Journal of Scientific & Industrial Research
Vol. 70, March 2011, pp. 210-214

Optimization of balise placement in a railway track using a vehicle, an odometer and genetic algorithm

Mohammad Ali Sandidzadeh* and Ali Khodadadi
School of Railway Engineering, Iran University of Science and Technology, Tehran, Iran

Received 01 September 2010; revised 07 February 2011; accepted

This study presents optimization of balise placements in a multi-sensor architecture for onboard navigation system of a train, which runs on a track equipped with balise-based signaling system. Determination of locations of balises depends on a variety of parameters. Genetic Algorithm (GA) and Kalman filtering concept were used to find optimum places of balices in the line to reduce tachometer errors, which are one of the most important sensors in train navigation.

Keywords: Balise transponder, Genetic algorithm, Kalman filter, Railway control, Train positioning

Introduction

Automatic train control (ATC) in railway systems benefits from various methods to render operation of running trains safe, and provide good quality of service. One of the most famous transmission mediums for transferring data between running trains and wayside equipments, and alongside tracks, is a special transponder, balise or beacon. Correct and optimized placement of balises through a line can make train to be controlled in a flexible and high performance manner and to meet various control and operation criteria. Proper balise placements also decrease headway times between trains and also balises can be used to reduce Tachometer errors as a reference. Due to technological advances and necessity of exact positioning of vehicles in commercial and military applications, dead-reckoning sensors are used. Each sensor has its own strength, weakness, and restrictions that result from its mechanical, electrical, and commercial limitations[1]. In new moving-block railway control systems, RAMS (Reliability, Availability, Maintainability and Safety) of systems are mainly based on dependability of two subsystems[2]: i) onboard positioning system and ii) communication system between trains and control centre. Communication based train control system (CBTC), transmission based train control system (TBS), and other equivalent systems need to provide necessary amount of RAMS in these subsystems before making control system of a moving train functional[3,4]. In order to achieve high accuracy in the positioning systems and omit weaknesses, three main approaches are usually used: i) Using more expensive sensors and complicated algorithms that have more precision than common methods; ii) Benefiting from sensor fusion algorithms in vehicles navigation; and iii) Using special ways to use sensors in high precision applications.

This study employed Genetic Algorithm (GA) for optimal determination of balise locations in order to minimize positioning errors of tachometer. Fusion algorithms were used to combine tachometer and balise data, and after that with GA, best locations for balises may be found.

Experimental Section

Sensors

Tachometer

A tachometer is a common sensor used for calculating distance traveled by automobiles or trains. Generally, there are two different types of tachometers (magnetic field tachometers; and Hall Effect tachometers). All tachometers are modeled as

$$V_{meas} = V + bias_v + nois_v \qquad (1)$$

*Author for correspondence
Tel: +98-21-77491030; Fax: +98-21-77451568
E-mail: sandidzadeh@iust.ac.ir

where, V, real speed; $bias_v$, random bias; V_{meas}, measured speed from sensor; $nois_v$, white Gaussian noise.

Balise

Balises are fixed point devices, which transmit and receive radio signals to and from vehicles, and used in air and sea navigations[5,6]. In railway systems, balises are used for passing information from tracks to trains, and also for train detection as a part of railway signaling systems. A magnetic balise comprises a transponder (usually passive device and track mounted) and an interrogator (active device and train mounted). As a train passes over balises, it interrogates data permanently stored inside or fed to balises via data links. Train-based receiver can receive the data that might include different information (line topography, speed restrictions, distance to next station, and also balise position). Positioning data of balises include two types of errors: i) caused by data transmission delay, ii) positioning data error, which is stored in balise itself. Positioning error is effectively a function of train speed. Positioning error of balises can be modeled as a Gaussian white noise with zero mean. Error caused by delay time can be modeled as a constant bias.

Implemented Algorithms

Kalman Filter

Kalman filter (KF) is a set of mathematical equations that provide efficient recursive solutions to minimize mean square error by using a form of feedback control. KF estimates state of the system and then obtains feedback from noisy measurements[7]. Process and measurement equations are as

$$X_{k+1} = \Phi X_k + Bu_k + Cw_k \tag{2}$$

$$Z_k = HX_k + v_k \tag{3}$$

where X_k, system state vector; Z_k, measurement vector; H_k, measurement matrix; Φ, state transition matrix, w_k represents noises caused by modeling errors and/or any alteration in the model parameters due to process or environmental condition changes; and v_k, noise measurement from different sources such as thermal and vibration noises, initialization errors and so on. Algorithm used in KF is:

Measurement update:

1 - Computing the Kalman gain

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1}$$

2 - Update the estimate with measurement Z_k

$$\hat{X}_k = \hat{X}_k^- + K_k (Z_k - H_k \hat{X}_k^-) \qquad \ldots(4)$$

3 - Update the error covariance

$$P_k = (I - K_k H_k) P_k^-$$

Time update (prediction):

1 - Project the state ahead

$$\hat{X}_{k+1}^- = f_k \hat{X}_k$$

2 - project the error covariance ahead $\qquad \ldots(5)$

$$P_{k+1}^- = f_k P_k f_k^T + Q_k$$

where K_k, Kalman gain of KF; P_k, error covariance of estimate; and $\hat{X}_k$, state vector at time k.

System Model

In this study, a Markov process acceleration model for train dynamic has been considered as

$$\begin{aligned} \dot{s} &= v \\ \dot{v} &= a \end{aligned} \tag{6}$$

$$\underline{x} = \begin{bmatrix} S \\ V \\ a \\ b \end{bmatrix} \quad \begin{matrix} \textit{Position} \\ \textit{velocity} \\ \textit{acceleration} \\ \textit{Tachobias} \end{matrix} \tag{7}$$

where S is position, V indicates velocity and a stands for acceleration.

Genetic Algorithm (GA) Implementation

GA is based on natural selection, and repeatedly modifies a population of individual solutions. At each step, GA selects individuals at random from current population to be parents, and uses them to produce children for next generation. Over successive generations, population evolves toward an optimal solution. GA can be used to solve a variety of optimization problems that are not well suited for standard optimization algorithms, including the

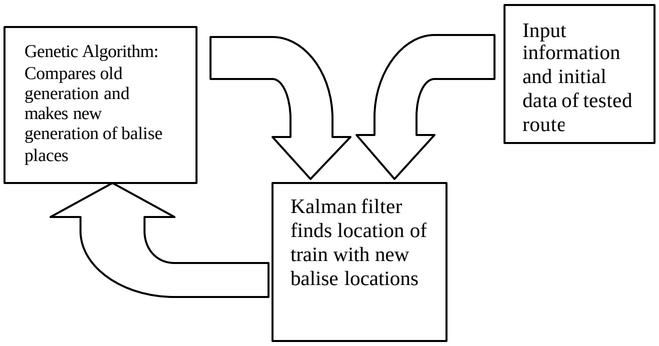

Fig. 1—General schematic of algorithm

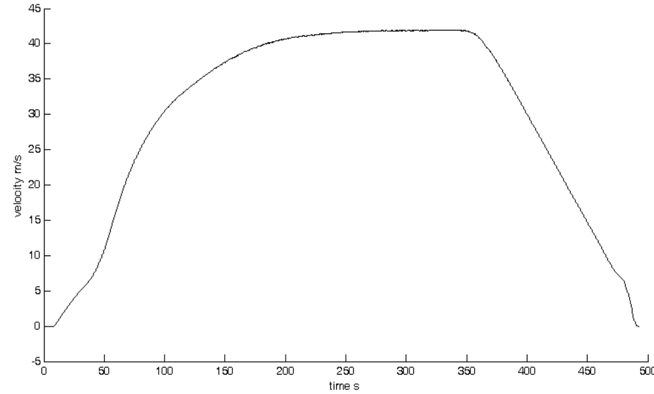

Fig. 2—Velocity graph of the vehicle during the simulation

Table 1—Balise locations and error estimation

Balise locations through the line								Velocity estimated error (RMS)
5034	7096	9041	11298	13970	14500	14700	14800	5.9081
1000	2000	3000	4000	7000	9000	12000	14000	3.0498
2430	4985	9900	11200	12000	13453	14000	14200	1.1029
400	800	1200	1600	2000	3000	5000	10000	1.2980
1660	3320	4980	6640	8300	9960	11620	13300	1.3219
Proposed balise locations with GA								
40	230	2711	5581	12060	13697	15405	14909	1.0128

problems in which objective function is discontinuous, not differentiable, stochastic, or highly nonlinear[8]. To create next generation from current population, GA uses following three main types of rules at each step: i) Selection rules % select the individuals, called parents, which contribute to the population in next generation; ii) Crossover rules % combine two parents form the children for the next generation; and iii) Mutation rules % apply random changes to individual parents to form children.

Proposed algorithm (Fig. 1) is set up as follows: i) Input parameters for algorithm should be determined; in this approach, balise locations are input variables of GA; ii) initial random population with its goal functions should be implemented; locations should be between 0 to 15000 m; and iii) variables in optimization function (KF) are implemented. Outputs are arranged and finally new generation is created by means of crossover and mutation functions. Genes are considered as input variables. Lengths of chromosomes are determined according to the number of variables (Number of balises). In this approach, it is 9. Value of initial population is 40. Fitness function is obtained as

$$X_{error} = RMS(X_{GPS} - X_{Kalman}) \quad \text{...(8)}$$

where X_{error} = Input data for fitness function of GA (Scale for accuracy of position with examined balises places), X_{GPS} = Position obtained from GPS as reference data, X_{Kalman} = position obtained from tachometer and balises data, and RMS = root mean square.

GA is a useful method, mainly where finding optimum point of a function involves a lot of mathematical and computational efforts. In addition to their large numbers, installation of balises is also a function of complicated economical and technical assumptions in railway lines.

Assumption considered in balise places are as follows: i) Installation of balises in the vicinity of stations and depots is economical in both installation and maintenance procedures; ii) In some lines, several balises are installed for many reasons (their locations are fixed); and iii) Existence of balises in some regions across the lines is not safe such as switches and crossovers are not suitable locations for balise installation. GA provides a chance to solve mentioned problems and to overcome mentioned concerns by adding goal-function parameter to present algorithm.

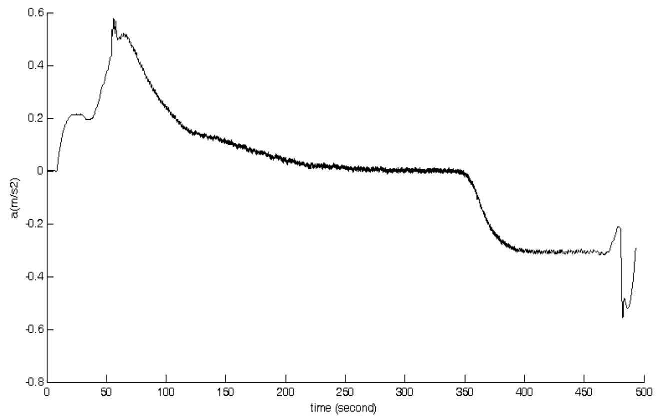

Fig. 3—Train acceleration data obtained from the tachometer sensor

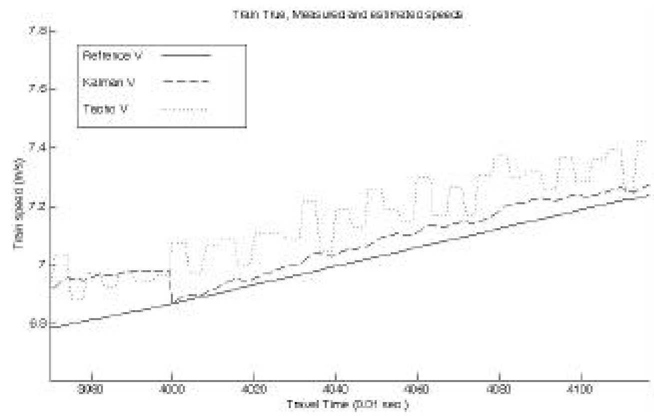

Fig. 4— Effect of the balises on Kalman Filter

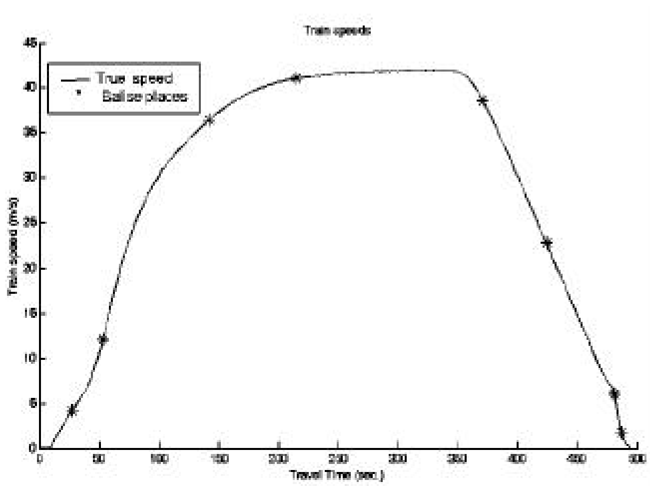

Fig. 5—Balise locations and velocity profile

Results and Discussion

Proposed algorithm was developed and improved by various simulated cases, and was implemented and verified in a real case of a train composed of a GM type locomotive with 10 passenger wagons that traveled a distance of 15 km. Train moved at a constant speed for 150 s, and after reaching the speed of 40 m/s, tachometer sensor had a sampling rate of 100 samples/s and 1p/s for GPS (Ground positioning system). Solid line (Fig. 2) shows reference speed data provided by GPS system. Fig. 3 shows acceleration profile during test. Reference speed (Fig. 4) is obtained from GPA data, tachometer speed data and KF speed data. Estimated velocity is corrected when train passed one of the balises at t=4000.

In the first step to test algorithm, balises are mounted through the line for several times randomly, and RMS errors of estimated velocity for each time were compared (Table 1). In the next step, GA was implemented to find best locations for balises (Table 1). RMS error obtained from proposed method is 1.0127, which is far better than existing method. If Balises are put through the line in a regular format then error is 3.04981, which is a higher value than GA results, may be due to being complimentary of the problem. This is one of the advantages of GA. Average error of sensor while vehicle gains and reduces speed is more than other situations (slipping and sliding causes). As a result, to minimize errors, balise positions have to be at places with high acceleration value. It is observed that balise locations are swept to the places with high differential in velocity values (Fig. 5).

Conclusions

One of the most common positioning approaches, applied in railway and automobile systems, is to use tachometer and balises together. Each sensor has its own noises and therefore is not a complete equipment to be used. For improving results and decreasing noises, this study used GA to obtain best locations for balises in the track. With the combination of GA and KF, proposed method gave best location of balises. In addition to train positioning, proposed algorithm can be used for car positioning and sea cruises methods.

Acknowledgments

Authors thank the personnel of Iranian Railway Company for kindness in giving permission and assistance for testing point machine, and also for granting access to their experimental data. Authors also appreciate Mr Hamed Zafari for assistance in revision of this paper.

References

1 Gustafsson F, *Statistical Sensor Fusion* (A B Studentlitteratur) March 2010.

2 Mirabadi A & Mort N, Design of fault tolerant train navigation systems, in *Proc Amer Control Conf, ACC'99* (San Diego, California, USA) June 1999.
3 Libor P E & Roman M M, Sensor data fusion for inertial navigation of trains in GPS-dark areas, in *IEEE Intell Vehicle Symp 2003*, 345- 350.
4 Ohringer F B & Geistler A, Comparison between different fusion approaches for train-borne location systems, in *IEEE Int Conf on Multi-sensor Fusion &Integration for Intell Syst* (Heidelberg, Germany) Sept 2006, 267-272.
5 Nicosevici T, Garcia R, Carreras M & Villanueva M, A review of sensor fusion techniques for underwater vehicle navigation, in *IEEE Techno-Ocean'04*, **vol 3**, 2004, 1600-1605.
6 Gunther R, Test and integration of location sensors for a multi-sensor personal navigator, *J Navig*, **60** (2007) 107-117.
7 Welch G & Bishop G, *An Introduction to the Kalman Filter* (Department of Computer Science, University of North Carolina, Chapel Hill) 24 Jul 2006.
8 Whitley D, *A Genetic Algorithm Tutorial* (Computer Science Department, Colorado State University).