5/9-67
161819
p-9

N93-29535

Clustering of Tethered Satellite System Simulation Data by an Adaptive Neuro-Fuzzy Algorithm

Sunanda Mitra and Surya Pemmaraju
Department of Electrical Engineering
Texas Tech University
Lubbock, TX 79409-3102

ABSTRACT

Recent developments in neuro-fuzzy systems indicate that the concepts of adaptive pattern recognition when used to identify appropriate control actions corresponding to clusters of patterns representing system states in dynamic nonlinear control systems, may result in innovative designs. A modular, unsupervised neural network architecture, in which fuzzy learning rules have been embedded is used for on-line identification of similar states. The architecture and control rules involved in Adaptive Fuzzy Leader Clustering (AFLC) allow this system to be incorporated in control systems for identification of system states corresponding to specific control actions. We have used this algorithm to cluster the simulation data of Tethered Satellite System (TSS) to estimate the range of delta voltages necessary to maintain the desired length and length rate of the tether. The AFLC algorithm is capable of on-line estimation of the appropriate control voltages from the corresponding length error and length rate error without a priori knowledge of their membership functions and familiarity with the behavior of the Tethered Satellite System.

I. INTRODUCTION

In spite of recent developments in nonlinear dynamical systems modeling, analytical as well as implementation difficulties still remain in many controller design problems [1]. Integration of fuzzy learning rules with neural networks may provide flexibility in designing models for these systems [2]. In supervised learning, a set of correct control actions can be learned and used to estimate other actions required in a dynamic control system whereas unsupervised learning may suggest appropriate control actions corresponding to system states forming a pattern cluster. Applications of tethers in space have demonstrated scope for control using the latter technique. Due to the elasticity and finite mass distribution of the tether, any tethered system has complex, nonlinear dynamics. As a result, control of these systems is not easily achieved. Fuzzy logic based controllers [3] have handled nonlinearities of such a system quite well with no requirement to fully understand the dynamics of the system. Normally, a fuzzy controller defines some linguistic variables and generates fuzzy membership functions and a rulebase for the controlling parameters, using some a priori information regarding the system. Instead, we have used a hybrid neural-fuzzy clustering algorithm namely, Adaptive Fuzzy Leader Clustering (AFLC) [4] to find optimal control actions. This clustering algorithm can be used for optimal clustering in many pattern recognition problems [4] as well as for examining control actions required for complex systems with nonlinear dynamics.

Cluster analysis has been a significant research area in pattern recognition for a number of years [5]-[6]. Despite significant improvements in clustering of specific data sets by incorporating fuzzy membership concepts into hard clustering [7]-[9], partitioning real and noisy data sets still poses difficulties, thus keeping this research area wide open. Integration of fuzzy clustering concepts with neural network architectures may provide further flexibility in identification of appropriate data clusters. One such attempt is presented here by using the AFLC algorithm to cluster the simulation data of the Tethered Satellite System (TSS). AFLC has an unsupervised neural network architecture developed from the concepts of ART-1 [10]-[11], and uses the set of nonlinear equations for centroid and membership values as developed in the fuzzy C means (FCM) algorithm [12] for updating the centroid locations. AFLC learns on-line in a stable and efficient manner and adaptively clusters input discrete or analog signals into classes without a priori knowledge of the input data structure. Each resultant output cluster has a prototype which represents all the data samples in that cluster. In this paper, we apply the AFLC algorithm to effectively cluster the simulation data of the Tethered Satellite system, containing the crucial parameters for controlling the system behavior. This paper is organized as follows. Section II gives a brief description of the AFLC system and algorithm. Section III outlines the behavior of a Tethered Satellite System and suggests a possible architecture for controlling it. Section IV presents the test results of AFLC operation on the TSS data set used. Finally, Section V addresses the potential applications of AFLC in recognition and control of complex data sets and systems respectively.

II. ADAPTIVE FUZZY LEADER CLUSTERING

A. The AFLC algorithm overview

AFLC is primarily used as a classifier of feature vectors employing an on-line learning scheme [4]. The algorithm basically consists of three procedures, recognition, comparison and updating. It involves a two-stage classification which takes place in the recognition and the comparison stages. The system is initialized with the input of the first feature vector X_1 and the number of clusters (C) is set to zero. Similar to leader clustering, this first input forms the prototype for the first cluster. This cluster is represented by a node in the recognition layer of the AFLC system. Connection of any such node i to the input vector X_j in the comparison layer is established through a set of multiplicative weights referred to as the bottom-up weights (b_{ij}), whose values correspond to a normalized version of the cluster prototype. Subsequent to this initialization, normal operation commences [4].

The normalized version of the next input vector is applied to the bottom-up weights of all the existing cluster nodes in a simple competitive learning scheme, or dot product. The activation level, Y, of node i in the recognition layer is

$$Y_i = \sum_{k=1}^{p} X_j b_{ij} \tag{1}$$

where p is the dimension of the input feature vector. The recognition stage winner is the node with the maximum value of Y. In the specific case of the second input vector, there is only one recognition layer node which was activated by the first input. This node will win the competition by default, which would lead to a very disappointing performance. Additional processing, however is obtained, as in ART [11], by attempting to match the input to a top-down expectation. This takes place in the comparison stage. The Euclidean distance between the original input vector and the cluster prototype of the winner node is calculated. This value is then compared to the average distance from the centroid of all the samples belonging to that cluster. If this distance ratio R is less than a user-specified threshold τ, then the input is found to belong to the cluster originally activated by the recognition layer. This relation can be represented as

$$R = \frac{\|x_j - v_i\|}{\frac{1}{N_i}\sum_{k=1}^{N_i}\|x_k - v_i\|} < \tau \qquad (2)$$

where : $j = 1.....N_i$, is the number of samples in class i and v_i is the centroid of class i. τ is called the vigilance parameter and determines the compactness within a cluster and the inter cluster separation. The choice of the value of τ is critical in some applications where unlabelled data consisting of overlapping clusters is to be classified precisely. If the comparison of the input and the cluster prototype does not satisfy the threshold requirement, a search is implemented. This is accomplished by deactivating the currently activated recognition layer neuron with the help of the reset signal and repeating the classification process. If no cluster exists which meets the distance ratio criterion, then a new node is established.

When an input is classified to belong to an existing cluster, it is necessary to update the expectation (prototype) and the bottom-up weights associated with that cluster. This is done in the last stage using the fuzzy C means formulae. The cluster prototype or centroid is recalculated as a weighted average of all the elements within the cluster. The membership values μ_{ij} of all the samples in the updated cluster with respect to the new centroid v_i are calculated. Since the membership values are dependent on the centroid positions, the relocation of the centroid in the winner cluster affects the membership values of the other data samples in the remaining clusters and hence they are recalculated. Equations 3 and 4 given below are the fuzzy C means [12] equations that have been employed for updating the cluster centroid and the membership values of the data samples. It is to be noted that equation 3 updates v_i only in the columns currently associated with class i whereas equation 4 involves a full membership updating process. The summation in equation 3 would extend from 1 to N in full FCM updating of the class prototypes. Here, N is the total number of data samples and m is the parameter which defines the fuzziness of the results and is normally set to be between 1.5 and 30. For the following application, m was experimentally set to a value of 2.

$$v_i = \frac{1}{\sum_{j=1}^{N_i}(\mu_{ij})^m} \sum_{j=1}^{N_i}(\mu_{ij})^m X_j \qquad for \quad 1 \le i \le C \qquad (3)$$

$$\mu_{ij} = \frac{\left(\dfrac{1}{\left\|X_j - v_i\right\|^2}\right)^{1/(m-1)}}{\sum_{k=1}^{C}\left(\dfrac{1}{\left\|X_j - v_k\right\|^2}\right)^{1/(m-1)}} \quad \text{for} \quad 1 \le i \le C \quad \text{and} \quad 1 \le j \le N \qquad (4)$$

The updating process is followed by a verification procedure whose function is to check if the previous classification is still valid. The location of the samples which come and join the cluster at a later stage can often cause the prototype (centroid) of the cluster to shift in a particular direction. This depends on the order in which the data is fed to the algorithm. As a result, the distance of some of the data samples from the new centroid of the cluster to which they originally belonged, will increase drastically and hence the vigilance condition might not be satisfied any more. This could result in a misclassification if these samples are found to be closer to another neighboring cluster and satisfy the vigilance condition with respect to it.

The modified AFLC avoids this problem by means of a verification procedure which tests if all the samples in the updated cluster conform to the original classification. A sample that does not satisfy the original classification condition is reclassified by minimizing a simple error function given in equation 5. This error function helps in selecting the cluster which is closest to the input sample, by minimizing the weighted sum of the squares of the distances [12]. Therefore this verification process ensures that the algorithm is immune to the order of data presentation.

$$J(\mu, v) = \sum_{i=1}^{C}\sum_{j=1}^{N_i}\left\|X_j - v_i\right\|^2 \left(\mu_{ij}\right)^m \qquad (5)$$

III. CLUSTERING OF TETHERED SATELLITE SYSTEM PARAMETERS

A. Behavioral Characteristics of TSS

The TSS consists of a reel powered by an electric motor, satellite thrusters, and the orbiter attitude control system [15]. Evaluation of the overall control of the TSS is done by means of tether length, tether tension, longitudinal and librational oscillations as shown in Figure 1. The elasticity of tether and the gravity gradient forces acting on the satellite result in longitudinal oscillations. The motion of the tethered satellite along the velocity vector, i.e., in a line from the nose to the tail of the orbiter causes in-plane libration and that towards the starboard side of the shuttle causes out-of-plane libration. Since it is only the tether length and tether tension that can be directly measured and controlled, the in-plane and out-of-plane libration amplitude have to be indirectly controlled through tether length and length rate maintenance.

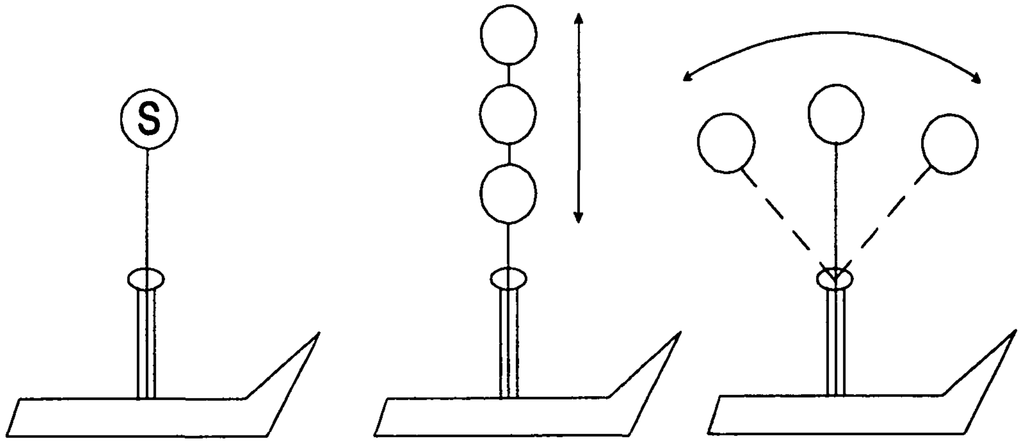

Figure 1. Longitudinal and Librational Oscillations in a Tethered payload System [15]

B. TSS Control Parameters

The parameters that are used to control the TSS are the Length Error, the Length Rate Error and the Delta Voltage. Using these three parameters, one can design a stable system by utilizing fuzzy membership functions for such parameters. However some familiarity with the TSS and its behavior is essential to estimate appropriate values for those membership functions [15]. When the TSS simulation data is fed to the AFLC system, it classifies the data into clusters depending on the value of the vigilance parameter. The output of the AFLC system is a classification of the input data into distinct clusters. Each cluster specifies the range of the length error and length rate error for a given value of delta voltage.

This performance is analogous to that of a rulebase describing the relation between the inputs and the output in terms of some linguistic variables. Membership functions for these linguistic variables are defined in terms of its range and its belief values using some intuitive knowledge of the physical system [15]. However, in our case, no a priori knowledge is required. The system being an unsupervised network, learns on-line from the data and classifies each data vector into the appropriate class depending on its past learning. The three TSS parameters can be considered to be the state variables of the system. Input data consists of the length error, the length rate error and the corresponding delta voltage values sampled at 50 seconds intervals.

Figure 2 shows the suggested schematic for an adaptive fuzzy control system using AFLC. Here the AFLC system combined with a functional link acts as a fuzzy controller. A look-up table and an estimator can form the basis for this functional link. The output of this controller is given as input feedback to the tethered satellite simulation system and the actual output of the physical system forms the next stage input to the AFLC algorithm.

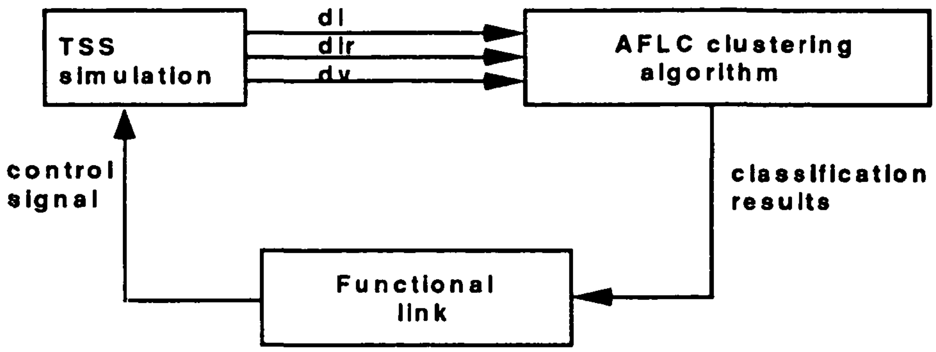

Figure 2. Schematic of a suggested fuzzy controller

IV. TESTS AND RESULTS

The data set consists of 1765 samples obtained from the massless tether model in the Tethered Satellite System simulation. This data has been collected at intervals of 50 seconds. Each input vector consists of length error (dl), length rate error (dlr) and the corresponding delta voltage (dv). The entire data set has been classified using the AFLC algorithm. The value of τ has been chosen to be 2.5. Figure 3 gives a table showing the classification results. It can be inferred from the table that the data set has been broadly classified into four categories. A few points that have not been classified as belonging to any of the four clusters can be treated as noise/outliers. Each cluster represents a specific range of input and output parameters.

CLASS	SAMPLES	RANGE OF dl	RANGE OF dlr	RANGE OF dv
1.	657	0.19 to 1.49	-0.135 and 0.0 to 0.0096	0.315 to 0.670
2.	528	1.96 to 2.63	-0.0304 to -0.000033 and 0.0 to 0.00786	1.77 to 2.85
3.	383	2.93 to 6.07	-0.0088 to -0.000009 and 0.00001 to 0.0087	3.00 to 4.29
4.	174	-5.2 to -1687.1	-0.765 to -0.01 and 0.01 to 0.354	-3.59 to -7.47

Figure 3. Adaptive Fuzzy Leader Clustering of TSS Data

The results from this table are comparable to those obtained from a rulebase, which specifies the output category for a given combination of input categories [15]. However, the results obtained by our algorithm do not require a priori information of the system parameters. AFLC actually classifies the data structure using an on-line learning scheme. Hence this is a more realistic approach of solving the problem without requiring any intuitive knowledge of the system. Figure 4 shows a plot displaying the clusters in a two-dimensional feature space. Incorporation of these clustered delta voltages into the orbiter operations simulator (OOS) should provide smoother operational characteristics of the TS system.

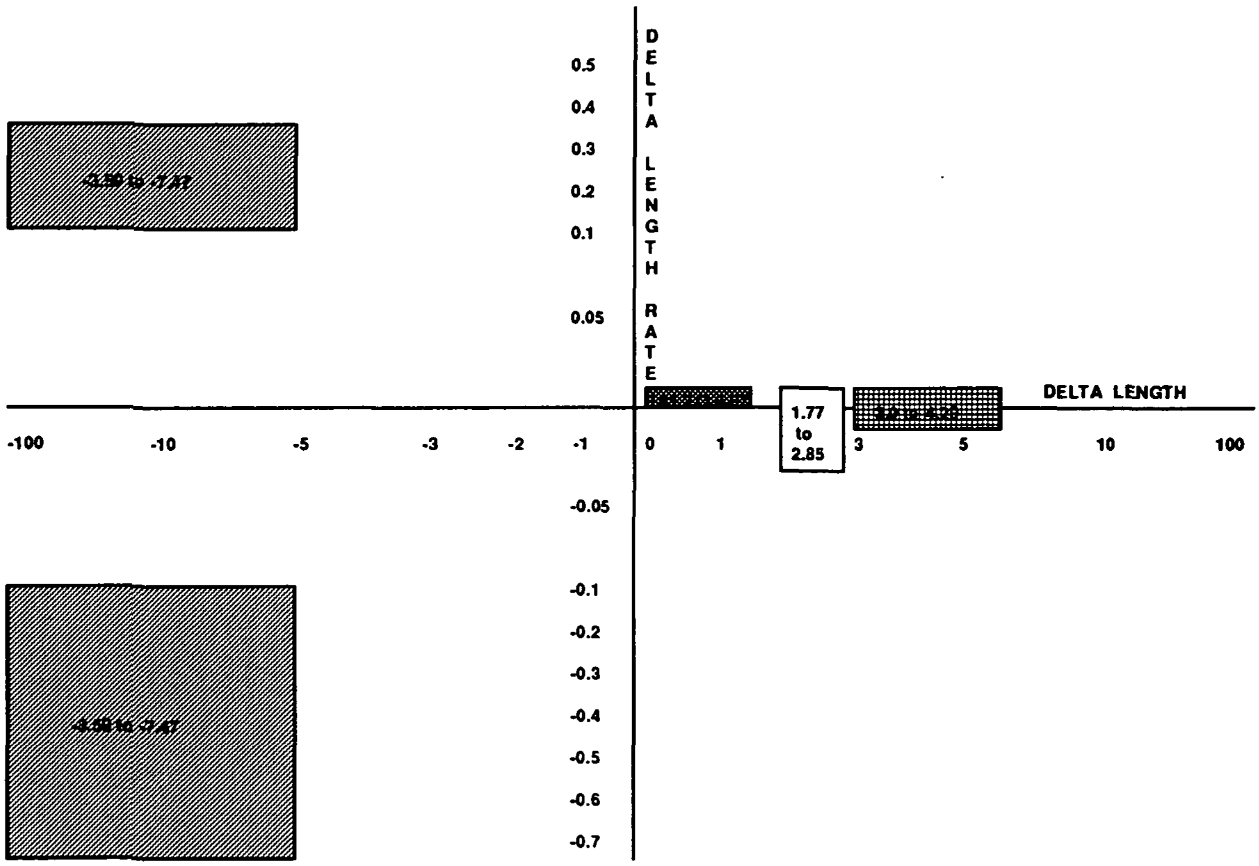

Figure 4. Control Delta Voltages corresponding to length error and length rate error

V. CONCLUSION

This neuro-fuzzy algorithm, namely AFLC ensures stable, consistent learning of the membership of the new on-line inputs without a priori knowledge of the data structure. The flexibility in the algorithm makes it possible to apply many of the concepts of AFLC operation to typical control problems.

The use of AFLC to generate dynamic control actions corresponding to system state clusters of a nonlinear dynamic system, stems from similar concepts suggested by recent works using neural network for control [1],[2],[16]. Such fusion of adaptive pattern recognition and control actions may result in innovative designs of dynamic nonlinear control systems and deserves further investigation. Better integration of fuzzy membership function with self-organizing neural network learning rules have been achieved recently [17],[18] demonstrating the applicability of neuro-fuzzy algorithms in complex decision making processes.

VI. ACKNOWLEDGMENT

This work was supported in part by a grant from NASA-JSC under contract # NAG9-509, a grant from Northrop Corporation under contact # HHH7800809K and a grant from Texas Advanced Technology program under contract # 003644-047. The authors wish to acknowledge collaboration with R. Lea, Y. Jani and J. Villarreal of NASA-JSC and thank them for providing us with the TSS simulation data.

VII. REFERENCES

[1] W.T. Miller III, R.S. Sutton and P.J. Werbos, Neural Networks for Control, MIT Press, Cambridge, MA , 1990.

[2] Y. H. Pao, Adaptive Pattern Recognition and Neural Networks, Addison-Wesley Reading, 1990.

[3] R.N. Lea and Y. Jani, "Applications of Fuzzy Logic to Control and Decision making," Technology 2000 Proc., NASA Conference Publication 3109, Vol 2, pp. 67, 1990.

[4] S.C. Newton, S. Pemmaraju and S. Mitra, "Adaptive Fuzzy Leader Clustering of Complex Data Sets in Pattern Recognition," IEEE Trans. on Neural Networks, Vol. 3, No. 5, Sep 1992.

[5] R.O. Duda and P.E. Hart, Pattern Classification and Scene Analysis, John Wiley and Sons, New York, 1973.

[6] A.K. Jain and R.C. Dubes, Algorithms for Clustering Data, Englewood Cliffs, NJ, Prentice-Hall, 1988.

[7] L.A. Zadeh, "Fuzzy Sets," Information and Control, 8, pp 338-353, 1965.

[8] J.C. Bezdek, "A Physical Interpretation of Fuzzy ISODATA," IEEE Trans. on Systems, Man and Cybernatics, Vol. 6, pp 387-389, 1976.

[9] I. Gath and A.B. Geva, "Unsupervised Optimal Fuzzy Clustering," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 11, no. 7, pp 773-781, 1089.

[10] G. A. Carpenter and S. Grossberg, "A massively Parallel Architecture for a Self-organizing Neural Pattern Recognition Machine," Computer Vision, Graphics and Image Processing, 37, pp 54-115, 1987.

[11] G.A. Carpenter and S. Grossberg, "ART-2 : Self Organization of Stable Category Recognition Codes for Analog Patterns," Applied Optics 2 , pp 4919-4938, 1987.

[12] J.C. Bezdek, Pattern Recognition with Fuzzy Objective Functions Algorithms, Plenum, New York, 1981.

[13] S.C. Newton and S. Mitra, "Self-organizing Leader Clustering in a Neural Network Using a Fuzzy Learning Rule," SPIE Proc. on Adaptive Signal Processing, Vol. 1565, July 1991.

[14] S.C. Newton and S.Mitra, "Application of Neural Networks in Fuzzy Pattern Recognition," presented at the Research Conference on Neural Networks for Vision and Image Processing, Boston University, May 10-12, 1991.

[15] R. N. Lea, J. Villarreal, Y. Jani and C. Copeland, "Fuzzy Logic Based Tether Control," NAFIPS 1991 Workshop Proc. pp 398-402, 1991.

[16] K.S. Narendra and K. Parthasarathi, "Identification and Control of Dynamical Systems using Neural Networks," IEEE Transactions on Neural Networks, Vol. 1, No. 1, pp 4-26, March 1990.
[17]. J.C. Bezdek, E.C. Tsao and N.R. Pal, "Fuzzy Kohonen Clustering Networks," Proc. of First IEEE Conference on Fuzzy Systems, San Diego, March 1992.
[18] Y.S. Kim and S. Mitra, "A Comparative study of the performance of Fuzzy ART type clustering algorithms in pattern recognition," SPIE Proc. on Intelligent Systems, Robotics and Computer Vision, Boston, November 1992.