Applied Soft Computing 12 (2012) 759–767

Contents lists available at SciVerse ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Soft computing methods applied to train station parking in urban rail transit

Dewang Chen [a,*], Chunhai Gao [b]

[a] State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing 100044, China
[b] National Engineering Research Center of Rail Transportation Operation and Control System, Beijing Jiaotong University, Beijing 100044, China

ARTICLE INFO

Article history:
Received 30 December 2009
Received in revised form 2 August 2011
Accepted 23 October 2011
Available online 4 November 2011

Keywords:
Soft computing
Train station parking
Generalized regression neural network
Subtractive clustering
Neuro-fuzzy system
Urban rail transit

ABSTRACT

This paper presents three models – a linear model, a generalized regression neural network (GRNN) and an adaptive network based fuzzy inference system (ANFIS) – to estimate the train station parking (TSP) error in urban rail transit. We also develop some statistical indices to evaluate the reliability of controlling parking errors in a certain range. By comparing modeling errors, the subtractive clustering method other than grid partition method is chosen to generate an initial fuzzy system for ANFIS. Then, the collected TSP data from two railway stations are employed to identify the parameters of the proposed three models. The three models can make the average parking errors under an acceptable error, and tuning the parameters of the models is effective in dynamically reducing parking errors. Experiments in two stations indicate that, among the three models, (1) the linear model ranks the third in training and the second in testing, nevertheless, it can meet the required reliability for two stations, (2) the GRNN based model achieves the best performance in training, but the poorest one in testing due to overfitting, resulting in failing to meet the required reliability for the two stations, (3) the ANFIS based model obtains better performance than model 1 both in training and testing. After analyzing parking error characteristics and developing a parking strategy, finally, we confirm the effectiveness of the proposed ANFIS model in the real-world application.

1. Introduction

Among different kinds of transportation, railway transportation is one of the most energy-saving and environment-protection transportation means [1]. In recent years, most research of the train control focused on the optimal energy-efficient operation [2,3] and the designs of control algorithms [4,5]. As a constraint condition of railway transportation, however, the station parking error is less studied. A possible reason for this is that the required accuracy for the train station parking (TSP) is often several meters, which is easy to meet for the long distance railway between different cities. As the fast development of the urban rail transit, however, accurate train station parking is becoming a key technique. Especially, train station parking with high accuracy (about ± 30 cm or ± 50 cm) is necessary to ensure that train doors will be opened and passengers can get on and off the train easily in train control systems with platform screen doors (PSD). Note that in the paper, TSP means that the last braking will make a train stop at the parking sign at a station.

To develop an accurate and reliable TSP algorithm is very hard due to multiple factors, such as the huge inertial-mass of a train, the nonlinear characteristics of a train braking system, the grade, the curvatures of railway tracks, weather, and the number of passengers. Furthermore, it is difficult to collect the field data because the train parking experiment is costly to conduct. Consequently, some train station parking systems try to achieve the high-accuracy parking by investing heavily on equipment, such as range sensors, transponders, and laser radars, to continuously verify the position and velocity of the train, and by performing many field tests to regulate the control parameters repeatedly [7,8]. Therefore, it is necessary to adopt a parameter updating method to improve the accuracy of TSP and to reduce the cost of a train control system. Yasunohu et al. employed the predictive fuzzy control to achieve better TSP performance than PID control in simulation [9]. However, how to automatically tune the parameters in fuzzy membership functions (MF) was not discussed and the real effect in operation was hard to assure due to lack of field data. To better evaluate the performance of TSP algorithms, a distinct feature of this paper is that the field TSP data are used instead of simulation data.

As a train runs on the same track repeatedly in a urban rail transit, many train parking data for the same railway station can be obtained in the field. Thus, it is possible to apply soft computing techniques into TSP based on the experiment data to learn the underlying parking rule like an experienced train driver. Soft computing simulates the human mind to have the ability of reasoning and learning to find hidden rules, to learn useful patterns from the data set. Soft computing consists of fuzzy logic, neural network, neuro-fuzzy system and so on [10]. Soft computing techniques have

* Corresponding author.
E-mail addresses: dwchen@bjtu.edu.cn (D. Chen), chhgao@bjtu.edu.cn (C. Gao).

1568-4946/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.asoc.2011.10.016

broad applications in engineering, such as medical image clustering [11], supply chain management [12], and engine valve manufacturing process [13]. As to the TSP modeling, we have done some preliminary research about using neuro-fuzzy system to build a TSP model from some field data by a PID controller [14]. However, it remains three shortcomings, (1) data was few and only from one station and the data from the PID controller is not good as that from a human driver, as PID controller frequently change the output even a train is very close to the parking sign, (2) the simplest grid partition method was used to generate initial fuzzy system, hence, advanced methods need to be explored, (3) only simulation, no application in that research. To address the TSP problem better, more advanced soft computing techniques need to be employed to develop error estimation models based on more experimental data collected by a train driver.

By mining the underlying driving rules through soft computing techniques, high parking accuracy is aimed to be achieved and the costly investment in equipment will be reduced. Generalized regression neural network (GRNN) [15], which is one of the neural networks with great regression ability, will be used to estimate the TSP error. On the basis of the fuzzy control on TSP, we will employ adaptive network based fuzzy inference system (ANFIS) [16] with different methods in generating initial fuzzy systems to explore its learning ability by combining least square method (LSM) [17] and the back propagation (BP) [18] method to increase the accuracy of TSP. Furthermore, the parameter updating methods will be developed to dynamically decrease the parking error after each parking operation.

The rest of the paper is organized as follows. In Section 2, three models for estimating the TSP error are developed based on linear regression model, GRNN and ANFIS respectively. In Section 3, five statistical indices are given to evaluate the accuracy and reliability of the three models and the parameter updating method is developed. In Section 4, the models are compared and analyzed in detail using the training data set and testing data set from two stations after the field TSP experiments are introduced. The parking strategy, the locations of the positioning sensors, and the results of the real-world application using the ANFIS based TSP model are provided in Section 5. Finally, conclusions and future research are outlined in Section 6.

2. TSP modeling

When a train attempts to park a station by braking, its parking error can be influenced by the braking force, the fiction force, the air resistance force, the gravity force caused by the track grade, and the eccentricity force by the track curvature. To simplify the problem, we assume a train as a mass point. Then the kinematics for describing the parking process are as follows.

$$\frac{dv}{dt} = -f_b b(v) - f_r(v) - f_g(s) \tag{1}$$

$$\frac{ds}{dt} = v \tag{2}$$

where v is the velocity of a train, f_b is the relative braking forces for different braking levels, $b(v)$ is the function of parking forces with the velocity of the train, $f_r(v)$ is the resistance force caused by friction and air, $f_g(s)$ represents the resistance force by grade and curvature and s is the coordinate value of the train.

In the beginning time of parking t_p, let the velocity of the train be v_p, the distance from the train to the parking sign be s_p. And let the ending time of parking be t_e. As the velocity for a still train is zero, we have:

$$v_p - \int_{t_p}^{t_e} (-f_b b(v) - f_r(v) - f_g(s))dt = 0. \tag{3}$$

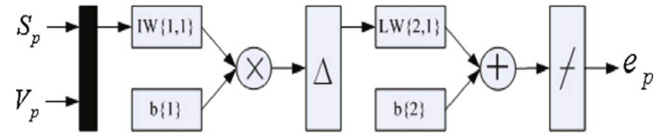

Fig. 1. GRNN model for TSP.

Then, the parking error is

$$e_p = s_p - \int_{t_p}^{t_e} vdt. \tag{4}$$

In general, the air resistance and the centrifugal force are the quadratic functions of the velocity and the friction force is the linear function of the velocity. As there exist delay and nonlinear characteristics in the braking, and coupling forces between different carriages, it is very difficult to study TSP by a precise mathematical model which considers all factors together.

As a train recurrently parks at the same railway parking sign, we can think TSP as a new problem in the view of soft computing, i.e., to use soft computing techniques to estimate the parking error given a certain initial velocity, position and braking level in the training parking data set. To simplify the problem, we have

$$e_p = f(v_p, s_p, B, R, D). \tag{5}$$

where v_p and s_p are the initial velocity and position of the train in parking respectively, B represents the braking force, R are all kinds of resistance forces and D represents all kinds of disturbances which include the weather, the number of passengers and so on.

If all factors are taken into consideration, the parking error will be a very complicated multivariate nonlinear function. As a result, it is hard to design a mathematical model to identify the parameters and develop a parameter updating method. Note that the velocity of the train in the beginning of parking is low, all kinds of resistance forces and disturbance factors are relatively small and vary slightly. Therefore, it is possible to simplify the TSP model with alternate models. To use the soft computing technologies, we recorded all data pairs in the same braking level in the TSP field tests in the form of (s_p^i, v_p^i, e_p^i) $(i = 1, 2, \ldots, n)$ to represent the initial position, velocity and the parking error for each parking operation. If a train surpasses the parking sign, the parking error is defined as negative value; otherwise, as positive. Then we can develop the following three models.

2.1. Model 1 based on linear regression

As a train's velocity is low in the parking phase, the resistance forces and disturbance force are usually relatively small. Although there are some nonlinear braking characteristics, it is reasonable to hypothesize that the parking error e_p has a linear relationship with s_p and v_p.

$$e_p = k + k_s \times s_p + k_v \times v_p \tag{6}$$

where k, k_s and k_v are three predetermined linear parameters, which can be identified by a least square estimation (LSE) method from the collected data set.

2.2. Model 2 based on GRNN

GRNN is one type of radial basis networks which is often used for function approximation. GRNN has a radial basis layer and a special linear layer which has as many neurons as the number of input vectors. Furthermore, the center of each radial basis function (RBF) as in Eq. (7) is an input vector and the weights of the linear layer are target vectors. It is easy to design a GRNN and update the model's parameter for a two-input-and-one-output TSP model, as illustrated in Fig. 1. The inputs of model 2 are the initial position and