7917

N94- 32423

2486

P 7

FUZZY-NEURAL CONTROL OF AN AIRCRAFT TRACKING CAMERA PLATFORM

Dennis McGrath
Naval Air Warfare Center - Aircraft Division
Lakehurst, New Jersey

ABSTRACT

A fuzzy-neural control system simulation was developed for the control of a camera platform used to observe aircraft on final approach to an aircraft carrier. The fuzzy-neural approach to control combines the structure of a fuzzy knowledge base with a supervised neural network's ability to adapt and improve. The performance characteristics of this hybrid system were compared to those of a fuzzy system and a neural network system developed independently to determine if the fusion of these two technologies offers any advantage over the use of one or the other. The results of this study indicate that the fuzzy-neural approach to control offers some advantages over either fuzzy or neural control alone.

INTRODUCTION

Intelligent control has been dramatically affected in recent years by both fuzzy logic systems and artificial neural networks. Fuzzy logic controllers (FLC) provide a method for encoding knowledge of a system in non-exact (fuzzy) terms. The performance of fuzzy control systems with several simple rules has been successfully applied to a variety of non-linear control tasks. Unfortunately, the tuning required for conventional fuzzy systems to achieve maximum performance is often a time consuming trial-and-error process. Alternatively, neural networks (NN) provide a less structured, self-adaptive method for control where knowledge is acquired from experience. The effectiveness of these networks, however, is dependent on the quality and quantity of the training data. Many researchers have recently proposed methods for combining fuzzy logic with neural networks. The fusion of these two technologies has yielded a new type of intelligent control which exploits the advantages of both technologies, allowing the control engineer to combine knowledge *a priori* with knowledge *a posteriori*.

A camera mounted on a positioning platform can be used in aircraft guidance and recognition during final approach to an aircraft carrier. The movement of such a platform is dependent on relative position error and range of the target, as shown in Figure 1. The problem of controlling this platform is well-suited for fuzzy control, since a few simple rules can describe the desired characteristics of the system. Previous studies, which have compared fuzzy tracking to traditional target tracking methods such as the Kalman Filter method, have shown that FLC trackers can match or surpass the performance of conventional tracking systems[2].

The success of previous research in this area inspired the idea that fuzzy target tracking could be improved through some adaptive learning method. In this project, an aircraft tracking camera platform simulation was developed using a FLC tracking system. For the sake of comparison, a simple backpropagation network was implemented for the same task. A fuzzy neural network (FNN) was then developed using the fuzzy-neural cooperative method proposed by Kawamura et. al.[1]. This method was explored as a means of achieving adaptive fuzzy control by combining the structure of a fuzzy logic control system with the adaptive learning capabilities offered by neural networks.

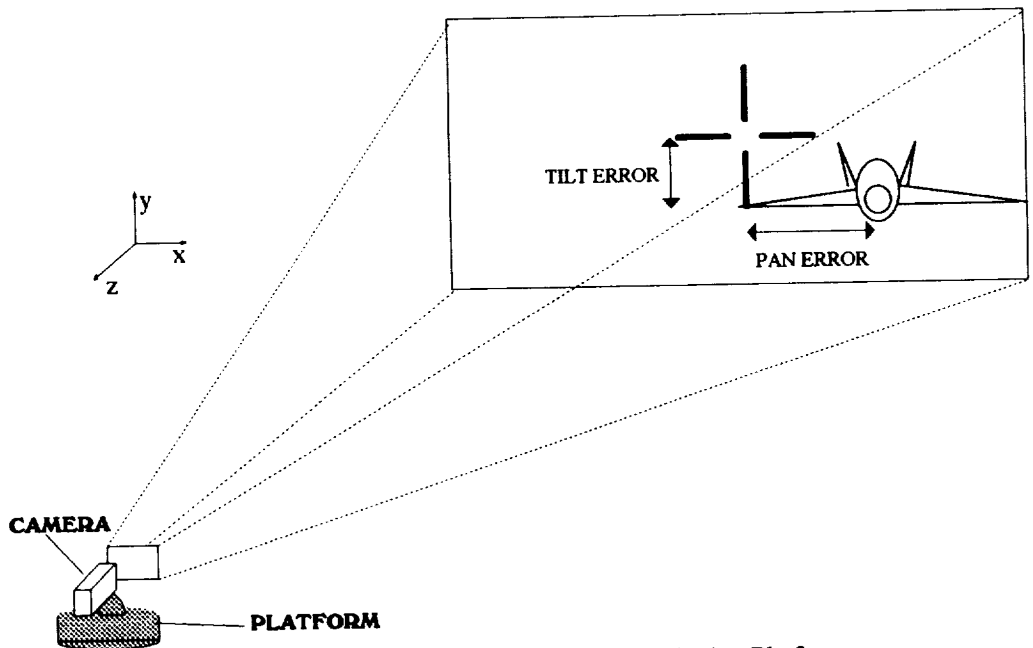


Figure 1 - Aircraft Tracking Using a Positioning Platform

FLC and NN TRACKING SYSTEMS

In camera tracking, the objective is to maintain a target image in the center of the camera's field-of-view (FOV). Typically, the exact x, y, and z location of the target is not available to the system[3]. The position error must be measured in pixels as the difference between the center of the camera's FOV and the centroid of the target image. Likewise, range can only be determined by the relative size (number of pixels) of the target image. For this simulation, however, explicit position information was available from radar observations of several approaches. The data was sampled at 20 Hz, and the approaches typically lasted 45 seconds giving approximately 900 data points per approach.

Pan and tilt are independent functions controlled by separate servos. The rates of pan and tilt are dependent on azimuth and elevation error, respectively, as well as range. In this simulation, a maximum pan rate of $\pm$ 15°/s and a maximum tilt rate of $\pm$5°/s were used. In the case of carrier approaches, panning is a much more difficult tracking task than tilting, since more abrupt aircraft movements take place in the x domain than in the y domain. Therefore, only the pan functions will be described for the remainder of this paper, with the understanding that the tilt control system is functionally identical to the pan system.

Fuzzy rules for camera tracking are easily derived by verbalizing the method that a person might use to track a moving target in photography or skeet shooting. These rules take the form of "if the target is to the left and it is close, then pan quickly to the left...", and so on. The FLC tracking system developed for this study is shown in Figure 2. It employs three membership functions for ERROR and two for RANGE. Six rules correspond to five output membership functions. Centroid defuzzification is used to determine the value of PANRATE.

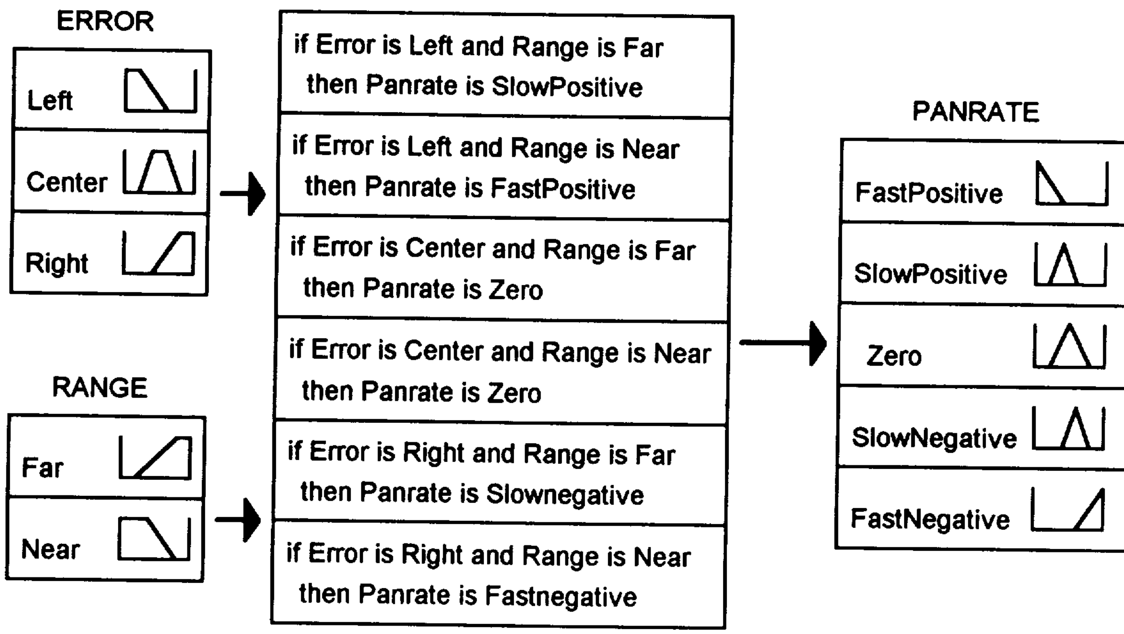


Figure 2 - A fuzzy logic controller for aircraft tracking

The same functions can be performed by a backpropagation network consisting of two inputs, one output, and one hidden layer as shown in Figure 3.

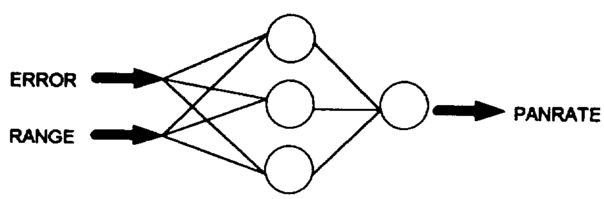


Figure 3 - A simple backpropagation network for aircraft tracking

3. FNN TRACKING SYSTEM

The development of a FNN tracking system required that a network be created which modeled the behavior of the fuzzy tracking system. In this way, the essential operations of fuzzification, inferencing, and defuzzification are accomplished through a series of functions represented by selectively connected nodes. The network architecture and initial system parameters must be empirically developed so that the FNN matches the functions of the existing FLC.

Input membership functions are easily realized through the use of the sigmoidal functions associated with neurons:

$$S(\Sigma w_{ij} o_i) = \frac{1}{1 + e^{-\Sigma w_{ij} o_i + \theta}} \qquad (1)$$

where $\Sigma w_{ij} o_i$ is the sum of the input-weight products and θ is the bias or threshold. As Figure 4 shows, the input functions for ERROR and RANGE are represented as logistic sigmoids for Left, Right, Near, and Far and as the difference between two sigmoids in the case of the bell-shaped Center function. The appropriate weight and threshold values were determined, and the antecedent membership functions take the following form:

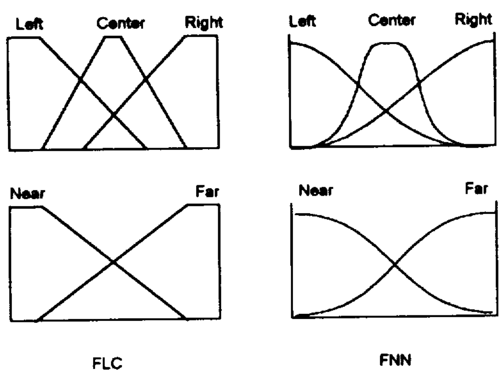
$$LEFT = \frac{1}{1+e^{13.3ERROR-3.3}} \quad (2a)$$

$$CENTER = \frac{1}{1+e^{-13.3ERROR+3.3}} - \frac{1}{1+e^{-13.3ERROR+9.9}} \quad (2b)$$

$$RIGHT = \frac{1}{1+e^{-13.3ERROR-9.9}} \quad (2c)$$

$$FAR = \frac{1}{1+e^{-13.3RANGE-6.6}} \quad (2d)$$

$$NEAR = \frac{1}{1+e^{13.3RANGE-6.6}} \quad (2e)$$



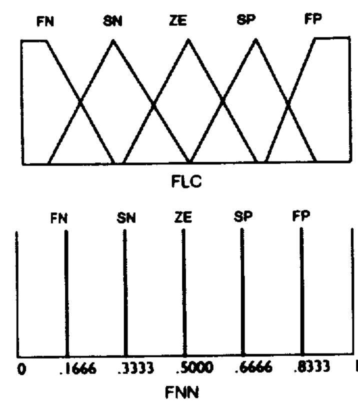


Figure 4 - Antecedent membership functions as sigmoids **Figure 5** - Consequent membership functions as singletons

The inference process correlates input membership values to output membership values. All rules in this system are conjunctive, meaning that they feature the AND operation. Using max-product inferencing, the result of the AND operation is simply the product of the inputs. Alternatively, max-min inferencing could be used, where the conjunctive inference nodes would use the min(A,B) operator.

Output (consequent) membership functions are represented as singleton functions for the purpose of simplifying the defuzzification process. Singleton functions have only one value. In this case they represent the approximate centroid of the original output membership functions, as shown in Figure 5. By assigning the singleton values to weights at the output layer, defuzzified values can be computed as a weighted average of these singletons based on the truth value of the consequent membership :

$$OUTPUT = \frac{\Sigma O_i w_{ij}}{\Sigma O_i} \quad (3)$$

The resulting network takes the form shown in Figure 6. Hidden layers represent input, rules, and output functions. When ERROR and RANGE values, normalized to [0,1], are fed forward into the network, they are is converted to membership values. These membership values are then processed by rule nodes, yielding output (consequent) membership values. These membership values are then defuzzified as explained previously. To this point, the behavior of the FNN is identical to that of the FLC. Parametric learning is then accomplished via the generalized delta rule, as in many other backpropagation networks.[4] The system error is simply the difference between the new center of the camera's field-of-view and the actual aircraft position. Adjustments are made to weights at the input and output layers based on the calculated deltas derived from the system error, according to the relation:

$$\Delta w_{jk} = \eta \, \delta_k o_j \quad (4)$$

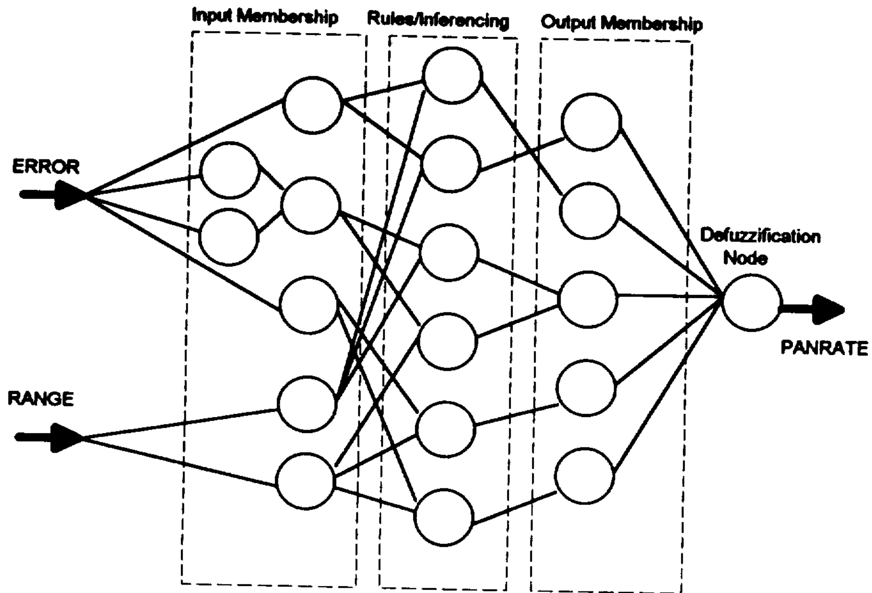


Figure 6 - A fuzzy neural network for aircraft tracking

SIMULATION RESULTS

Using data from several different carrier approaches, the FLC, NN, and FNN tracking methods were implemented in simulation. Figure 7 is a comparison of the performance of each system on a section of the test approach after one, fifteen, and fifty approaches. The dotted lines represent the aircraft movement about the centerline of the approach path, and the solid lines represent the line of sight of the camera.

The performance of the FLC was good, but the camera's line-of-sight lagged slightly behind the aircraft's postition. Since it has no parametric adjustment capability, its performance obviously did not improve with time. As was expected, the NN tracker began with random behavior on the first approach. By the end of the first approach, the network began to learn the basics of panning and tilting, but left much room for improvement. The performance had improved substantially by the fifteenth approach, where the line-of-sight lagged slightly and tended to overshoot the aircraft during motion reversals. By the fiftieth approach, the lag had diminished to near-zero, but some overshoot was still evident.

The FNN, even on the first approach showed improvement over the FLC, demonstrating much less lag and a slight overshoot. By the fifteenth approach, the lag was nearly eliminated and the overshoot was significantly reduced. After the fiftieth approach, the line-of-sight of the FNN tracker was virtually indistinguishable from the path of the aircraft. Since it began with the intrinsic knowledge of the FLC, its learning outpaced that of the NN tracker. Under all conditions, the FNN outperformed the other two methods.

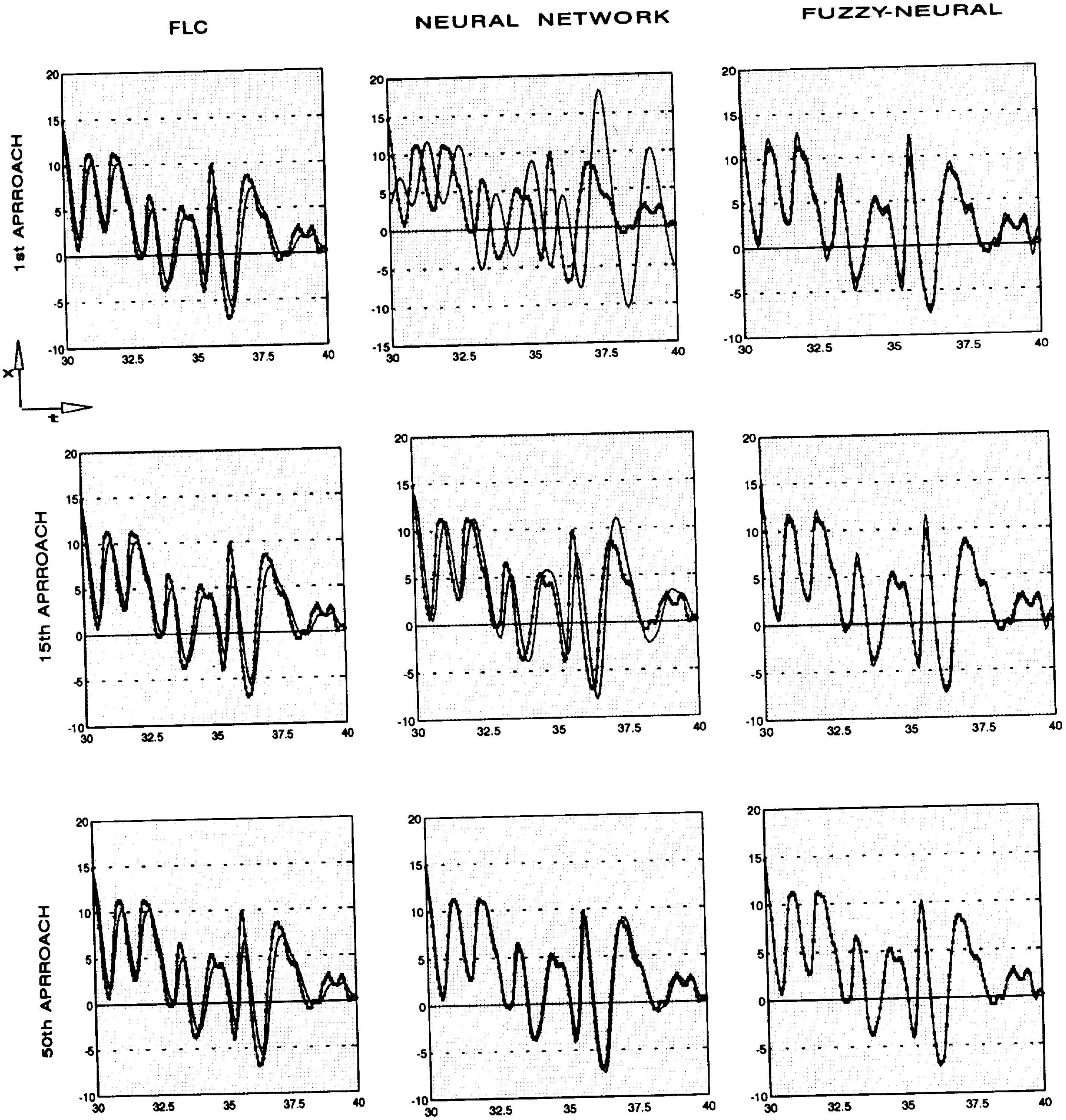


Figure 7 - Comparison of results from FLC, NN, and FNN tracking systems

CONCLUSIONS

A fuzzy-neural network was successfully implemented for automatic camera platform positioning, eliminating the need to manually adjust FLC parameters for optimum performance. The simulations conducted in this study demonstrated that when structured knowledge and learning ability are combined, the result is robust performance that improves over time. The fusion of these two types of machine intelligence represents an important step in the evolution of intelligent control systems.

The FNN may be applied to a wide range of non-linear control systems, particularly where fuzzy control has already proven successful. In any system whose desired behavior can be described through a series of fuzzy rules, those rules can be translated into a network and thereby achieve the ability to adapt. The learning algorithm does not add an excessive amount of computation to an existing FLC, and thus it should be feasible for real-time applications where the system can learn "on the fly". Such experiments are planned for this system in upcoming research.

REFERENCES

1. Kawamura, S., Watanabe, L., Okada, H., and Asakawa K., "A Prototype of Neuro-Fuzzy Cooperation System", *Proceedings of the IEEE International Conference on Fuzzy Systems 1992*, San Diego, CA, 1992

2. Kosko, B., Neural Networks and Fuzzy Systems, Prentice Hall, Englewood Cliffs, NJ, 1991

3. Lea, R., Chowhudry, I., Shehadeh H., and Jani, Y.: "Design and Performance of the Fuzzy Tracking Controller in Software Simulation", *Proceedings of the IEEE International Conference on Fuzzy Systems 1992*, San Diego, CA, 1992

4. Pao, Y., Adaptive Pattern Recognition and Neural Networks, Addison-Wesley Co. Inc., 1989.

54-63

7918

N94-32424

2487

p. 10

EMPIRICAL MODELING FOR INTELLIGENT, REAL-TIME MANUFACTURE CONTROL

Xiaoshu Xu
American Welding Institute
10628 Dutchtown Rd.
Knoxville, TN, 37932

ABSTRACT

Artificial neural systems (ANS), also known as neural networks, are an attempt to develop computer systems that emulate the neural reasoning behavior of biological neural systems (e.g. the human brain). As such, they are loosely based on biological neural networks. The ANS consists of a series of nodes (neurons) and weighted connections (axons) that, when presented with a specific input pattern, can associate specific output patterns. It is essentially a highly complex, non-linear, mathematical relationship or transform. These constructs have two significant properties that have proven useful to the authors in signal processing and process modeling: noise tolerance and complex pattern recognition. Specifically, the authors have developed a new network learning algorithm that has resulted in the successful application of ANS's to high speed signal processing and to developing models of highly complex processes. Two of the applications, the Weld Bead Geometry Control System, and Welding Penetration Monitoring System is discussed in the body of this paper.

INTRODUCTION: ARTIFICIAL NEURAL SYSTEMS

Artificial Neural Systems (ANS) are loosely based on biological neural networks offer a computer technology that is a useful tool in process modeling and signal processing. The ANS consists of a series of nodes (neurons) and weighted connections (axons). As with a biological neural network, the assignment of the values of the weights and the size and configuration of the network is the key to a successful net. Unfortunately, we have only begun to understand the inner workings of these constructs. Consequently, relatively crude tools are currently employed to develop working networks.

Typically, the approach to model development is to develop a thorough understanding of the underlying basic scientific or engineering principles of the process. Then, a model is developed that is based on mathematical relationships inherent in the process parameters. The principal drawback with this approach is the time and effort required to develop an understanding of these basic scientific or engineering relationships. Depending on the complexity of the problem, it can take many years and an extensive research program to develop the relationships. Often, instead, a number of simplifying assumptions are made and an approximate model is developed. That approach, while being an expedient method for developing an approximate model that could be useful, provides only a theoretical approximation that may not be valid for the actual problem application.

The artificial neural system, when a network can be found to solve the problem, provides an accurate model of the process or signal. Neural networks are empirical bases systems that, when presented with a specific input pattern, can associate specific output patterns. It is, essentially, a highly complex, non-linear, mathematical relationship or transform. However; it is not necessary for the developer of such a system to understand the basic underlying principles of a process in order to develop a highly accurate ANS based model of the process. Thus, in this way it is quite different from other mathematical modeling approaches.

Basic Principles