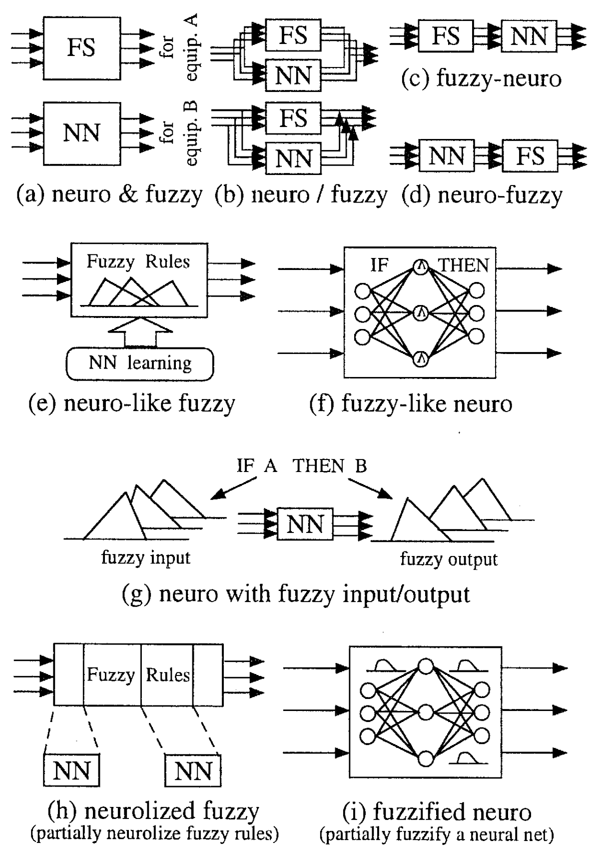


Fig. 2. Hayashi and Umano's categorization of NN+FS models (Hayashi and Umano, 1993).

(g) is the case where an NN configures fuzzy reasoning rules or conducts fuzzy reasonings. Some are a feed-forward NN handling fuzzy values (Furuya *et al.*, 1988; 1989; Horikawa *et al.*, 1990; 1991; 1992; Imasaki *et al.*, 1992; Kawamura *et al.*, 1992; Takagi *et al.*, 1991; 1992), fuzzy associative memory based on BAM (bi-directional associative memory) (Yamaguchi *et al.*, 1990a; 1990b), and CMAC type of NN dealing with fuzzy rules (Ozawa *et al.*, 1992). An NN conducting fuzzy clustering may be categorized in this type of model, too (Izumida *et al.*, 1988; Tan and Ejima, 1989).

(h) is the case in which certain parts of an FS are replaced by feed-forward NNs, such as the NN-driven fuzzy reasoning in Section 4.1, or a vector quantization NN (Yamaguchi *et al.*, 1989). A simplified model in which only the gains of membership functions are adjusted by NNs is another model (Morita *et al.*, 1988). Besides its application to fuzzy reasoning, fuzzy regressive analysis is categorized by this model (Ishibuchi and Tanaka, 1992).

(i) is the fuzzification of an NN. Modifying the NN to handle fuzzy numbers of inputs, outputs and fuzzy weights (Ishibuchi *et al.*, 1992a; 1992b), and fuzzifying the whole learning (Hayashi *et al.*, 1992) are categorized in this case.

There are several categorizations of NN+FS with different perspectives (Furuhashi, 1993; Takagi, 1990a; 1990b; Takahashi, 1993; Yamaguchi, 1993).

3.2. Why it is Easy to Fuse NNs and FSs

Both NNs and FSs realize complex non-linearities by combining and interpolating multiple basis functions. Due to these characteristics, both the technologies are effective when application tasks are non-linear and too difficult to be clearly described using mathematical equations or logic.

Membership functions correspond to basis 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 basis 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 the same strategy to realize non-linearity, but they have different historical backgrounds.

Furthermore, their basic structures are the same. Figure 3 looks like an NN. Some might even consider it as 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 Fig. 3.

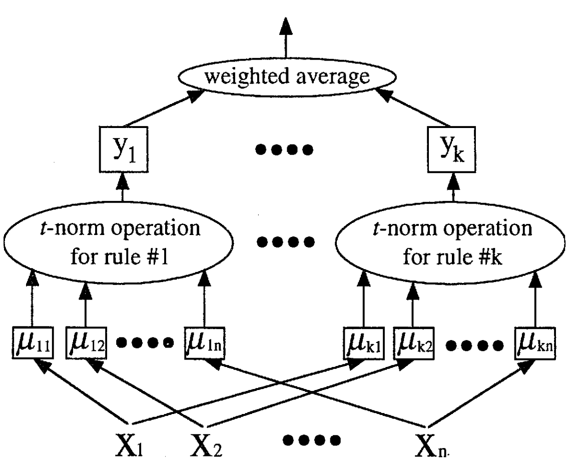


Fig. 3. Network expression of a normal FS. The FS structure is essentially similar to that of an NN.

Once we understand that the structure of Fig. 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 optimizing membership functions, μ_{ij}, and subsequent outputs, y_i, based on error minimum criteria between the outputs of a real FS and a design specification. NN+FS models from (e) to (h) of Section 3.1 are based on this idea.

The reason why NNs and FSs are fused is not only the similarity, but also the 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. When we want to use both the advantages, we choose fused NN+FS 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 necessary for the FS to borrow techniques from other technical fields to systematically design membership functions which define the qualitative parts or to adapt to a dynamic situation of application tasks. When we do not have knowledge about 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 a given task and data which keep implicit task knowledge, any single use of AI, FS, or NN technology cannot use all information. In such a case, fusing technologies that complement each other as for the capabilities of logic description and learning from data is effective.

4. Research and Development

4.1. Designing FSs Using NNs

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

A first approach to explicitly apply NNs to design an FS is NN-driven fuzzy reasoning (Hayashi and Takagi, 1988a; Takagi and Hayashi, 1988; 1991) shown in Fig. 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 nonlinear and multi-dimensional. The outputs of the NN are the rule strengths of each rule, which are combinations of membership values in antecedents.

This model was used to control a Hitachi rolling mill, and the system has been running since 1991 (Nakajima *et al.*, 1993). The purpose of the rolling mill is to flatten the metal plate by controlling 20 rolls.

The surface shape of a plate reel is detected by scanning (see Fig. 5). The scanned shape is put into an NN. The NN categorizes the surface pattern and detects a similarity between the input pattern and standard template patterns. Since fuzzy control rules are created for each standard surface pattern, the outputs of the NN determine how the input surface pattern matches each fuzzy rule, i.e. how 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.

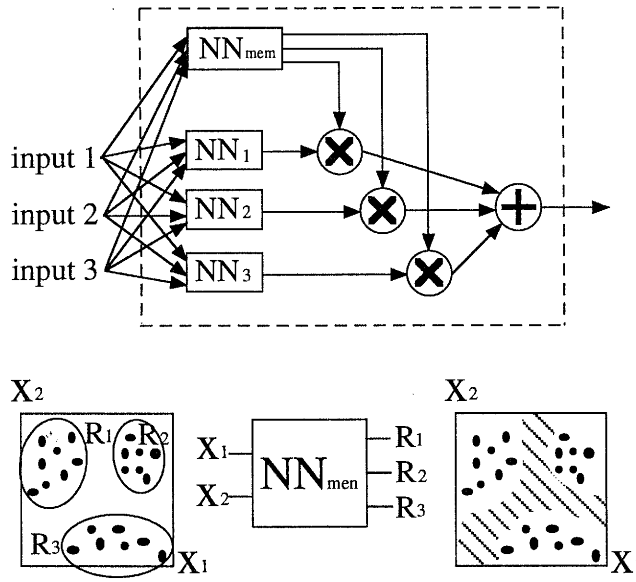


Fig. 4. NN-driven Fuzzy Reasoning. Fuzzy partitioning of the input space is performed by NN_{mem} that corresponds to antecedent parts. The figure below illustrates how to train the NN_{mem}. The training data are roughly clustered, and then NN_{mem} is trained by the input vector and cluster numbers.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

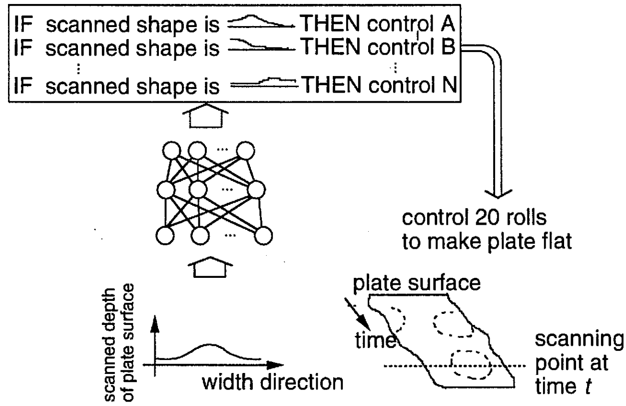


Fig. 5. Rolling mill control by an FS and an NN. The scanned pattern of the plate surface is recognized by the NN. The output value of each output unit of the NN is used as a rule strength for the corresponding fuzzy rule.

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

Furuya *et al.* (1988; 1989) proposed a Neuro Fuzzy System (NFS) that uses matching levels between an input vector and learned patterns as membership values. Yamaguchi *et al.* (1989; 1990b) proposed a method that forms membership functions using learning vector quantization (LVQ). They also proposed a learning fuzzy-NN to control the fuzziness in forming membership functions by introducing fuzzy reasoning into BAM.

Unlike these methods that design the shapes of membership functions by NNs, Morita *et al.* (1988) proposed a simplified method that fixes the shape of membership functions and tunes only their gains by an NN.

Ishibuchi and Tanaka (1989) carried out an experiment to construct various membership functions using NNs, and derived a fuzzy language by which the form of a membership function was transformed into that of an instructor, and they further derived an interval-valued membership function whose values were broadened.

Watanabe *et al.* (1990) expressed FL rules by an NN that was finely tuned through learning by allocation of one fuzzy variable to one neuron, and the weighting factor was initialized so that its sigmoid characteristics came closer to a predetermined membership function.

Another important approach since 1989 was to parameterize the shape of membership functions as in Fig. 6 and to optimize the parameters in the framework of NN learning. A triangular shape (Nomura *et al.*, 1992), a combination of sigmoidal functions (Horikawa *et al.*, 1990; 1991; 1992), a Gaussian function (Ichihashi and Watanabe, 1990; Wang and Mendel, 1992), or a bell shape (Jang, 1993) were adopted as the shapes of membership functions and the parameters of their shapes were then tuned.

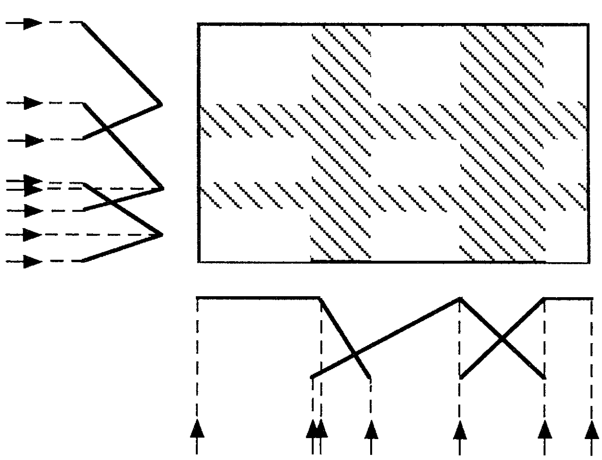


Fig. 6. NN adjusts the shape parameters of membership functions.

Sometimes we cannot obtain training data for certain applications beforehand. Fuzzy reasoning networks that allow for tuning by reinforcement learning were proposed for such a case (Berenji and Khedkar, 1992; Nauck and Kruse, 1993). Several public domain and commercial software packages based on NN+FS were announced. Some examples of free software are NFCON-I (Nauck and Kruse, 1993) and FuNeGen 1.0 based on the Fune I model (Halgamuge and Glesner, 1994).

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 the whole shapes of membership functions via 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 fans, range-hoods, and photocopiers.

Figure 7 shows an FS that was used in photocopiers (Takagi, 1995). The NN is used in the development phase, and only the tuned FS is implemented in the final product.

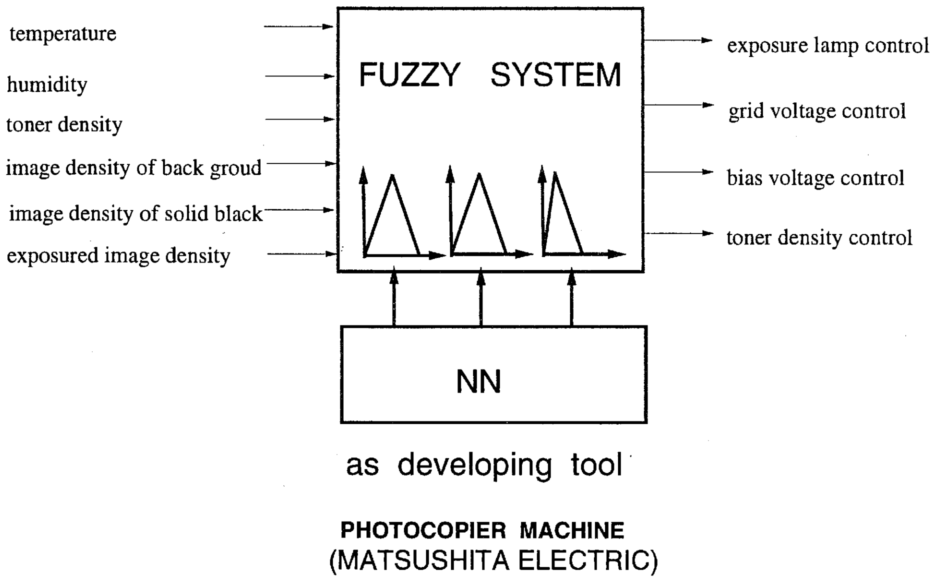


Fig. 7. An NN adjusts the parameters of the membership functions for photocopier control. Only an optimized FS is implemented into the photocopier, and the NN is used for design by the manufacturer.

4.2. Designing Fuzzy Systems Using GAs

Auto-designing FSs by NNs is equivalent to parameter optimization. We can then apply a GA instead of the NN. The pioneering work of Karr *et al.* (1989) 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 GAs (Lee and Takagi, 1993).

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

Samsung's refrigerators, introduced to the market in 1994, adopt two FSs that are designed by a GA (Kim *et al.*, 1995). The first FS estimates the temperature distribution inside the refrigerator, and the other inputs the estimated temperature distribution and determines the spout position of cool air. Both the FSs are TSK models, and the 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 (Kim *et al.*, 1995). 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 in which the parameters of the I/O membership functions are determined by a GA. The GA design is conducted in the developing phase and only designed FSs are used in products.

LG Electric has applied GAs in several products (Shin *et al.*, 1995). Their dishwashers, rice cookers, and microwave ovens use neuro-fuzzy models to estimate the number of dishes, to estimate the amount of rice, and to cook well, respectively. These NN+FS systems were also designed by GAs.

Other consumer products whose FSs were designed by GAs include refrigerators, washing machines and vacuum cleaners. GAs were used to auto-tune the fuzzy rules of these models. A GA was also used to acquire fuzzy process control rules for a plant that experiences long delays (Shin *et al.*, 1995).

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

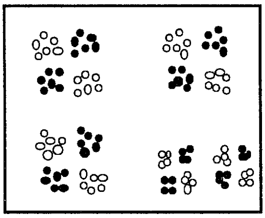
4.3. NN Learning and Configuration Based on a Fuzzy Rule Base

One way to reduce the complexity of a given task and to 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, e.g. selecting training data, setting initial weights determined by the knowledge, limiting the searching space, etc.

NARA is a structured NN that is constructed based on the IF–THEN fuzzy rule structure (Takagi *et al.*, 1992). Small sub-NNs that correspond to each consequent part and an NN that corresponds to the 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 a given task, which is obtained e.g. by analyzing the training data. Then the complexity in each fuzzy partitioned input space is much less than that of the entire task. Therefore, it becomes easier for the NARA to solve the given task.

Figure 8 shows a toy task and draws a comparison of the performance between a conventional NN and the NARA. Nevertheless, both the systems have the same number of synaptic weights, and the NARA shows a better performance because of embedding *a priori* knowledge of this task (Takagi *et al.*, 1992). Furthermore, it is easier for the NARA to be internally modified to improve its performance because of the explicit structure (Takagi *et al.*, 1992).



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

Fig. 8. Comparison of the NARA and a normal NN. The task is to classify B/W patterns in the 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. The NARA is constructed according to this rule structure.

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

This FAX ordering system is requested to have high recognition rates, because it is used for business customers. This is a 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 (Toyota, 1992). 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 the NN performance based on an FL rule base (Arabshahi *et al.*, 1992; Halgamuge *et al.*, 1994; Xu *et al.*, 1992).

4.4. The Other Combinations

There are many consumer products that use both NNs and FSs in several combinations, such as independent use of FSs and NNs, correcting output mechanisms, and a cascade combination (Takagi, 1995), besides an NN development tool for FSs mentioned previously in Section 4.1:

(a) Independent use of an FS and an NN is the case when one equipment uses them for different purposes and they do not cooperate with each other. For example,

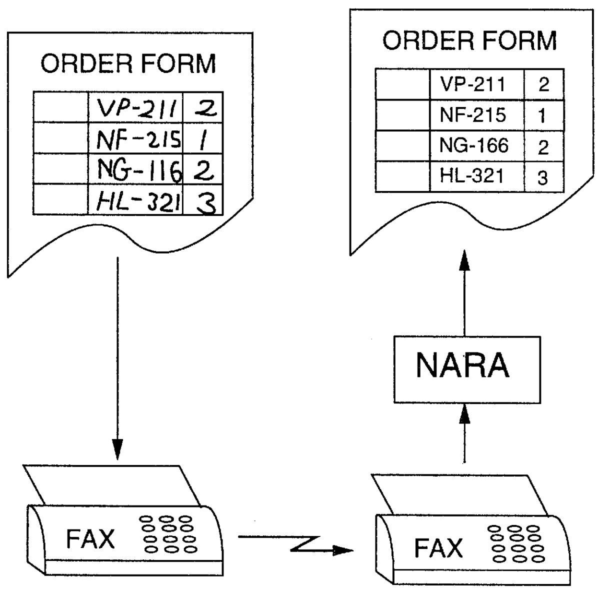


Fig. 9. FAX OCR: hand-written character recognition system for a FAX ordering system.

air conditioners of Matsushita Electric use an FS to control a compressor so as not to freeze in winter, and use an NN to estimate the 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 (Fanger, 1970).

(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 has been 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. the washing time, water amount, etc. In their next version, they added a new sensor to increase the washing function that changed washing control according to a level of dirtiness. Here, they had two choices: making a new FS from the beginning or combining the old FS for old sensors and a new system for the newly added sensor. To reduce the 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, too (see Fig. 10).

The idea of this model includes something essential to solve complex tasks. Even if we cannot describe the details of complex tasks, we can sometimes describe an outline or a 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 the fuzzy rules, the NN finds a solution subject to the constraint described by the fuzzy

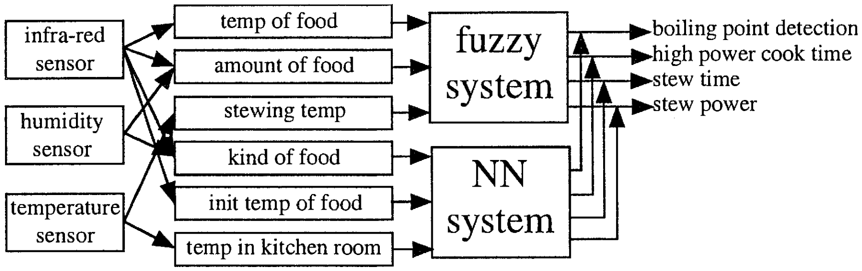


Fig. 10. Sanyo's oven range whose NN fine-tunes the output of its FS a from different view point.

rules. This idea is important to maintain the safety when a user-trainable function is provided to consumers. Explicit logic maintains the safety; under this control of the logic, the NN changes the characteristics of the products according to the user's preference or lifestyle.

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

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 Fig. 11).

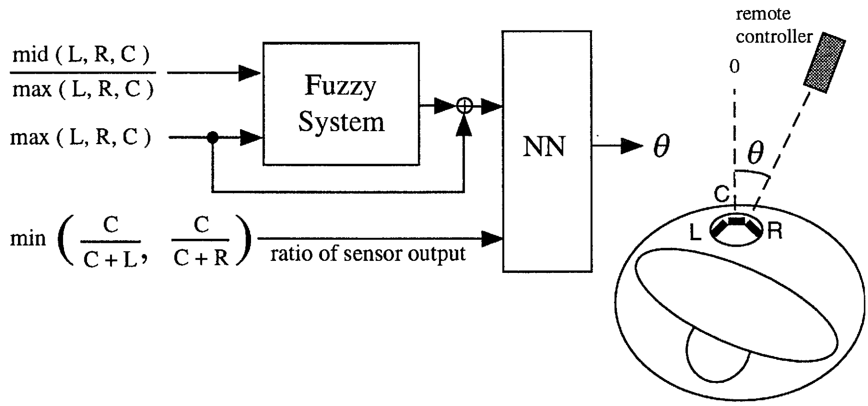


Fig. 11. Sanyo's electric fan. L, R, and C are three infrared sensors in the pedestal of the fan. An FS estimates the distance between the remote controller and the pedestal, and an NN estimates the angle between the center front and the remote controller using the estimated distance and the three sensor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

The difficulty of the angle estimation is that the sensor outputs depend not only on the angle, but also on the distance between the sensors and the remote controller. That was why the statistical approach resulted in $\pm 10°$ of the estimated error in their first trial. 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 the same sensor information. The final estimation error of the products becomes $\pm 4°$.

Oven ranges of Toshiba use the same combination. An NN first estimates the initial temperature and the 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 a wooden or a concrete room, influences the heating time. An NN estimates the room characteristics, and an FS determines the final control values using both the NN output and sensing data.

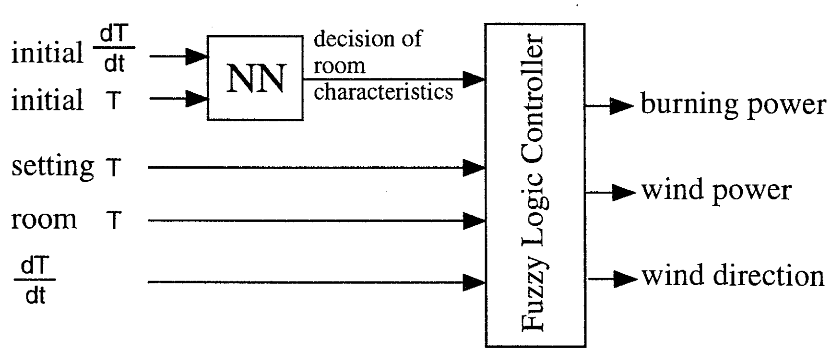


Fig. 12. Samsung's fan heater. An NN estimates the room characteristics, and an FS inputs the decision of the NN as well as sensing values so as to finally select control values.

There are also industrial applications for this cascade combination model. Toshiba applied this model to recover chemicals at a pulp factory (Ozaki, 1994). 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 to recover sulfuratled 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 sulfuratled 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 sulfuratled sodium is recovered. The size and shape of the pile determines the deoxidization time and have an effect on the performance of recovering chemicals. It is necessary to monitor the shape of the pile and to maintain a suitable shape by controlling the temperature and air.

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

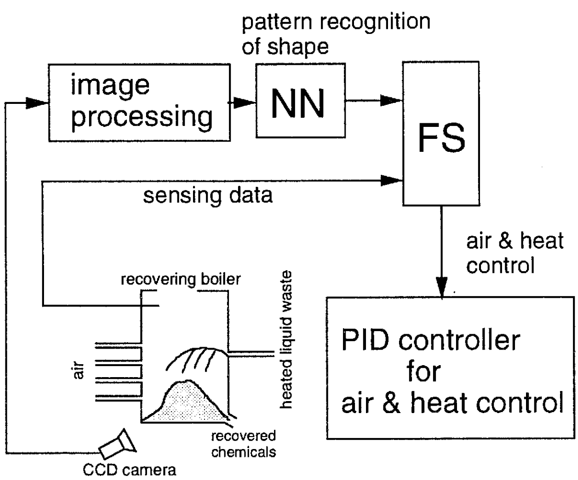


Fig. 13. Chemicals recycling system at a pulp factory. The NN identifies the shape of a chemical pile from an edge image, and the FS decides on control values for air and heat control to recover chemicals effectively.

4.5. NN Learning and Configuration Based on GAs

One important trend in consumer electric engineering is to realize a function that adapts to the user environment or preference and customizes mass-produced products on the user's side. The NN learning function is a leading technology for this purpose. It has been applied in Japan to kerosene fan heaters that learn and estimate when their owners use the heater during the day and in 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 a GA for air conditioners (Shin *et al.*, 1995). The RCE network (Reilly *et al.*, 1982) in their air conditioners inputs the room temperature, outdoor temperature, time, and user-set temperature, and outputs a control value to keep the user-set temperature. Suppose that a user indicates that he or she wishes to change the control to adapt the environment to his or her preference. Then, the GA changes the characteristics of the NN by changing the number of neurons and weights (see Fig. 14).

4.6. NN-Based Fitness Function for GAs

The strategy of a GA search consists in multi-point searching, and the GA searches for an optimum point based on individuals. All individuals are applied to an application

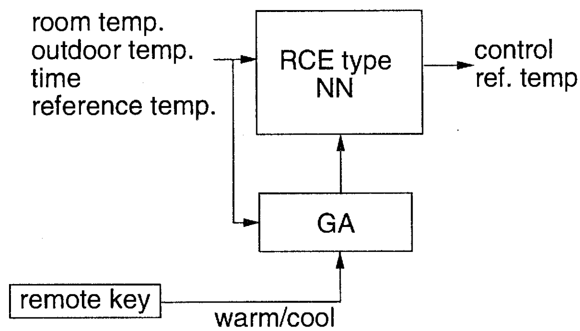


Fig. 14. Temperature control via an RCE neural network controller designed by a GA on the user's side.

task and evaluated by a fitness function for the search in the 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. That is why GAs are difficult to use for on-line processes.

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

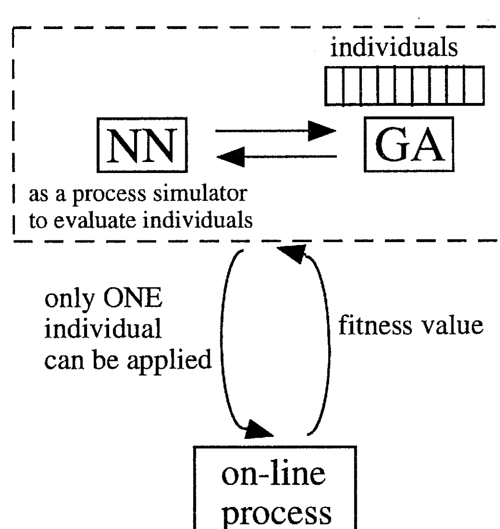


Fig. 15. GA with an NN fitness function that emulates the on-line process.

A GA whose fitness function uses an NN as a simulator of the plant growth was used for a hydroponics system (Morimoto *et al.*, 1993), cf. Fig. 16. The hydroponics system controls the temporal pattern of the water drainage and supply to the target plant in order 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. The 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 simulation and applied to the actual plant.

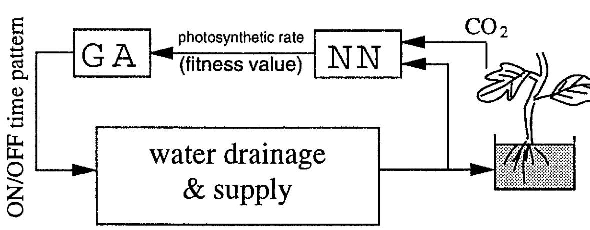


Fig. 16. Water control for a hydroponics system.

A similar approach is used for ethyl alcohol production (Karim and Rivera, 1992). A bioprocess takes the plant's place of the aforementioned hydroponics system. An NN of recurrent type learns the fermentation characteristics, and the GA decides against the optimum control parameters to maximize the alcohol fermentation.

Although an industrial application has not been developed yet, a computerized color recipe prediction system using a GA and NNs (Mizutani *et al.*, 1995) is an actual application expected soon. The NNs in this model are used as parts of a fitness function to increase the performance of the GA.

The color recipe prediction system inputs the spectrum of a given color and estimates the mixture rate of color pigments. Mizutani *et al.* applied a GA to estimate the ratios. To improve the precision of its estimate, they constructed a complex fitness function consisting of two NNs and a rule-based system (Mizutani *et al.*, 1995).

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

4.7. GAs Whose Performance Is Controlled by Fuzzy Rule Bases

A dynamic parametric GA of Fig. 18 with a fuzzy rule base changing GA parameters adaptively was proposed in 1993 (Lee and Takagi, 1993), and some papers followed. However, the research on this application has been insufficient. There has been no report on this type application in industry so far, except lab level research.

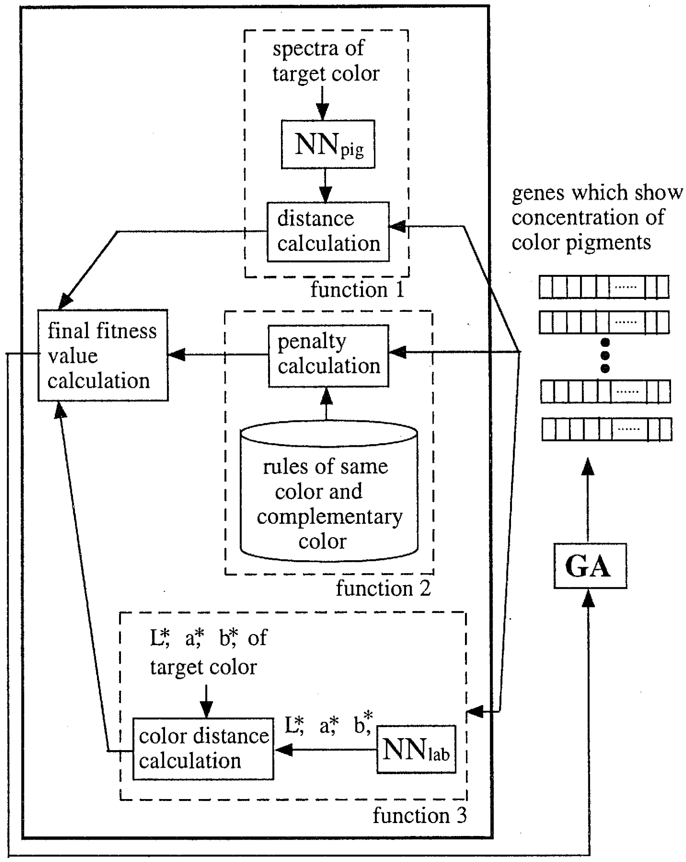


Fig. 17. GA fitness function consisting of two NNs and one rule base for a computerized color recipe prediction system.

5. Fundamental Patents

Since we have not surveyed NN+FS patents, a review of all of the patents cannot be presented 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 deals with any FS whose antecedent and/or consequent parts consist of an NN (Japan patent, 1996; US patent, 2000). Five of the models, (b), (e), (f), (g), and (h) among eight NN+FS models in Section 3.1 are of this type. A well-used ANFIS in a Matlab fuzzy toolbox is the model (e). There may be a 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.

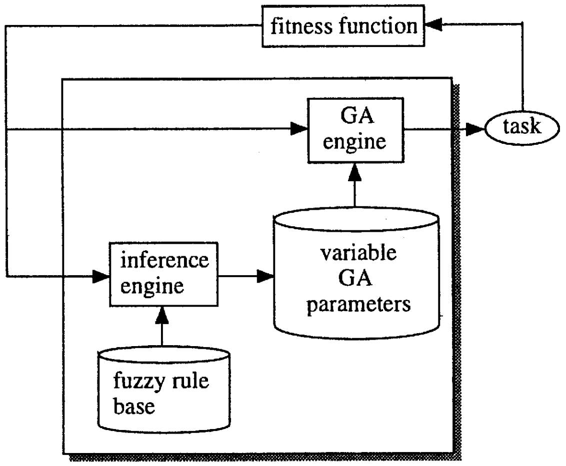


Fig. 18. FS observes the changes in the fitness values and dynamically changes GA parameters such as the population size, crossover rate, and mutation rate.

The second patent pertains to any FS that is designed using NNs (Japan patent, 1998; US patent, 2000). 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 as important. Since the NN+FS model that has the most significant contribution to business is the model (d), this patent is very practical.

The third patent applies to the model of NN-driven fuzzy reasoning (Japan patent, 1998; US patent, 1992; 1993) and is included in the first patent.

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 exciting conditions surrounding R&D on NN+FS, especially during the early stage. The NN+FS research rapidly spread from Japan to the rest of the world, and NN+FS technology has been widely used in commercial products and industrial systems since then. 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 any more. 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 NNs or FSs 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 technologies that will make a mark in the history of computational intelligence.

References

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. IEICE Part D-II, Vol.J72-D-II, No.8, pp.1200–1206 (in Japanese).

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

Berenji H.R. and Khedkar P. (1992): *Learning and tuning fuzzy logic controllers through reinforcements.* — IEEE Trans. Neural Networks, Vol.3, No.5, pp.724–740.

Butnariu D. (1977): *L-fuzzy automata. Description of a neural model.* — Proc. Int. Congress *Cybernetics Systems*, Bucharest, Romania, Vol.3, No.2, pp.119–124.

Chorayan O.G. (1982): *Identifying elements of the probabilistic neuronal ensembles from the standpoint of fuzzy sets theory.* — Fuzzy Sets Syst., Vol.8, No.2, pp.141–147.

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

Fukuda T., Ito S. and Arai F. (1993): *Recognition of human face using fuzzy interface and neural network.* — Trans. Japan Soc. Mech. Eng., Vol.59, No.558, Series C, pp.508–514 (in Japanese).

Furuhashi T. (1993): *Representation of fuzzy rules using neural networks.* — J. Japan Soc. Fuzzy Th. Syst., Vol.5, No.2, pp.204–217 (in Japanese).

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

Furuya T., Kokubu A. and Sakamoto T. (1989): *NFS: Neuro fuzzy inference system.* — Trans. Inf. Process. Soc. Japan, Vol.30, No.6, pp.795–798 (in Japanese).

Halgamuge S.K. and Glesner M. (1994): *Neural networks in designing fuzzy systems for real world applications.* — Fuzzy Sets Syst., Vol.65, No.1, pp.1–12.

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

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

Hayashi I. and Takagi H. (1988a): *Formulation of fuzzy reasoning by neural network.* — Proc. 4th Fuzzy System Symp. IFSA Japan, Tokyo, pp.55–60 (in Japanese).

Hayashi I. and Umano M. (1993): *Perspectives and trends of fuzzy-neural networks.* — J. Japan Soc. Fuzzy Th. Syst., Vol.5, No.2, pp.178–190 (in Japanese).

Hayashi Y., Buckley J.J. and Czogała E. (1992): *Direct fuzzyfication of neural network and fuzzyfied delta rule.* — Proc. 2nd Int. Conf. *Fuzzy Logic and Neural Networks (IIZUKA'92)*, Iizuka, Fukuoka, Japan, pp.73–76.

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

Hirota K., Yamauchi K. and Murakami J. (1992): *Image recognition based on fuzzy-neuro technology by using binocular stereoscopic vision.* — Proc. 8th Fuzzy System Symp., Hiroshima, Japan, pp.657–660 (in Japanese).

Horikawa A., Furuhashi T., Okuma S. and Uchikawa Y. (1990): *Composition methods of fuzzy neural networks.* — Proc. 16th Annual Conf. IEEE Industr. Electron. Soc., Pacific Grove, CA, USA, Vol.2, pp.1253–1258.

Horikawa S., Furuhashi T., Okuma S. and Uchikowa Y. (1991): *A Learning Fuzzy Controller Using a Neural Network.* — Trans. SICE, Vol.27, No.2, pp.208–215 (in Japanese).

Horikawa S., Furuhashi T. and Uchikawa Y. (1992): *Composition methods and learning algorithms of fuzzy neural networks.* — J. Japan Soc. Fuzzy Th. Syst., Vol.4, No.5, pp.906–928 (in Japanese).

Ichihashi H. and Watanabe T. (1990): *Learning control by fuzzy models using a simplified fuzzy reasoning.* — J. Japan Soc. Fuzzy Theory Syst., Vol.2, No.3, pp.429–437 (in Japanese).

Imasaki N., Kiji J. and Endo T. (1992): *A fuzzy rule structured neural networks.* — J. Japan Soc. Fuzzy Theory Syst., Vol.4, No.5, pp.985–995 (in Japanese).

Ishibuchi H. and Tanaka H. (1989): *Fuzzy regression analysis on fuzzy group: linear programming approach and neural networks approach.* — Proc. SICE Kansai Branch Symp., Osaka, Japan, pp.13–18 (in Japanese).

Ishibuchi H. and Tanaka H. (1992): *Fuzzy regression analysis using neural networks and its application.* — Trans. SICE, Vol.28, No.1, pp.116–124 (in Japanese).

Ishibuchi H., Fujioka R. and Tanaka H. (1992a): *Classification of fuzzy vectors by neural networks.* — Trans. Inst. Syst. Contr. Inf. Eng., Vol.5, No.5, pp.198–206 (in Japanese).

Ishibuchi H., Okada H. and Tanaka H. (1992b): *Learning of neural networks with fuzzy weights.* — Proc. 8th Fuzzy System Symp., Hiroshima, Japan, pp.185–188 (in Japanese).

Izumida M., Murakami K. and Aibara T. (1988): *Fuzzy clustering on neural network.* — IEICE Tech. Rep. PRU88-71, Vol.88, No.279, pp.1–8 (in Japanese).

Jang J-S.R. (1993): *Self-learning fuzzy controllers based on temporal back propagation.* — IEEE Trans. Neural Networks, Vol.4, No.1 pp.156–159.

Karim M. and Rivera S.L. (June, 1992): *Use of recurrent neural networks for bioprocess identification in on-line optimization by micro-genetic algorithms.* — Proc. 1992 Amer. Contr. Conf., Chicago, IL, USA, pp.1931–1932.

Karr C.L., Meredith D.L. and Stanley D.A. (1989): *Control of liquid level via learning classifier system, the SPIE.* — Int. Soc. Optical Eng., Vol.1095, Part 1, pp.78–85.

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

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: Fuzzy Logic for the Applications to Complex Systems (Chiang W. and Lee J., Eds.), Singapore: World Scientific, pp.478–482.

Kosko B. (1986): *Fuzzy cognitive maps.* — Int. J. Man Mach. Stud., Vol.24, No.1, pp.65–75.

Kosko B. (1991): *Neural Networks and Fuzzy Systems*, Englewood Ciffs, NJ,: Prince Hall, USA.

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

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

Lee S.C. and Lee E.T. (1974): *Fuzzy sets and neural networks.* — J. Cybern, Vol.4, No.2, pp.83–103.

Lee S.C. and Lee E.T. (1975): *Fuzzy neural networks.* — Mathematical Biosciences, Vol.23, No.1–2, pp.151–177.

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

Mizutani E., Takagi H. and Auslander M. (1995): *A cooperative system based on soft computing methods to realize higher precision of computer color recipe prediction.* — Proc. SPIE, Int. Soc. Optical Eng., Vol.2492, Part 1, pp.303–314.

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

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

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).

Nauck D. and Kruse R. (1993): *A fuzzy neural network learning fuzzy control rules and membership functions by fuzzy error backpropagation.* — Proc. 1993 IEEE Int. Conf. *Neural Networks*, San Francisco, CA, USA, Vol.2. pp.1022–1027.

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

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

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. Japan Soc. Fuzzy Theory Syst., Vol.4, No.2, pp.379–388 (in Japanese).

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

Ogawa Y., Morita T. and Kobayashi K. (1991): *A fuzzy document retrieval system using the keyword connection matrix and a learning method.* — Fuzzy Sets Syst., Vol.39, No.2, pp.163–179.

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

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

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

Reilly D.L., Cooper L.N. and Elbaum C. (1982): *A neural model for category learning.* — Biol. Cybern., Vol.45, No.1, pp.35–41.

Rocha A.F. (1981): *Neural fuzzy point processes.* — Fuzzy Sets Syst., Vol.5, No.2, pp.127–140.

Rocha A.F. (1982): *Basic properties of neural circuits.* — Fuzzy Sets Syst., Vol.7, No.2, pp.109–121.

Rocha A.F., Francozo E., Handler M.I. and Balduino M.A. (1980): *Neural languages.* — Fuzzy Sets Syst., Vol.3, No.1, pp.11–35.

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. Syst. Res. Amer. Assoc. Adv. Sci., Washington D.C., Vol.2, pp.528–533.

Sakawa M., Sawada K. and Douki Y. (1992): *Fuzzy multiobjective combinatorial optimization through neural networks.* — Proc. 8th Fuzzy Syst. Symp., Hiroshima, Japan, pp.145–148 (in Japanese).

Sawaragi T., Shibata K., Katai O., Iwai S. and Tsukada K. (1992): *Integrated Cognitive architecture for image understanding using fuzzy clustering und structured neural network.* — Proc. IEEE Int. Conf. *Fuzzy Systems*, San Diago, CA, USA, pp.21–28.

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, (Chiang W. and Lee J., Eds.), Singapore: World Scientific Publishing, pp.483–488.

Shiue L.C. and Grondin R.O. (1987): *On designing fuzzy learning neural-automata.* — Proc. IEEE 1st Int. Conf. *Neural Network (ICNN'87)*, san Diago, CA, USA, Vol.2, pp.299–307.

Takagi H. (1990a): *Ambiguity and knowledge information processing.* — J. System, Control and Information, Vol.34, No.5, pp.263–273 (in Japanese).

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

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

Takagi H. and Hayashi I. (1988): *Artificial-neural-network-driven fuzzy reasoning.* — Proc. Int. Workshop *Fuzzy System Applications (IIZUKA'88)*, IIZUKA, Japan, pp.217–218.

Takagi H. and Hayashi I. (1991): *NN-driven fuzzy reasoning.* — Int. J. Approx. Reason., Vol.5, No.3, pp.191–212.

Takagi H., Kouda T. and Kojima Y. (1991): *Neural-networks designed on approximate reasoning architecture.* — J. SOFT, Vol.3, No.1, pp.133–141 (in Japanese).

Takagi H., Suzuki N., Kouda T. and Kojima Y. (1992): *Neural networks designed on approximate reasoning architecture.* — IEEE Trans. Neural Networks, Vol.3, No.5, pp.752–760.

Takahashi H. (1993): *Features and applications of "The fuzzy-neuro system".* — J. Japan Soc. Fuzzy Theory Syst., Vol.5, No.2, pp.233–244 (in Japanese).

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

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

Toyota A. (1992): *News flush.* — J. Inst. Elect. Inf. Commun. Engineers, Vol.72, No.7, p.812 (in Japanese).

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

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

Wang P-Z. and Zhang D. (1992): *The netlike inference process and stability analysis.* — Int. J. Intell. Syst., Vol.7, No.4, pp.361–372.

Watanabe N., Kawamura A., Masuoka R., Owada Y. and Asakawa K. (1990): *Fuzzy control using neural network.* — Proc. 40th Meeting IPSJ, 3C-1 0, Tokyo, Japan, pp.148–149 (in Japanese).

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

Yamaguchi T. (1993): *Fuzzy associative memory system.* — J. Japan Soc. Fuzzy Theory Syst., Vol.5, No.2, pp.245–260 (in Japanese).

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

Yamaguchi T., Imasaki N. and Haruki K. (1990a): *A reasoning and learning method for fuzzy rules with associative memory.* — Trans. IEE Japan, Vol.110-C, No.3, pp.207–215 (in Japanese).

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

Japan patent No. 2553725 (1996).

Japan patent No. 2813567 (1998).

US patent No. 5168548 (1992).

US patent No. 5255344 (1993).

US patent No. RE36,823 (2000).