Traffic control optimization on isolated intersection using fuzzy neural network and modified particle swarm optimization

To cite this article: N Angraeni et al 2019 J. Phys.: Conf. Ser. 1321 032023

View the article online for updates and enhancements.
Traffic control optimization on isolated intersection using fuzzy neural network and modified particle swarm optimization

N Angraeni*, M A Muslim and A T Putra

Computer Sciences Department, Universitas Negeri Semarang, Indonesia

*Corresponding author: nearly.angraeni@gmail.com

Abstract. Traffic density in big cities due to congestion problems in various points of the city. This problem will occur worse at crucial times such as when rush hours and active working hours. The existence of a traffic light system as a traffic signalling device is a solution to overcome traffic congestion. Appropriate traffic light settings can minimize vehicle waiting times at intersections. The aim of this study is to optimize an adaptive traffic control that can adjust the conditions of traffic flow on certain road segments at isolated intersections. In this study optimization uses methods of Fuzzy Neural Network (FNN) and Modified Particle Swarm Optimization (MPSO). The optimization results will be compared with a regular method of Adaptive Neural Fuzzy Inference System without using MPSO. The simulation results show that the efficiency and adaptability of the combination method of FNN and MPSO are better than the Neural Fuzzy Controller without MPSO. A better result is also indicated by the value of Mean Squared Error (MSE) that decreased from 6.3299 becomes 2.065.

1. Introduction

Urban traffic congestion is critical problem in many countries and becoming a major concern in the field of management of transportation system. The existing method for traffic control management is not adequately efficient in term of the performance, cost and the effort for maintenance and support [1]. The congestion have happened to be more complex so there is no exact model of traffic flow due to its high complexity, chaotic behaviors and unique characteristic to local condition [2]. There are two types of control methods of traffic signal include traditional and intelligent control [3]. Generally, traditional control method work by giving duration of red or green phase based on the traffic pattern in each time condition at the intersection. The traffic designed with the preset cycle time without in-depth analysis analysis of traffic condition and it cannot satisfy the real time condition. The traffic light should keep green phase more to complete cycle time.

The fuzzy control strategy proposed in for the traffic controller applied to isolated intersection is extension green time as an output [4]. Determination of green light duration can be done with a formula based on historical data on traffic condition, or with the methodology of artificial intelligence. Fuzzy logic is a component of soft computing which is a fundamental methodology to process linguistic information to deal with uncertainty and imprecision [5]. Fuzzy logic algorithms have started to be applied in traffic control systems that allow describing and simulating of complex systems behavior, which can be hardly achieved by means of mathematical models. However that does
not mean that the fuzzy logic system has no weaknesses. One of the disadvantages of a fuzzy system is cannot learn or adapt its knowledge. So it takes an artificial neural network method to complete it [6].

On the other hand to optimize the fuzzy neural network system, the Particle Swarm Optimization (PSO) method was used in this study with some modification. Basically, the PSO algorithm is included in a swarm intelligence algorithm where each individual in the population can exchange information and experiences of previous learning outcomes. Therefore, the aim of this study is improving the optimization of traffic control on isolated intersection by implementing combination method between fuzzy neural network and modified PSO.

2. Methods
In this research there are two data inputs, the first is vehicle waiting time on the intersection which is symbolized by $W_t$ and the second is vehicle queue length on the intersection which is symbolized by $Q$. Data generation is carried out from rule-base an adaptive neuro-fuzzy inference system method and tree diagram. From generate process produces 2550 input-output sample data. The training process is carried out by modifying the PSO using Constriction Factor Method (CFM) in the FNN model [11]. The results of the FNN data training with PSO were compared with the FNN training without PSO to see how much the optimization level was generated.

2.1. Fuzzy Neural Network (FNN)
The FNN algorithm is a combined method Artificial Neural Network (ANN) and the fuzzy logic system. A fuzzy system can give an information where is the knowledge from it encodes but cannot learn or adapt its knowledge from training examples, so the problem of fuzzy model are lack of the flexibility in decision making. While the neural network can learn and decide the result with the training process but cannot explain more what it has learn. Expert systems are computer-based system that uses knowledge, factual data and reasoning techniques in solving the problems that usually can only solved by a real human expert [14]. Neuro-fuzzy characteristics are accurate learning and adaptive abilities of the neural network along with the generalization and rapid learning abilities of fuzzy logic system [3]. The fuzzy neural network used in this study is a neural network consisting of five layers as shown in Figure 1.

![FNN Model of Traffic Control](image)

**Figure 1.** FNN Model of Traffic Control

In generally the FNN algorithm as follows:
1. The first layer is input layer. Input to first layer are the vehicle waiting time ($W_t$) and the vehicle queue length ($Q$).
2. The second layer is fuzzification. Each node will represents the membership function. The node function is Gaussian function that given in Eq. (1).

$$F(x; \sigma c) = e^{-\frac{(x-c)^2}{2\sigma^2}}$$  \hspace{1cm} (1)
Where \( c \) is the constant of the function. The output of each nodes value of this function is in range [0, 1] and represent to the membership grade of the input.

3. The third layer corresponds to the rule base. As the characteristics of fuzzy rules which are the relationships between the antecedent node is connected to the consequent node corresponds to a single Sugeno-type fuzzy rule. The output of neurons \( i \) in this layer is obtained using Eq. (2).

\[
y_i^{(3)} = \prod_{j=1}^{k} x_{ji}^{3}
\]  
(2)

Where \( x_{ji}^{3} \) are the inputs in this layer and \( y_i^{(3)} \) means the output of the rule neuron \( i \) in this layer.

4. The forth layer corresponds to the rule consequents. The output of neuron \( i \) in this layer shown in Eq. (3).

\[
y_i^{(4)} = \frac{x_i^{(4)}}{\sum_{k=1}^{l} x_{ki}^{4}}
\]  
(3)

Where \( x_{ji}^{4} \) is the input neuron \( j \) located in layer 3 to neuron \( i \) in layer 4.

5. The fifth layer is defuzzification layer. In this layer the node represent the output linguistic variable.

2.2. Particle Swarm Optimization (PSO)

The PSO algorithm is a optimization method by using a population to find the optimal solution using particles of the population it self. The algorithm basically developed through on the behavior of a flock of birds or fish [7]. The swarm is assumed to have a certain size with the initial position of each particle in a random location. In this case the factors that give character to the particle status in the search space called with the particle position and the particle velocity [8]. Mathematical formulations that describe the position and speed of a particle in a given space dimension shown in Eq. (4) and Eq. (5).

\[
X_i(r) = x_{i1}(r), x_{i2}(r), ..., x_{in}(r)
\]  
(4)

\[
V_i(r) = v_{i1}(r), v_{i2}(r), ..., v(r)
\]  
(5)

Each particle is treated like a point that searches for an optimal solution in a search space dimension. Namely by each particle making adjustments to the best position or called the local best and also making adjustments to the best position of all swarms or called local best [12]. By using local best and global best then update status of each particle using the equation shown in Eq. (6) and Eq. (7).

\[
V_i(t) = V_i(t-1) + c_1 r_1 (X_i^*(t) - X_i(t+1)) + c_2 r_2 (X_i^G - X_i(t+1))
\]  
(6)

\[
X_i(t) = V_i(t) + X_i(t+1)
\]  
(7)

2.3. Modified Particle Swarm Optimization

The PSO algorithms are often applied to overcome optimization problems and produce more optimal weight as an alternative to backpropagation in artificial neural network research. Generally, the procedure performed on this method is almost the same as the classic PSO method. In this research modifications used Constriction Factor Method (CFM) to guarantee the convergence and oscillation of particle amplitude decreases over time without maximum speed regulation [10]. The most desirable of these enhancements and modifications is to guarantee a trace in the PSO algorithm so that the coverage is faster. The Constriction factor formulated on equation that shown in Eq. (8).
The mathematical formula for speed on modified PSO can be determined using equation that shown in Eq. (9).

\[ v_{k+1}^i = C \{ v_k^i + c_1 \cdot \text{rand} \cdot (p_k^i - x_k^i) + c_2 \cdot \text{rand} \cdot (p_k^g - x_k^i) \} \] (9)

Where the parameter \( C = C_1 + C_2 \) and \( \Phi > 4 \) so to meet this requirement, the \( c_1 \) and \( c_2 \) values are usually 2.05. The flowchart of modified PSO is illustrated in Fig. 1. While the procedures of the algorithm are [9]: (1) generate initial position of a quantity of particles and their initial velocity randomly; (2) evaluation of the fitness value of each particle based on its position; (3) specify particles with the best fitness value and set it as global best; (4) update the velocity and then with the updated velocity, update the position of each particle; (5) make a reevaluation of each particle and determine the particle with the best fitness value and set it as global best. This is done by comparing current position with local best from the previous iteration; (6) check stopping criteria, if appropriate it stops but if it doesn't return to step 1 [8]. The Process of PSO algorithm is explained briefly in Figure 2.

**Figure 2.** Flowchart of Modified PSO Process

### 3. Result and Discussions

This research conducted by implementing the methods using MATLAB. There are two variables input fuzzy to evaluate the possibility of green phase extend as an output, namely vehicle waiting time (Wt) and vehicle queue length (Q). The vehicle waiting time has a range between 0 and 50 seconds corresponding to Gaussian membership function. Each of membership function has a standard derivation of 2 with the constant are 0 second, 10 seconds, 20 seconds, 30 seconds, 40 seconds. The input membership function of vehicle waiting time is shown in Figure 3.a. While for input queue length (Q) assumed that has arrange between 0 and 50 vehicles which corresponding to Gaussian membership function. The fuzzy sets for Wt consist of VS (for Very Short), S (for Short), L (for Long), VL (for Very Long), and EL (for Extremely Long). Each of membership function has a standard derivation of 2 with the constant are 0 vehicle, 10 vehicles, 20 vehicles, 30 vehicles, 40 vehicles. The input membership function of vehicle queue length is shown in Figure 3.b.
The output extension green time has fuzzy sets that consist of Z (for Zero), S (for Short), L (for Long), VL (Very Long), EL (Extremely Long). From 25 fuzzy rules obtained 2550 sample data which is divided into training data used to find the best model of fuzzy neural and test data used to evaluate accuracy of the model. Some sample data used from generated process of rules 1 in the fuzzy rules based are shown in Table 1.

| Table 1. Some Sample Data from R1 |
|-----------------------------------|
| Input (Wt) | Input (Q) | Output (Ex) |
| 0          | 0          | 0           |
| 0          | 1          | 0           |
| 0          | 2          | 0           |
| 0          | 3          | 0           |
| 0          | 4          | 0           |
| 0          | 5          | 1           |
| 0          | 6          | 1           |
| 0          | 7          | 2           |
| 0          | 8          | 2           |
| 0          | 9          | 2           |
| 1          | 0          | 0           |
| 1          | 1          | 0           |
| 1          | 2          | 0           |
| 1          | 3          | 0           |
| 1          | 4          | 0           |
| 1          | 5          | 1           |
| 1          | 6          | 1           |
| 1          | 7          | 2           |
| 1          | 8          | 2           |
| 1          | 9          | 2           |

3.1. The Result of FNN performance

During the training of FNN, each particle brings a solution. Performance of FNN is shown by computing how well the Mean Square Error (MSE) value is generated. The smaller the MSE value means the better the resulting performance and vice versa, if the MSE value is higher it means there are still many errors that are obtained. The formula for calculating MSE is shown in Eq. (10).

\[
MSE = \frac{1}{N} \sum_{i=1}^{n} (Y_i - D_i)^2
\]  

Where \( Y_i \) is an actual output target and \( D_i \) is a data output from FNN while \( N \) is a total of training data.

The performance of FNN is less than optimal. In the graphic of train data, there are 2 lines which is the red one is the outputs from the training and the black one is the targets. It is mean still there is a significant distance or gap between the output and the desired target, so that’s why the performance is
considered not maximal. Besides that it is proven by the calculation of MSE on the graph with a blue line. The value of MSE tends to be still high that is 6.3299.

3.2. The Result of Training FNN using PSO

The implementation of PSO for optimized the fuzzy neural network system using a maximum iterations parameter of 2000 and a population parameter of 25. The value of best cost will be more stable when approaching the 2000 iteration. The changes of value look more significant in the process at the beginning of the iteration.

Based on the result in Figure 4. The graphic of train data indicates that the output results coincide with the desired target, that mean the output from the process has a lot of conformity with the expected target. In this case the performance of FNN will increase. Furthermore, the value of MSE has a lower value than before that is 2.065. Different result before training and after training can be seen in Figure 4.

![Figure 4. Different Result Before Training and After Training](image)

4. Conclusion

In this research we applied the Fuzzy Neural Network (FNN) and Modified Particle Swarm Optimization (MPSO) to control a traffic signal. The FNN method is tuned by using MPSO for learning process. A better iteration result is indicated by a smaller MSE value of 2.065 compared to without using PSO which produces MSE 6.3299. The result shows that the implementing of both methods can improve the performance by 4.26%. The accuracy between the output and the target from training process by using MPSO is higher then basic FNN model. Because of the success of the MPSO learning process for traffic case that is means FNN controller has the ability to adapt dynamic traffic condition as the proposed in this study.

References

[1] Yousef K M, Jamal N A and Ali M S 2010 J. Inf. Sci. Eng. 26 753
[2] Zaky M, Davindra G A, Endra J and Herman Y S 2017 Internetworking Indo. J. 9 1
[3] Royani T, Javad H and Mohammad A 2013 Int. J. Comp. Electrical Eng. 5 87
[4] Lai G R, Azura C S, Haslina M S, Ramli Z A R and Mohd K H 2015 J. Eng. Sci. Tech. 10 1009
[5] Debnath S B C, Pintu C S and Kazuyuki M 2013 Int. J. Artif. Intelligne. 4 37
[6] Muslim M A, Iin K, and Endang S 2015 J. Theor. App. Inform. Tech. 78 70
[7] Azura C S and Haslina M S 2010 Int. J. Comp Sci. Eng. 2 924
[8] Permana K E and Siti Z M H 2010 Int. J. Open Probl. Compt. Math 3 27
[9] Stotsko Z, Yevhen F and Ihor M 2013 Transport Prob. 8 5
[10] Dutta D, Roy R and Choudhury K 2013 Int. J. adv. Reseach Comp. Sci. Softw. Eng. 3 430
[11] Kuo R J, Hong S Y and Huang Y C 2010 J. ELSEVIER Applied Math. Model 34 3976
[12] Muslim M A, Siti H R, Endang S, Budi P and Alimah S 2018 J. Phys.: Conf. Ser. 983 012063