

FUZZY CLUSTERING-BASED MODEL FOR PRODUCTIVITY FORECASTING

*Farid Mirahadi¹ and Tarek Zayed²

¹*Concordia University*
1515 St. Catherine W., Room: EV-10.225
Montréal, Canada H3G 2W1
(*Corresponding author: s_miraha@encs.concordia.ca)

¹*Concordia University*
1515 St. Catherine W., Room: EV-6.401
Montréal, Canada H3G 2W1

FUZZY CLUSTERING-BASED MODEL FOR PRODUCTIVITY FORECASTING

ABSTRACT

Forecasting productivity of construction operations is a difficult but crucial task in planning construction projects. Over the past decades, many models have been developed to forecast productivity for different construction operations. Models made up of several functional relations and controlled by a specific number of control rules are more in line with human reasoning and logic. Neural-Network-Driven Fuzzy Reasoning (NNDFR) structure as one of these models shows a great performance for modeling datasets among which clear clusters are recognizable. Lack of the compatibility of conventional NNDFR with fuzzy clustering algorithms besides the insufficient attention paid to the optimization of number of clusters in this model, created a potential area for further research. The main contribution of the proposed model is to develop a modified NNDFR system to model construction data. To this end, Fuzzy C-Means (FCM) algorithm is substituted for K-means in NNDFR structure, and its parameters such as the number of clusters and weighting exponent are optimized through genetic algorithm. The proposed model is further verified through simulation of a construction operation in which several qualitative and quantitative factors are considered. Its implementation to the case study shows a considerable improvement of model performance with lower Mean Squared Error (MSE). The developed model assists researchers and practitioners in utilizing historical construction data to forecast productivity of construction operations with a high accuracy that could not be obtained by traditional techniques.

KEYWORDS

Productivity forecasting, Fuzzy reasoning, Fuzzy clustering, Neural network, Clustering-based model, Genetic algorithm, Multi-dimensional membership function

INTRODUCTION

Fuzzy models and artificial neural network (ANN) systems have provided an effective tool for addressing uncertainty in decision-making. Uncertainty, as the ineradicable nature of construction projects convinced researchers to approach such intelligent systems. In the past few years, these systems were dramatically applied to develop forecasting models in the construction management area (Boussabaine, 1996; Bowena & Edwardsa, 1985; Chan, Chan, & Yeung, 2009; Cheng & Ko., 2003; Kim, An, & Kang, 2004; Leu, Chen, & Yang, 2001; Li, 1995; Martin Skitmore & Thomas Ng, 2003; Moselhi, Hegazy, & Fazio, Potential applications of neural networks in construction, 1992; Tah & Carr, 2000). Productivity estimation of construction operations, as a decision criterion in project planning and control, has become an interesting target for forecasting models.

Integration of basic soft computing techniques such as, neural networks, fuzzy logic and evolutionary algorithms (EAs) empowered researchers to make more efficient forecasting models. In such models, mainly the gaps of each method are treated by the complementary ones. This trend led to creation of a variety of hybrid intelligent structures. Based on the area of study and inherent features of the data, one of those structures showing the best accuracy can be selected as the most appropriate one. One of the structures that has been considered as a notable introductory step to today's innovative neuro-fuzzy systems, is Neural-Network-Driven Fuzzy Reasoning (NNDFR). With the help of self-learning ability of ANN, this model succeeded to present a state-of-the-art method for fuzzification and inference procedure to a conventional fuzzy system. This capability, besides emerging tendencies to the application of cluster-

based fuzzy and neural network systems led us to scrutinize the benefits and potential constructive modifications of NNDFR system. The objectives of the current research can be summarized as follows:

1. Develop a modified NNDFR system to model construction data
2. Optimize the NNDFR model parameters, such as number of clusters and weighting exponent

LITERATURE REVIEW

Simulation is one of the most widely used techniques in operational research and managerial science (Law & Kelton, 2000). Construction simulation is a powerful tool that can be used by a construction company for several tasks, such as productivity measurement, risk analysis, resource planning, and the design and analysis of construction methods (Sawhney, AbouRizk, & Halpin, Construction project simulation using CYCLONE, 1998). Among all these applications, productivity measurement might be considered as the most important factor in construction planning and control. Most research works in construction simulation have mainly focused on simulation modeling with little emphases on studying the qualitative variables that affect the simulation process itself (Elwakil, 2011). Although some studies have investigated the effect of variables, such as weather (Moselhi, Gong, & El-Rayes, Estimating weather impact on the duration of construction activities, 1997), on productivity of construction processes, there is still a lack of research on this area. In most cases, there is no pre-identified functional relation between variables affecting operational level of simulation process and their outputs. This matter convinced researchers to utilize mathematical models like ANN that is used to model complex relationships between a set of dependent and independent variables.

ANN tries to simulate the structure and operation of human neural network system. An ANN is comprised of an interconnected set of artificial neurons operating based on a connectionist approach to computation. The ability of learning by examples turned this technique to a very useful tool in data modeling (Lawrence, 1994). This technique can develop a predictive model where the relationships between inputs and outputs are not sufficiently known. Patterns and relationships of historical data are recognized to acquire the “knowledge” used to predict unknown output values for a given set of input values (Sawhney & Mund, Adaptive probabilistic neural network-based crane type selection system, 2002). In the past few decades, ANN dramatically was applied to develop forecasting models in construction management area (Boussabaine, 1996; Kim, An, & Kang, 2004; Li, 1995; Moselhi, Hegazy, & Fazio, Potential applications of neural networks in construction, 1992). These models are well suited to the problems where the underlying reasons and quality of input-output relations are not studied. Fuzzy logic was introduced as a response to the need of a systematic reasoning more in line with the human logic. In this approach, unlike ANN, interpretability of the inference procedure is the center of attention. Construction management following other areas of research has applied this concept in different areas, such as risk analysis, construction simulation and etc. (Carr & Tah., 2001; Zhang, Tam, & Shi, 2002). As these researches and many studies alike show, the empirical approach of acquiring membership functions had been remained as the only method of design for a long period of time. Lack of a definite method to design membership functions was resolved with the integration of ANN and fuzzy logic which lets membership functions be determined by self-learning ability of ANN. Adaptive Neuro-Fuzzy Inference System (ANFIS) is a well-known example of these neuro-fuzzy systems. Although ANFIS is more flexible and robust than the more traditional modeling systems, it only supports the singleton and linear output membership functions.

Clustering is considered as one of the most important and frequently used techniques in data analysis (Beringer & Hüllermeier, 2006). Data clustering is the task of organizing a data set into different groups, such that the objects of the same cluster are more similar to each other compared to those in other clusters. Fuzzy C-Means (FCM) (developed by Dunn in 1973; improved by Bezdek in 1981) clustering can be considered as one the most dominant algorithms in both theoretical and practical applications of data mining. FCM possesses a fuzzy approach for reporting the memberships to different clusters. It allows each data point to belong to more than one cluster. Many of the clustering algorithms are based on

knowing the number of clusters beforehand. FCM algorithm is of that type and therefore requires this initial value before clustering has been accomplished. Clustering techniques extensively have enhanced hybrid intelligent systems. As a well-known example, Takagi-Sugeno (TS) type fuzzy models define different regions in the data space each of which represents a linear input-output mapping (Takagi & Sugeno, 1985). It is prevalent in TS models that an automatic method, like fuzzy clustering, is exploited to attain candidates for linear regions (Jantzen, 1998). Takagi and Hayashi (1991) utilized clustering techniques to partition the inference rules for a Neural-Network-Driven Fuzzy Reasoning model (NNDFR). In this model, the data is grouped by K-Means and the number of inference rules is equal to the number of clusters.

NNDFR (Takagi & Hayashi, 1991) was the first application of neural networks in self-regulating design of membership functions (Jain & Jain, 1997). This approach forms a nonlinear multi-dimensional membership function, which internally combines all the fuzzy variables via clustering algorithms. Proposed model is beneficial to the datasets among which clear clusters are recognizable. Compared to the ANFIS that only accepts constant and linear functions as consequent relationships, NNDFR inference part is typically constituted from separately trained neural networks. Multi-dimensional membership functions enable the model to consider the interdependence of the input variables, which is regarded as the main improvement of NNDFR over the conventional fuzzy reasoning frameworks. The main drawbacks of this model are: 1) Lack of compatibility with fuzzy clustering algorithms, which enables the user to regulate fuzziness of membership functions. 2) Disregarding the optimization of the number of clusters, which could notably distort the fitness of resulted model.

Each type of hybrid intelligent systems has its own parameters and structure, which can extensively alter its outcome. Those parameters usually are attained through an optimization driven by historical data. Consequently, Evolutionary Algorithms (EAs) have been turned into one of the basic components in the structure of many hybrid intelligent systems. Genetic Algorithm (GA) initiated by John Holland in 1975, is considered as the most well-known EA. In GA, an initial population of randomly generated individuals is produced and it evolves toward better generations by altering and mutating the properties of the population. This EA, which mimics the natural evolution process, involves following main steps (Haupt & Haupt, 2004): 1) Generate initial population. 2) Evaluate fitness of population. 3) Selection. 4) Crossover. 5) Mutation. 6) Generate new population and evaluate fitness. This evolutionary process allows fine-tuning of many hybrid intelligent structures, such as neuro-fuzzy systems, and subsequently enables the model to show a higher level of performance.

PROPOSED FRAMEWORK

Introduction to NN-driven Fuzzy Reasoning

In this section, first the structure of NNDFR is analyzed to detect the potential areas of improvement. The design procedure of NNDFR could be summarized in three steps: clustering the training dataset, training the membership neural networks (NN_{mem}), training the consequent neural network (NN_{1-k}) of each cluster. In the first step, input data space is partitioned to hard clusters. In this fuzzy system, the number of rules equals the number of clusters. In the second step, NN_{mem} is trained between each input vector and its corresponding cluster assignment vector, as illustrated in figure 1. For example, the supervised part of the learning process for a vector which belongs to cluster 3 is (0,0,1). In the third step, the consequent neural networks are trained between the members of each cluster, previously partitioned in the first step, and their corresponding outputs.

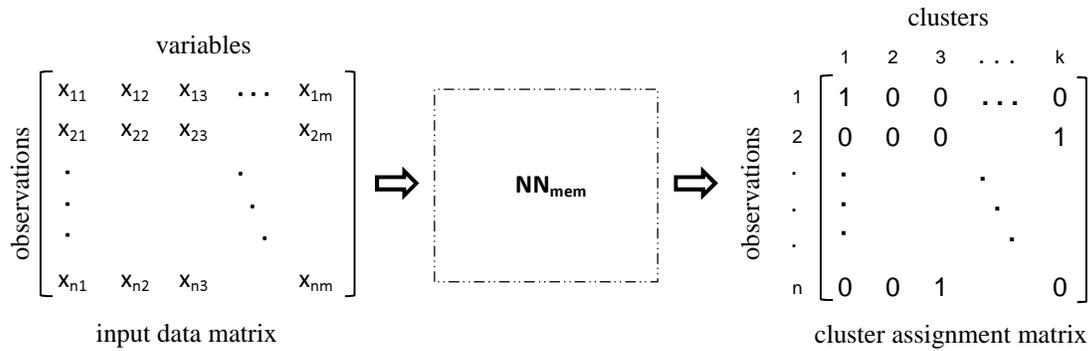


Figure 1 - Second step, training the membership neural network. m , number of input variables; n , number of observations; k , number of clusters

Figure 2 shows a holistic diagram of the NNDFR. The NN_{mem} generates the membership functions of the premise (IF) parts of the rules and NN_{1-k} prepare consequent input-output relationships (THEN parts). This system calculates the final estimated output based on a weighted average of the output of THEN parts, such that the weights are the membership values produced by NN_{mem} .

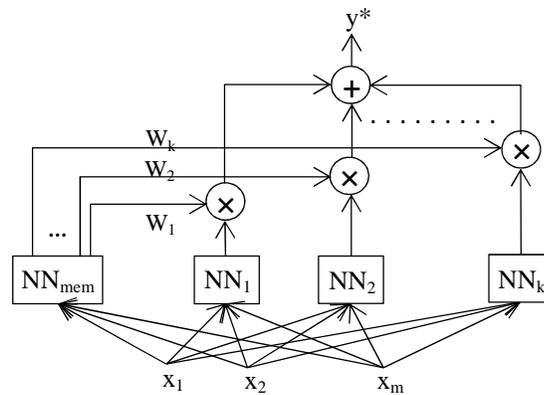


Figure 2 - The structure of NNDFR, where W is the membership value and y^* is final estimated output (Takagi & Hayashi, 1991)

The resulting fuzzy model is expressed by the following (Takagi & Hayashi, 1991):

- Rule 1: IF $x = (x_1, x_2, \dots, x_m)$ is C_1 , THEN $y_1 = NN_1(x_1, x_2, \dots, x_m)$
- Rule 2: IF $x = (x_1, x_2, \dots, x_m)$ is C_2 , THEN $y_2 = NN_2(x_1, x_2, \dots, x_m)$
- ...
- Rule K : IF $x = (x_1, x_2, \dots, x_m)$ is C_K , THEN $y_k = NN_k(x_1, x_2, \dots, x_m)$

Where C_{1-k} denote existing clusters. The final estimated value can be delivered through equation 1 (Takagi & Hayashi, 1991).

$$y_i^* = \frac{\sum_{s=1}^k W_s(x_i) \cdot y_s(x_i)}{\sum_{s=1}^k W_s(x_i)} \quad (1)$$

Modified NNDFR

The above-mentioned system utilizes a hard clustering algorithm, such as K-Means, in order to determine and assign the membership values. It then builds a hyper surface membership function by means of pattern recognition ability of ANN, as described in the second step of the NNDFR design procedure. This innovative fuzzification approach automatically considers the interdependence of input variables, which has been the definite weakness of conventional fuzzification process. In other words, this NN-driven hyper-surface membership function combines all 2D membership functions of different variables and results in a single multi-dimensional membership function. However, the NN_{mem} concluded from the results of a hard clustering algorithm provides us a fuzziness which is not under control and capable of being regulated. The supervised part of the learning process does not have enough flexibility and it only accepts the values 0 and 1. Thus, we cannot decide about the shape of the membership functions and modify the fuzziness assigned to each point (Figure 3). In other words, the system has the same attitude towards all of the points in the same cluster and does not involve the distance from the centroids in making decisions.

Applying FCM algorithm as a substitution for K-Means gives a fuzzy approach to membership assignment procedure. As will be explained in the following, FCM formula has an initial parameter, which regulates the relative weights of the membership values to all existing clusters and consequently controls the fuzziness of subsequent membership functions. In that case, the model is able to compute the final estimated outcome based on an optimized contribution of all consequent relationships. FCM allows each data point to belong to more than one cluster. This iterative algorithm is performed through minimization of following objective function (Matteucci, 2006):

$$J_m = \sum_{i=1}^N \sum_{j=1}^C u_{ij}^m \|x_i - c_j\|^2 \quad (2)$$

Where u_{ij} is the degree of membership of x_i to the cluster j , m is any real number greater than 1, x_i is the i^{th} measured data, c_j is the center of the j^{th} cluster, and $\|*\|$ is a kind of distance between the d -dimensional vector x_i and d -dimensional vector c_j .

The iterations of optimizing above-mentioned objective function proceeds while u_{ij} and c_j are updating in each step by (Matteucci, 2006):

$$u_{ij} = \frac{1}{\sum_{k=1}^C \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}} \quad (3)$$

$$c_j = \frac{\sum_{i=1}^N u_{ij}^m \cdot x_i}{\sum_{i=1}^N u_{ij}^m} \quad (4)$$

Once $\max_{ij} \left\{ \left| u_{ij}^{(k+1)} - u_{ij}^{(k)} \right| \right\} < \varepsilon$ ($0 < \varepsilon < 1$) is satisfied, the iterations will stop; where k is the iteration step.

The two main problems that we face while placing FCM in NNDFR structure are as follows:

1. Separate Training Sets

Duo to the fuzzy nature of FCM, each data point belongs to all clusters, but with different degrees. However, consequent neural networks (NN_{1-k}) need separate training data sets. To mitigate this problem, we propose a way in which each data point will be tagged to one cluster with the maximum membership

value. For example, point A with the membership values of (0.15 0.2 0.65) will be separated as a member of NN_3 training data set.

2. Fuzzifier and Number of Clusters

The weighting exponent m , which is called fuzzifier, can greatly influence the performance of FCM. When the fuzzifier is close to 1, the result of FCM is identical to that of k-means. When the fuzzifier approaches infinity, each cluster is only assigned to its centroid and has a membership value of 0 in the rest of the points. Therefore, FCM regulates the fuzziness of the clusters by adjusting this weighting exponent. Figure 3 presents an illustration of membership neural networks trained by the FCM algorithm of different fuzzifiers. In this example, some random data points in 2D (X_1, X_2) data space are generated such that they can be clearly partitioned into two well-separated clusters. After the application of FCM on generated data points, NN_{mem} is trained between the (X_1, X_2) coordinates and corresponding FCM membership outputs. By simulation of enough number of input points within this area and plotting the outputs against corresponding (X_1, X_2) coordinates, following surfaces will appear. As is shown, where the m is very close to 1, FCM acts as K-Means and except a narrow overlapping area between clusters, all data points take the membership value of almost 1 to one cluster and 0 to the others. It means that mostly, only one consequent neural network is in effect. However, in case of using any higher values of m , only the center of each cluster is solely assigned to that cluster and corresponding consequent neural network. In that case, any deviation from the center of cluster will result in activation of other consequent relationships.

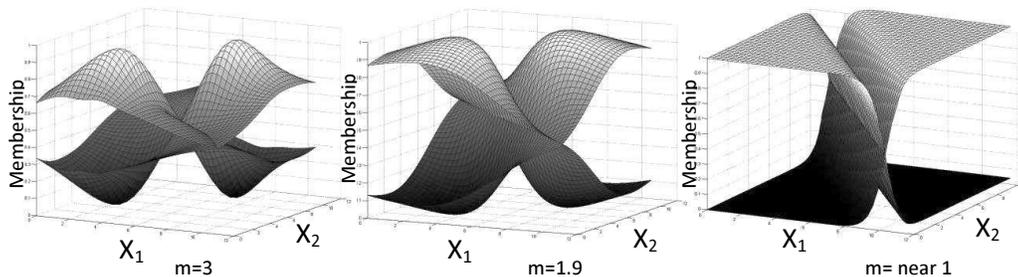


Figure 3 - Multi-dimensional membership function of two random variables (X_1, X_2) for different values of fuzzifier (m)

Hence, the desirable shape of the membership function could be reached via fine-tuning the fuzzifier. However, there is still no robust theoretical way of finding the optimal value for this exponent. It seems that determination of the fittest value is highly dependent on the characteristics of the problem. Here, an optimization process in which the performance of NNDFR is set as the target function can deliver the optimal value for the fuzzifier.

The initial value for the number of clusters is the other parameter that must be determined before clustering starts. Considering that in NNDFR the number of clusters defines the number of rules, an improper determination of the initial value can significantly distort the fitness of the model. Thus, an optimization on this parameter can make a reasonable compromise between complexity and efficiency of the model. As is obvious, separate optimizations of these two parameters cannot lead us to a comprehensive evaluation. The best solution should consider all the possible combinations of these two values.

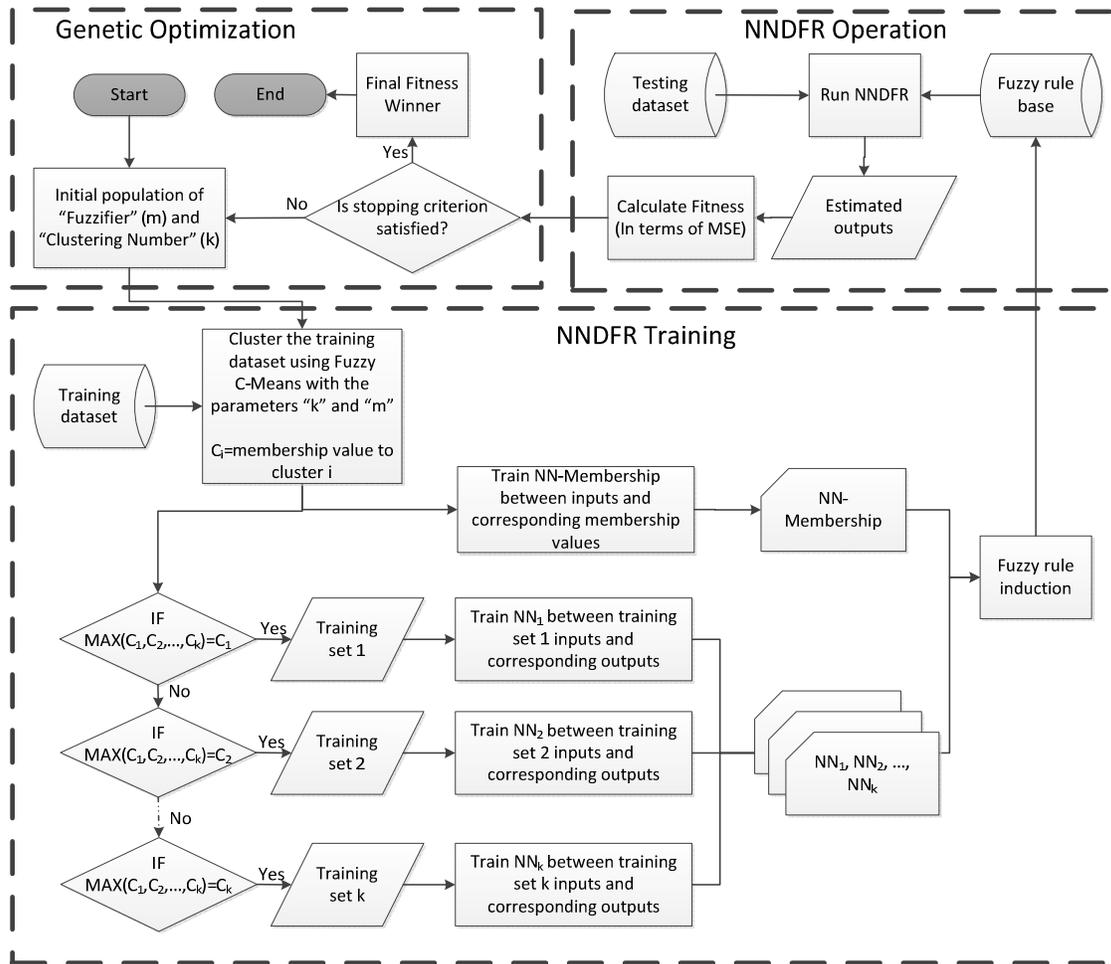


Figure 4 - Methodology flowchart

Optimization

Simultaneous optimization of the two parameters, "Fuzzifier" and "Clustering Number", can be reached through application of genetic optimization. In this optimization procedure, each individual (chromosome) should be made up of two parameters (gen) respectively "Fuzzifier" and "Clustering Number". Iterative cycles of GA then repeat until the termination condition is satisfied. Here, we consider a specific number of generations as the stopping criterion. Fitness function that evaluates the appropriateness of each individual is set to the MSE of NNDFR system. Once the maximum number of generations is met, the highest ranked individual is adopted as the final solution. Figure 4 shows the entire flowchart of methodology for this research.

CASE STUDY

This case study analyses the data gathered from construction processes of Engineering, Computer Science and Visual Arts complex of Concordia University. The data set consists of several quantitative and qualitative variables affecting concrete pouring operations and their corresponding daily productivity. Nine factors are considered: Temperature, Humidity, Precipitation, Wind Speed, Gang Size, Labor Percentage, Work Type, Floor Level and Work Method. These input variables are elaborated in Table 1. Following, Table 2 presents a small sample of these 131 record points.

Table 1: Concrete pouring process variables

Variables	Description
1 Temperature (°C)	Daily average of eight working hours
2 Humidity (%)	Daily average of eight working hours
3 Precipitation	Reported in terms of four numerical values: No precipitation = 0, Light rain = 1, Rain = 2, and Snow = 3
4 Wind Speed (km/h)	Daily average of eight working hours
5 Gang Size (workers)	Number of people in the gang
6 Labor Percentage (%)	The percentage of the labor (non-skilled workers) in the gang
7 Work Type	Reported in terms of activity type: Slabs= 1 and Walls = 2
8 Floor Level	The floor number
9 Work Method	Crane and bucket arrangement=1 and Pumping=2
10 Daily Productivity (m ³ /man/hour)	Total cubic meters done during the day divided by the gang size and working hours

Table 2: A sample of concret pouring data

Temperature (°C)	Humidity (%)	Precipitation	Wind Speed (km/h)	Gang Size (workers)	Labor Percentage (%)	Work Type	Floor Level	Work Method	Daily Productivity (m ³ /man/hour)
-8	87	2	14.2	22	36	1	3	1	1.27
-6	37	0	19.9	19	33	1	8	2	1.23
25	77	0	24	20	30	1	14	1	1.65

We randomly divide our collection of 131 data points to two fractions of 118 and 13. As the model runs, GA creates different individuals formed by “Clustering Number” and “Fuzzifier” and then transfers them to FCM part of the NNDFR in order to initialize its parameters. The bigger fraction of the data collection, which is considered as the training dataset, trains the neural networks and builds the appropriate structure of the NNDFR with respect to each particular individual. The other fraction named testing dataset subsequently is provided to the prepared model and evaluates the model performance in terms of MSE. This procedure continues until the last generation is met and the fittest answer is selected.

The parameters of the GA used in this framework are:

- Fitness function: MSE of NNDFR output
- Population size: 30
- Number of generations: 20
- Selection function: *Stochastic Uniform*
- Elite count: 2
- Crossover fraction: 0.8
- Crossover function: *Intermediate Crossover* (Ratio=1)
- Mutation function: *Adaptive Feasible*
- Search bounds: Clustering Number: [2,10]; Fuzzifier: (1,3)

Table 3 tabulates the winners of each generation based on the MSE, which has been our fitness criterion. Figure 5 plots the best and mean penalty of different generations during the evolution process. The algorithm is terminated by reaching the maximum number of generations proposing 3 number of clusters and a fuzzifier of 1.2513. These optimized values result in the best fitness in terms of MSE, which is improved by 52 percent compared to the winner of first generation. The final winner with $m=1.2513$ implies that a typical NNDFR structure made up of K-Means algorithm ($m=1$) could not deliver the best possible solution.

Table 3 – Genetic Optimization Result

Generations	Individuals		MSE
	Clustering Number	Fuzzifier	
1	3	1.8516	0.01954
2	4	1.3957	0.01949
3	4	1.3957	0.01949
4	3	1.7288	0.01734
5	3	1.4151	0.01484
6	3	1.4151	0.01484
7	3	1.4151	0.01484
8	3	1.4151	0.01484
9	3	1.4151	0.01484
10	3	1.4687	0.01221
11	3	1.4687	0.01221
12	3	1.5054	0.00976
13	3	1.5054	0.00976
14	3	1.2513	0.00932
15	3	1.2513	0.00932
16	3	1.2513	0.00932
17	3	1.2513	0.00932
18	3	1.2513	0.00932
19	3	1.2513	0.00932
20	3	1.2513	0.00932

CONCLUSION

The current research presents a framework in which the conventional structure of NNDFR is improved through substituting K-Means algorithm with FCM. In this innovative structure, the number of clusters and weighting exponent of FCM are two parameters that highly affect the accuracy of modeling. These two parameters should be optimized through a simultaneous optimization process that considers all possible combinations of them. To this end, genetic algorithm is selected to perform the optimization by mimicking natural evolution process. Different models with different parameters are developed such that their estimation performance in terms of MSE defines the fitness function for genetic algorithm. The proposed model is further verified through simulation of a construction operation in which several qualitative and quantitative factors are considered. Empirical results show that the model performance is improved by 52 percent during the procedure of optimization. The developed research helps researchers and practitioners by providing them with an effective way of construction modeling where the structure of model is fine-tuned based on the inherent features of the data.

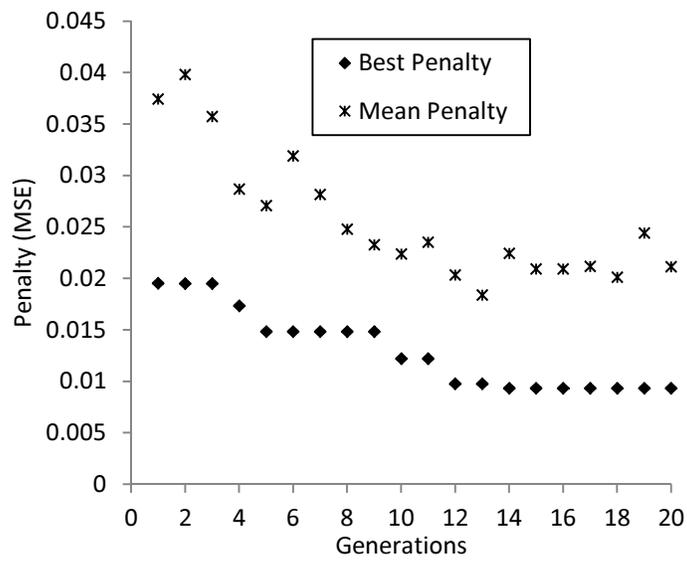


Figure 5 – Best/Mean penalty against generations

REFERENCES

- Beringer, J., & Hüllermeier, E. (2006). Online clustering of parallel data streams . *Data & Knowledge Engineering*, 58(2), 180-204.
- Bezdek, J. C. (1981). *Pattern recognition with fuzzy objective function algorithms*. New York, NY, USA: Plenum Press.
- Boussabaine, A. H. (1996). The use of artificial neural networks in construction management: a review. *Construction Management and Economics*, 14(5), 427-436.
- Bowena, P. A., & Edwardsa, P. J. (1985). Cost modelling and price forecasting: practice and theory in perspective. *Construction Management and Economics*, 3(3), 199-215.
- Carr, V., & Tah., J. H. (2001). A fuzzy approach to construction project risk assessment and analysis: construction project risk management system. *Advances in Engineering software*, 32(10), 847-857.
- Chan, A. P., Chan, D. W., & Yeung, J. F. (2009). Overview of the application of “fuzzy techniques” in construction management research. *Journal of Construction Engineering and Management*, 135(11), 1241-1252.
- Cheng, M. Y., & Ko., C. H. (2003). Object-oriented evolutionary fuzzy neural inference system for construction management. *Journal of Construction Engineering and Management*, 129(4), 461-469.
- Dunn, J. C. (1973). A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters. *Journal of Cybernetics*, 3(3), 32–57.
- Elwakil, E. (2011). *Knowledge Discovery Based Simulation System in Construction*. Doctoral dissertation, Concordia University.
- Haupt, R. L., & Haupt, S. E. (2004). *Practical Genetic Algorithms*. New York, NY, USA: John Wiley & Sons.

- Holland, J. H. (1975). *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. Ann Arbor, MI, USA: University of Michigan Press.
- Jain, L. C., & Jain, R. K. (1997). *Hybrid Intelligent Engineering Systems*. World Scientific.
- Jantzen, J. (1998). *Neurofuzzy modelling*. Technical University of Denmark, Department of Automation. Lyngby: Denmark.
- Kim, G. H., An, S. H., & Kang, K. I. (2004). Comparison of construction cost estimating models based on regression analysis, neural networks, and case-based reasoning. *Building and Environment*, 39(10), 1235–1242.
- Kim, G.-H., An, S.-H., & Kang, K.-I. (2004). Comparison of construction cost estimating models based on regression analysis, neural networks, and case-based reasoning. *Building and Environment*, 39(10), 1235–1242.
- Law, A. M., & Kelton, W. D. (2000). *Simulation modeling and analysis* (3rd ed.). McGraw-Hill.
- Lawrence, J. (1994). *Introduction to neural networks: design, theory, and applications*. California Scientific Software.
- Leu, S. S., Chen, A. T., & Yang, C. H. (2001). A GA-based fuzzy optimal model for construction time–cost trade-off. *International Journal of Project Management*, 19(1), 47-58.
- Li, H. (1995). Neural networks for construction cost estimation. *Building Research & Information*, 23(5), 279-284.
- Martin Skitmore, R., & Thomas Ng, S. (2003). Forecast models for actual construction time and cost. *Building and Environment*, 38(8), 1075–1083.
- Matteucci, M. (2006). *A tutorial on clustering algorithms*. Politecnico di Milano, Dipartimento di Elettronica e Informazione, Como, Como, Italy.
- Moselhi, O., Gong, D., & El-Rayes, K. (1997). Estimating weather impact on the duration of construction activities. *Canadian Journal of Civil Engineering*, 24(3), 359-366.
- Moselhi, O., Hegazy, T., & Fazio, P. (1992). Potential applications of neural networks in construction. *Canadian Journal of Civil Engineering*, 19(3), 521-529.
- Sawhney, A., & Mund, A. (2002). Adaptive probabilistic neural network-based crane type selection system. *Journal of construction engineering and management*, 128(3), 265-273.
- Sawhney, A., AbouRizk, S. M., & Halpin, D. W. (1998). Construction project simulation using CYCLONE. *Canadian Journal of Civil Engineering*, 25(1), 16-25.
- Tah, J. H., & Carr, V. (2000). A proposal for construction project risk assessment using fuzzy logic. *Construction Management & Economics*, 18(4), 491-500.
- Takagi, H., & Hayashi, I. (1991). NN-driven fuzzy reasoning. *International Journal of Approximate Reasoning*, 5(3), 191-212.
- Takagi, T., & Sugeno, M. (1985). Fuzzy identification of systems and its applications to modeling and control. *Systems, Man and Cybernetics, IEEE Transactions on*, 15(1), 116-132.
- Zhang, H., Tam, C. M., & Shi, J. J. (2002). Application of fuzzy logic to simulation for construction operations. *Journal of computing in civil engineering*, 17(1), 38-45.