Cluster Formation Using Kohonens Self-Organising Map

Vivekanand S Gogi[1] & K S Badarinarayana[2]

Dept of I E & M, R V College of Engineering, Bangalore, India
[2]Principal, M V J college of Engineering, Bangalore, India
E-mail : gogivs@rediffmail.com[1], bakesu@yahoo.com[2]

***Abstract* - An Artificial Neural Network (ANN) is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. The key element is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems. The objective of this research paper is to study the clustering of requests from the Flexible Manufacturing System machines to the Automated Storage/Retrieval system and to optimize the clustering of the requests. Artificial Neural Networks has been used for clustering the requests from the machines. Unsupervised Learning (training) algorithm in Artificial Neural Networks is used and implemented using C++ programming language. The requests from the machines are successfully analyzed and optimization of clusters is done using Kohonens Self-Organizing Map technique.**

Keywords - Artificial Neural Networks (ANN), Flexible Manufacturing System Scheduling, Kohonens Self-Organizing Map technique

I. INTRODUCTION

A flexible manufacturing system (FMS) is a manufacturing system in which a flexibility allows the system to react to the changes, whether predicted or unpredicted. This flexibility is generally considered to fall into two categories, which both contain numerous subcategories.

An automated storage and retrieval system consists of a variety of computer-controlled methods for automatically placing and retrieving loads from specific storage locations.

Artificial Neural Networks are relatively crude electronic models based on the neural structure of the brain. The brain basically learns from experience. It is natural proof that some problems that are beyond the scope of current computers are indeed solvable by small energy efficient packages. The brain modeling promises a less technical way to develop machine solutions. It provides a more graceful degradation during system overload.

II. LITERATURE REVIEW

The first step toward artificial neural networks came in 1943 when Warren McCulloch conducted a study on neurons. The research has been vigorously followed and enhanced by many researchers. Shouhong Wang and Norman Archer P (1994) proposed that neural networks, which use the back-propagation learning algorithm under monotonic function constraints, could be used in modeling multiple criteria multiple person decision-making. This is done by training the neural networks with the judgment data of a set of individual decision makers. Shouhong Wang (1994) presented neural network techniques for monotonic nonlinear models. The author used neural network models for monotonic nonlinear regression. He proposed an algorithm for the same. Bill Hardgrave C., Rick Wilson L. and Kent Walstrom A (1994) used neural networks for predicting graduate student success and the decision to accept a student into a graduate program, which is a difficult one. Kate Smith, Palaniswami M. and Krishnamoorthy M (1996) proposed a hybrid neural approach to combinatorial optimization. Both the Hopfield neural network and Kohonen's principles of self-organization have been used to solve difficult optimization problems, with varying degrees of success. Wang Q., Stockton D.J. and Baguley P (2000) works on establishing cost-estimating relationships that appears to offer advantages is that of neural networks as they are 'model-free estimators', hence providing key advantages over traditional techniques. Mehmet Savsar and Hisham Choueiki M (2000) developed a generalized systematic procedure for determining the optimum kanban in just-in-time controlled production lines. Saglam H. and Unuvar A (2003) worked on tool condition monitoring in milling based on cutting forces by a Neural Network. The application of a multilayered Neural Network for tool condition monitoring in face milling is introduced and evaluated against cutting force data. The work uses the back-propagation algorithm for training Neural Network of 5x2x10 architecture. Jih-Gau Juang, Hao-Hsiang Chang and Wu-Ben Chang (2003) worked to devise an intelligent automatic landing system using time delay Neural Network controller and linearised inverse aircraft model to improve the performance of conventional automatic landing systems. Monfared M.A.S. and Yang J.B (2005) worked on multi level intelligent scheduling employing the techniques of artificial neural networks. Current manufacturing scheduling and control systems are incapable of coping with complex system dynamics inherent in real world situations and, hence, human intervention is required to maintain real-time adaptation and optimization. Shobha G.T., Sharma S.C. and Doreswamy (2005) suggested the use of neural networks for knowledge discovery for large data sets. The paper deals with the application of artificial neural network for extracting knowledge of the predicted income for a prospective employee.

III. Problem Definition

The use of Artificial Neural Network model in order to optimize the clustering of requests of the machines in a FMS environment has been explored. The problem that has been taken into consideration revolves around the clustering of the Automated Storage and Retrieval System.

The objective is to optimize the distribution of the job of catering to the needs of the receiving stations in an industry. The different stations in any manufacturing industry place an order in random way to the raw material distribution station and the tool room (tool crib). There cannot be separate equipments to take the goods required to the stations as this is not economically feasible and may also result in excessive traffic. However more than one station may place an order at the same time. Hence, efficient clustering is required to cater to the needs of the workstations effectively.

IV. Methodology

i. Data collection plan - The data required in this effort is the number of orders placed by each work station per unit time, the time between two consecutive orders, the tools and raw materials required, the type of equipment used for catering the need, the various paths and the shortest path and the time required to cater to the needs. This data has been obtained through a set of fictitious data sets.

ii. Artificial neural networks can be used for clustering material handling equipment in FMS environment. In this research paper, the Kohonens Self-Organizing Map is used to train the model using unsupervised learning method. The model is coded in C++.

Kohonens Self-organising map is used to train the Artificial Neural Network to cluster the input data sets. The training is done by continuously changing the learning rate and the weightage given to the inputs. The training process involves millions of iterations which help the model learn the concept of clustering the fictitious data. Once the model has been trained i.e. the obtained output is the same as the desired output; the learning rate and the corresponding weights are fixed. The model is said to provide optimized results.

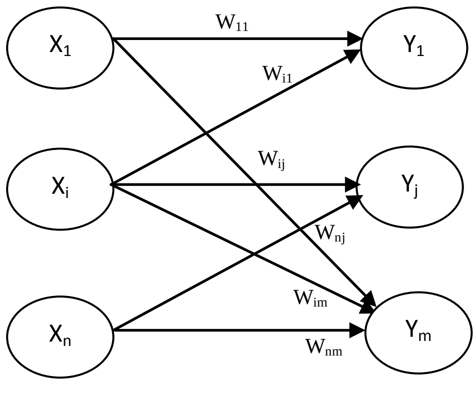
iii. Kohonen Self-Organizing Maps also called as called topology preserving maps, assuming topological structure among the cluster units. Kohonen network aims at using the Kohonen learning to adjust the weights and finally result in a pattern classifier. There are m cluster units arranged in a row of one or two dimensional array. The input signals are arranged as n-tuples. All inputs are given to all the neurons. The weights of the cluster units serves as an exemplar of the input pattern associated with the cluster unit. Kohonen network also follows the “winner-takes-all” policy. The network cluster unit, whose weight vectors matches more closely with the input pattern, is considered the winning neuron. The weights of the winning unit and its neighbouring units are updated. The winning is decided based on the Euclidean distance which is given by,

Euclidean distance, $D = \sqrt{\sum (xi – wij)2}$

The weight adjustment is performed as

$$\Delta wij = \alpha(xi – wij)$$

The architecture of Kohonen Self-Organizing Maps is given below



x_i = input vector w_{ij} = weight $y_j =(x_i-w_{ij})^2$

Fig.1: Architecture of Kohonen Self-Organizing Map

iv. Algorithm

5555\STEP 1: Initialize weights wij .

Set learning rate and neighbourhood parameters.

STEP 2: While stopping condition is false, do steps 3 to 8.

STEP 3: For each input vector, x do steps 4 to 6.

STEP 4: For each j, compute D(j) as :

$D(j) = \sum (x_i - w_{ij})^2$ For i = 1 to n.

STEP 5: Find the index, j such that D(j) is minimum.

STEP 6: For all j

$w_{ij} (new) = w_{ij} (old) + \alpha[x_i - w_{ij}(old)]$

STEP 7: Update learning rate.

STEP 8: Reduce radius of topological neighborhood at specified times.

STEP 9: Test for stopping condition.

Learning rate α is a gradually decreasing function of time. A geometric decrease would also produce similar results. During the clustering process, the radius around a cluster unit is decreased.

V. Results

The requests from the machines are analyzed and the identification of the clusters is done using Artificial Neural Network technique. The input and output matrix is given in the table below.

VI. Conclusion

The model that has been generated here clusters the scattered input data in Automated Storage and Retrieval Systems.

VII. Acknowledgement

I express my sincere gratitude to my guide, all staff members and students of Industrial Engineering and Management for their guidance and support. All Maity for providing the encouragement in preparation of this research paper.

VIII. References

[1] B. Yegnanarayana, Artificial Neural Networks, Prentice Hall of India, 3rd Edition, 2004.

[2] S. Rajasekaran and G.A. Vijayalakshmi Pai, Neural network, Fuzzy logic and Genetic algorithm, Prentice Hall of India, 4th Edition, 2004.

[3] S. N Sivanandam and M. Paulraj, Introduction to Artificial Neural Networks, Vikas Publishing House Pvt. Ltd., 2003 Edition.

[4] Yeshwant Kanetkar, Let Us C, BPB Publications, 3rd Edition, 1999

[5] E. Balaguruswamy, Programming with ANSI C, Tata Mc-Graw Hill Publications, Edition 2.1, 2002.

[6] Laura Burke I. and James Ignizio P., 1992, Neural Networks and Operation Research: An Overview, Computers Ops Res., Vol. 19, No. ¾, pp 179-189.

[7] Chee-Kit Looi, 1992, Neural Networks Methods In Combinatorial Optimization, Computers Ops Res., Vol. 19, No. ¾, pp 191-208.

[8] Johnston M. D. and Adorf H. M., 1992, Scheduling With Neural Networks- The Case Of The Hubble Space Telescope, Computers Ops Res., Vol. 19, No. ¾, pp 209-240.

[9] Shiva Kumar Vaithyanathan and James Ignizio, P., 1992, A Stochastic Neural Network For Resource Constrained Scheduling, Computers Ops Res., Vol. 19, No. ¾, pp 241-254.

[10] Qiwen Wang, Xiaoyun Sun, Bruce Golden L., Lenore DeSilets, Edward Wasil A., Scott Luco and Adam Peck, 1993, A Neural Network Model For The Wire Bonding Process, Computers Ops Res., Vol. 20, No.8, pp 879-888.

[11] Shouhong Wang and Norman Archer P., 1994, A Neural Network Technique In Modeling Multiple Criteria Multiple Person Decision Making, Computers Ops Res., Vol. 21, No. 2, pp 127-142.

[12] Shouhong Wang, 1994, Neural Network Techniques For Monotonic Nonlinear Models, Computers Ops Res., Vol. 21, No. 2, pp 143-154.

[13] Bill Hardgrave C., Rick Wilson L. and Kent Walstrom A., 1994, Predicting Graduate Student Success: A Comparison Of Neural Networks And Traditional Techniques, Computers Ops Res., Vol. 21, No. 3, pp 249-263.

[14] Laura Burke, 1996, "Conscientious" Neural Nets for Tour Construction in the Traveling Salesman Problem: The Vigilant Net, Computers Ops Res., Vol. 23, No. 2, pp 121-129.

[15] Kate Smith, Palaniswami M. and Krishnamoorthy M., 1996, A Hybrid Neural Approach To Combinatorial Optimization, Computers Ops Res., Vol. 23, No. 6, pp 597-610.

[16] Wang Q., Stockton D.J. and Baguley P., Process Cost Modeling Using Neural Networks, 2000, Int. J. Prod. Res., Vol. 38, No. 16, pp 3811-3821.

[17] Saglam H. and Unuvar A., 2003, Tool Condition Monitoring In Milling Based On Cutting Forces By A Neural Network, Int. J. Prod. Res., Vol. 41, No.7, pp 1519-1532.

[18] Jih-Gau Juang, Hao-Hsiang Chang and Wu-Ben Chang, 2003, Intelligent Automatic Landing System Using Time Delay Neural Network Controller, Applied Artificial Intelligence, 17, pp 563-581.

[19] Monfared M.A.S. and Yang J.B., 2005, Multilevel Intelligent Scheduling And Control System For An Automated Flow Shop Manufacturing Environment, International Journal Of Production Research, Vol.43, No.1, pp 147-168.

[20] Shobha G.T., Sharma S.C. and Doreswamy, 2005, Knowledge Discovery For Large Data Sets Using Artificial Neural Network, International Journal Of Innovative Computing, Information And Control, Vol.1, No.4, pp 635-642.

TABLE I

Input Matrix

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
10	0	**5**	0	0	**5**	0	0	0	0	0	0	**4**	0	**9**	0	0	0	**5**	0	0	0	0	0	**7**
7	0	**2**	0	0	**4**	0	0	0	0	0	0	**3**	0	**8**	0	0	0	**6**	0	0	0	0	0	**6**
9	0	**4**	0	0	**6**	0	0	0	0	0	0	**5**	0	**10**	0	0	0	**7**	0	0	0	0	0	**8**
8	0	**3**	0	0	**7**	0	0	0	0	0	0	**6**	0	**11**	0	0	0	**8**	0	0	0	0	0	**9**
0	0	0	**12**	0	0	0	0	**12**	0	0	0	0	0	0	0	0	0	0	0	0	0	**13**	0	0
0	0	0	**16**	0	0	0	0	**17**	0	0	0	0	0	0	0	0	0	0	0	0	0	**18**	0	0
0	0	0	**17**	0	0	0	0	**18**	0	0	0	0	0	0	0	0	0	0	0	0	0	**19**	0	0
0	0	0	**14**	0	0	0	0	**15**	0	0	0	0	0	0	0	0	0	0	0	0	0	**16**	0	0
0	0	0	**13**	0	0	0	0	**14**	0	0	0	0	0	0	0	0	0	0	0	0	0	**15**	0	0
0	0	0	**10**	0	0	0	0	**12**	0	0	0	0	0	0	0	0	0	0	0	0	0	**20**	0	0
0	0	0	0	0	0	**22**	0	0	**23**	0	0	0	0	0	**24**	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	**24**	0	0	**25**	0	0	0	0	0	**26**	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	**28**	0	0	**27**	0	0	0	0	0	**27**	0	0	0	0	0	0	0	0	0
0	**32**	0	0	0	0	0	0	0	0	0	**31**	0	0	0	0	**30**	0	0	0	0	0	0	0	0
0	**30**	0	0	0	0	0	0	0	0	0	**29**	0	0	0	0	**28**	0	0	0	0	0	0	0	0
	31	0	0	0	0	0	0	0	0	0	**30**	0	0	0	0	**29**	0	0	0	0	0	0	0	0
0	0	0	0	**36**	0	0	0	0	0	0	0	0	**34**	0	0	0	0	0	0	**35**	0	0	0	0
0	0	0	0	**37**	0	0	0	0	0	0	0	0	**35**	0	0	0	0	0	0	**36**	0	0	0	0
0	0	0	0	**38**	0	0	0	0	0	0	0	0	**36**	0	0	0	0	0	0	**37**	0	0	0	0
0	0	0	0	0	0	0	**41**	0	0	0	0	0	0	0	0	0	**42**	0	0	0	0	0	**43**	0
0	0	0	0	0	0	0	**40**	0	0	0	0	0	0	0	0	0	**41**	0	0	0	0	0	**42**	0
0	0	0	0	0	0	0	**42**	0	0	0	0	0	0	0	0	0	**43**	0	0	0	0	0	**44**	0
0	0	0	0	0	0	0	0	0	0	**48**	0	0	0	0	0	0	0	0	**46**	0	**47**	0	0	0
0	0	0	0	0	0	0	0	0	0	**49**	0	0	0	0	0	0	0	0	**47**	0	**48**	0	0	0
0	0	0	0	0	0	0	0	0	0	**50**	0	0	0	0	0	0	0	0	**48**	0	**49**	0	0	0

TABLE II

Rearranged Matrix

3	13	6	19	25	1	15	4	9	23	10	7	16	17	12	2	14	21	5	8	18	24	20	22	11
5	**4**	**5**	**5**	**7**	**10**	**9**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	**3**	**4**	**6**	**6**	**7**	**8**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	**5**	**6**	**7**	**8**	**9**	**10**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	**6**	**7**	**8**	**9**	**8**	**11**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	**12**	**12**	**13**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	**16**	**17**	**18**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	**17**	**18**	**19**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	**14**	**15**	**16**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	**13**	**14**	**15**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	**18**	**19**	**20**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	**23**	**22**	**24**	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	**25**	**24**	**26**	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	**27**	**28**	**27**	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	**30**	**31**	**32**	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	**28**	**29**	**30**	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	**29**	**30**	**31**	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**34**	**35**	**36**	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**35**	**36**	**37**	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**36**	**37**	**38**	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**41**	**42**	**43**	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**40**	**41**	**42**	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**42**	**43**	**44**	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**46**	**47**	**48**
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**47**	**48**	**49**
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**48**	**49**	**50**

◈◈◈