JOURNAL OF ADVANCED TRANSPORTATION
J. Adv. Transp. 2016; **50**:2060–2076
Published online 7 March 2017 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/atr.1445

Train movement simulation by element increment method

Gaowei Xu[1,2], Feng Li[1]*, Jiancheng Long[3] and Ding Han[3]

[1] *State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing 100044, China*
[2] *Department of Civil Engineering and Applied Mechanics, McGill University, Montréal, QC H3A 2K6, Canada*
[3] *School of Transportation Engineering, Hefei University of Technology, Hefei 230009, China*

SUMMARY

The article presents an element increment method that is developed by current time increment method of train traction calculation. A railway route was divided, breaking it down into elements of different lengths. A whole train movement simulation curve (*v–t* curve and *v–S* curve) was formed by splitting the joints of each of the elements' individual simulation curves. During this process, the train velocity variance was calculated by time increment method with assistance of polynomial fitting technology. Additionally, a step-by-step method with iteration was used to combine each element and makes the whole simulation curve continuous. Meanwhile, the energy-saving issue was also taken into account to optimize the simulation curve. This article gives more details about the modeling by providing an example of a railway route based on moving block control. The element increment method is a more effective way to calculate train traction of high-speed railway, and it is an alternative method to train movement simulation for aiding macroscopic railway transportation planning. Copyright © 2017 John Wiley & Sons, Ltd.

KEY WORDS: discrete model; polynomial fitting; element increment method; energy saving

1. INTRODUCTION

In a complicated railway system, the most critical question for senior managers is how to maximize the economic benefits and running efficiency while ensuring transportation safety. To reach the answer, we should accurately simulate the running conditions of trains.

Nowadays, train simulation can normally be calculated by train dynamic calculation [1–3]. One of the most important components of train dynamic calculation is to calculate the net force, which is composed of powering force or braking force, resistance force, and additional resistance force. The additional resistance force is mainly caused by a longitudinal gradient, but also can be caused by a tunnel or by horizontal curvature. Because the net force will vary with the change of velocity, the acceleration in train powering or braking procedure is not constant. A commonly used algorithm is to assume that the net force acting on the train is constant over a short section in order to simplify the calculation of train dynamics. Thus, we should use step-by-step method by time interval (called time increment approach) to calculate the velocity curve of a train, and the step length is normally 1 second [4].

Train dynamic calculation is widely used in many applications of railway engineering. Capillas illustrated a train operation simulation method for calculating train dynamics and electric demand [5]. According to the characteristics of underground trains tracking, Yiping Fu generated a cellular model to simulate the operation of Beijing subway line 2 [6]. Geng simulated the operation of a heavy haul train on Datong–Qinghuangdao railway [7]. Shi and Guo built the model and the algorithm of an automatic constant speed train dynamic calculation in order to obtain indexes in China Railway High-speed running process on a high-speed railway [8]. Hwang [9] and Bocharnikov [10] did

*Correspondence to: Feng Li, State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing 100044, China. E-mail: fengli0925@gmail.com

research regarding the optimization of traction energy consumption during a single-train journey by balancing energy saving and running time increases on the direct current suburban railway and high-speed railway, respectively. The development of automatic train operation and automatic train protection are all based on the results of train dynamic calculation [11–13]. Furthermore, the results of train simulation can be applied to do research on railroad improving [14], train performance evaluation [15], rail networks simulation [16], railway economy [17], and train scheduling [18–23].

2. DEVELOPMENT OF TRAIN MOVEMENT SIMULATION

Because the train movement simulation is so basic, some research regarding the improvement of this topic has been proposed. There are three basic ways to represent the mass of a train. Single particle is a computationally efficient but less accurate representation method. Both multiple points and line part are complex but more accurate representation methods. However, single-particle representation is still an adequate approach if the time interval of step-by-step method is divided finely enough [24]. Uher proposed a way to find the location and initial speed for deceleration when a train is ready to arrive at a station or before entering the block forward with lower limited speed, which is by finding the intersection of the forward and backward speed trajectories [25]. Kikuchi developed a kind of method using distance increment method (approximately 0.02 miles step length) to simulate train operation [26]. Zhang *et al*. put forward a combined simplified model to develop EMU trains' dynamic calculating system based on business needs [27]. Furthermore, some widely used software such as TrainSim, OpenTrack and RailSys have been developed for train running simulation [28–30].

Although the step-by-step method of time-increment approach can help us to accurately simulate a train, its calculation efficiency is reduced for the very short step intervals [31]. We need to calculate thousands of steps in order to finish the whole simulation procedure particularly for longer railway lines and higher velocity trains. Therefore, some efforts based on train traffic flow theory have been generated for solving the train movement simulation issue. Ho proposed an event-based model to simulate multi-train moving simulation that substantially saves computational effort [32]. However, it relies on the assumption that the passage of time is irregular and the updates of train movement are not carried out synchronously, so its applications are mainly in traffic control and train scheduling, as only the timing information at certain events and quick simulation results are expected in the model [33]. Li *et al*. developed a cellular automaton model for simulating railway traffic and train trajectory, which is also an effective computational method, but some key factors such as also railroad geometry, traction equipment, and train length have not been taken into account [34].

However, if the train movement simulation model is built by separating the railway into long-distance sections and using the distance increment method, it will need to calculate only dozens of steps instead of the thousands of steps needed in the time increment method. Thus, the calculation efficiency is improved. The train movement simulation curve (v–t curve and v–S curve) can be drawn by splicing the discrete distances. A distance increment method that can achieve the results just mentioned, and it is suitable for high-speed railway and rail transit in which trains are controlled by moving block.

The paper is organized as follows: we illustrate the concept of the element increment method (EIM) in Section 3, and introduce the model in Section 4. An example of the modeling process with five subsections is shown, and the numerical and analytical results are attached. Finally, a conclusion of the approach is presented.

3. CONCEPT OF ELEMENT INCREMENT METHOD

Figure 1 shows the profile of EIM. A route from A to B is divided into several end-to-end elements. After working out train movement simulation curve (here is v–t curve) from elements ① to ⑤, then we put them together to form the train's v–t curve of the whole route. However, because each element is analyzed individually, they cannot be directly joined together. An iteration work is adopted to make adjacent elements intercoordinated, resulting in a continuous curve. This issue will be discussed in Section 4.2. Based on v–t curve, we integrate to obtain v–S curve and t–S curve.

DOI: 10.1002/atr

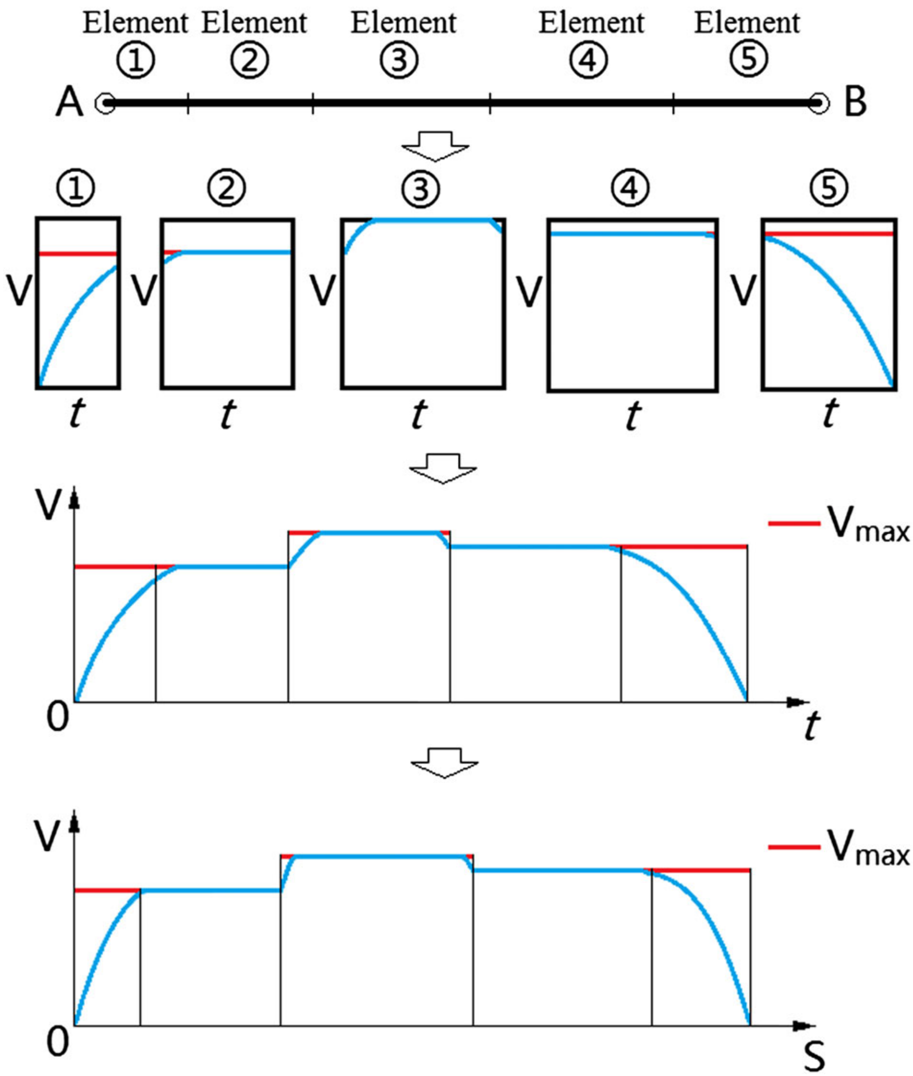


Figure 1. Concept of element increment method.

The methodology is named EIM instead of distance increment method, because, on one hand, in the time increment method, all increment steps are the same; on the other, in different element, the distance is different.

4. MODEL FORMULATION

4.1. Modeling assumption

In order to simplify the issue, listed here is the modeling hypothesis:

(1) Train operate regularly without malfunction.
(2) The mass of a train is simplified as a single particle.
(3) For electrified railway, the influence of neural zone on velocity can be ignored.
(4) A train tries to travel at velocities as close to the velocity limits as possible but not exceeding the limits.
(5) All trains can operate with uniform velocity if it is required.
(6) Simplify the shifting rule of train operation mode to Figure 2. Simplify the coasting operation mode in the processing of mutual transformation between pulling operation mode and braking operation mode to uniform motion, and denote t_0 as duration time (normally t_0 equals 10 seconds).

DOI: 10.1002/atr

$$\textbf{\textit{Pulling}} \rightleftharpoons \textbf{\textit{Coasting}} \rightleftharpoons \begin{array}{c}\textbf{\textit{Braking}}\\ \textbf{\textit{(Dynamic/Air)}}\end{array}$$

Figure 2. Regulation of switching operation modes.

4.2. Discrete railway element

Referring to the concept of EIM, because we need to splice the simulation curve of a train on each element to form the whole curve of the train dynamic operation, we should investigate all conditions when the train passes every element and work out the *v*–*t* curve of each element. In the procedure when a train passes an element, the maximum velocity of the train is v_i^m. A represent element *i* is as the following form (refer to Figure 3). We use line segment to represent every discrete railway element, each element is numbered, and assign the real distance L_i to them (subscript *i* is the number of element).

In Figure 3, assume that the train operates from nodes **a** to **b**. Use v_i^e and v_i^l to represent the arrival velocity (enter) and departure velocity (leaving) respectively in element *i*, and use v_i^m and v_{i+1}^m to represent the maximum velocity in element *i* and in element $i+1$, respectively. The departure velocity in the former element v_i^l is the arrival velocity of the next element, that is, $v_{i+1}^e = v_i^l$.

Here, the direction of train operation is from **a** to **b**, so the solving process is also from **a** to **b**. The first arrival velocity is known (train departure $v_1^e = 0$), and the following velocity at each node the train passes should depend on the property of relative element (i.e., the parameters mentioned before). In order to compute t_i that is the time the train passes through element *i*, the v_i^e and v_i^m must be known. However, the v_i^l cannot be confirmed because it could be equal to or less than v_{i+1}^m. For example, the train speeds up in element *i*, but L_i is not long enough for the leaving velocity v_i^l to decelerate to v_{i+1}^m. Therefore, L_i, v_i^e, v_i^m, and v_{i+1}^m are known, and the v_i^l and t_i are unknown unless the train stops at node **b**; thus, $v_i^l = 0$.

However, when we calculate element *i*, we cannot simultaneously consider the influence on element $i+1$ from the element *i*. For example, if the train decelerates when it passes the element *i*, its velocity drops down to v_{i+1}^m ($v_i^l = v_{i+1}^m$). But when the train passes element $i+1$, there is a probability that the train cannot decelerate its v_{i+1}^l equal to or less than v_{i+2}^m because L_{i+1} is not long enough. So, the train should decelerate in element *i* in order to make v_i^l less than v_{i+1}^m. Consequently, we use the following iterative method to solve this problem.

When the train decelerates in element *i*, if the L_i is not long enough to let the train finish decelerating, use L_i to calculate a new entering velocity of this element that is used for replacing v_i^D for avoiding overspeed when the train leaves the previous element. Repeat doing the whole calculation process from the first element to the last element again and again until no replace happens any more.

4.3. Simulation of accelerating and decelerating

The velocity change of a train may happen on the following four running modes: powering, braking, coasting acceleration, and coasting deceleration. It is necessary to obtain *v*–*t* curves of these modes.

As we know, the net force of a train at each moment can be computed by:

$$c = \frac{\left(F_y - W - B\right) \times 10^3}{(P+G) \bullet g} = f_y - w - b \tag{4.3.1}$$

where Fy is traction force, W is resistant force, B is braking force, P + G is the total mass of the train, c is the net force. fy, w, and b are traction force on unit weight, resistant force on unit weight, and

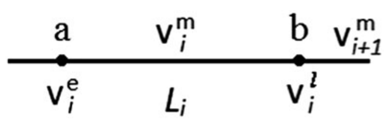


Figure 3. Parameters of the element *i*.

DOI: 10.1002/atr

braking force on unit weight, respectively. As a result, the acceleration and velocity increment can be computed by:

$$a = \frac{g}{1060}c \tag{4.3.2}$$

$$\Delta v = a\Delta t \tag{4.3.3}$$

However, because w and b are varied by velocity, the net force is a function of velocity. Consequently, the acceleration is also a function of velocity. Thus, in order to simplify each running mode, we divide velocity range every 10 km/hour with a constant c value in each interval, as shown in Figure 4, and then integrate acceleration to time to obtain a v–t relationship.

The v–t relationship has to cover the whole velocity range that a train could reach in the four running modes, and, through the analysis of a large number of typical dynamic characteristic curves, the cubic polynomial can well match these v–t curves [35] as shown in Figure 5. Meanwhile, we record the highest velocity that the train could reach (denote as v^t) and the corresponding time cost (denote as $t_{\max}$). We use $t_{\max}^T$, $t_{\max}^B$, $t_{\max}^{CA}$, and $t_{\max}^{CD}$ to represent the time cost in the four motions, respectively.In order to make the later process simple, here are some requirements for fitted curves:

① Should pass the terminal points—(0,0) and ($t_{\max}$, v^t).
② Either monotonously increase or monotonously decrease.
③ Try to make R^2 as close to one as possible.

4.4. Element types and solving

Based on the relationship of the arrival velocity v_i^e, the maximum velocity v_i^m, and the maximum velocity of its next element v_{i+1}^m, we can classify the elements into six groups: $v_i^e \le v_i^m \le v_{i+1}^m$, $v_i^e \le v_{i+1}^m\ v_i^m$, $v_{i+1}^m \le v_i^e \le v_i^m$, $v_{i+1}^m \le v_i^m \le v_i^e$, $v_i^m \le v_{i+1}^m \le v_i^e$, and$v_i^m \le v_i^e \le v_{i+1}^m$. If the velocity of a train does not exceed the design velocity v_i^D or downgrade limit brake velocity v_i^b or the maximum velocity the train could reach if it is on an upslope (denote as v_i^s), then the groups containing $v_i^e \ge v_i^m$ will be excluded. Thus, in the remaining groups—$v_i^e \le v_i^m \le v_{i+1}^m$, $v_i^e \le v_{i+1}^m \le v_i^m$, and $v_{i+1}^m \le v_i^e \le v_i^m$—we keep classifying based on various L_i in order to enumerate all types of operation mode. When applying this model in practice, we should fully consider the influence of coasting mode on velocity because of the energy-saving issue during the daily operation. Therefore, we use the balanced speed of coasting (denote as v_i^c), the critical accelerated speed of coasting (denote as v_{ac}^c), and the critical decelerated

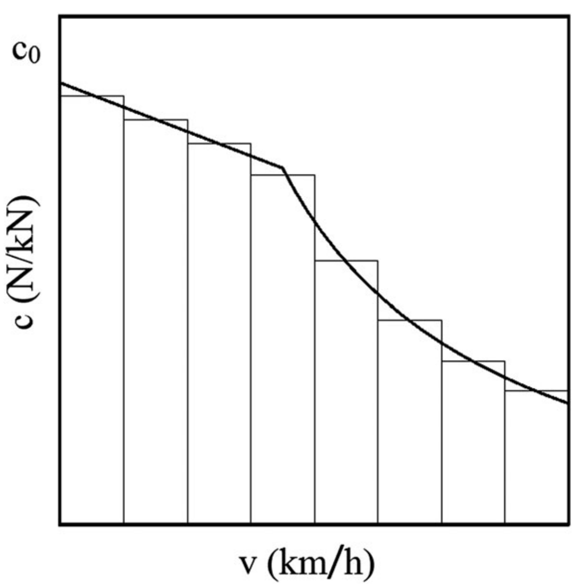
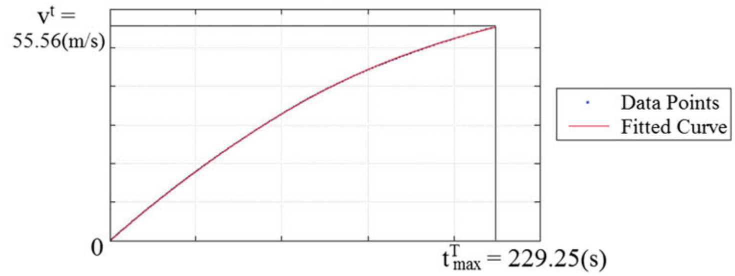


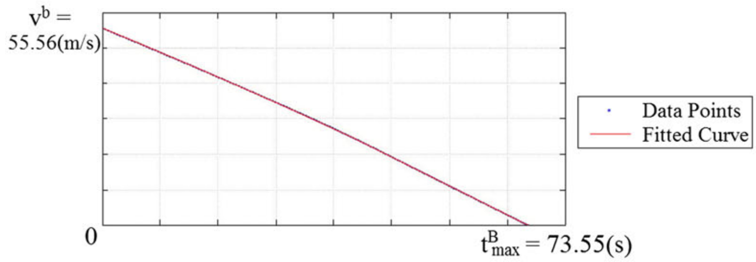
Figure 4. A typical dynamic characteristic curve of a train.

DOI: 10.1002/atr



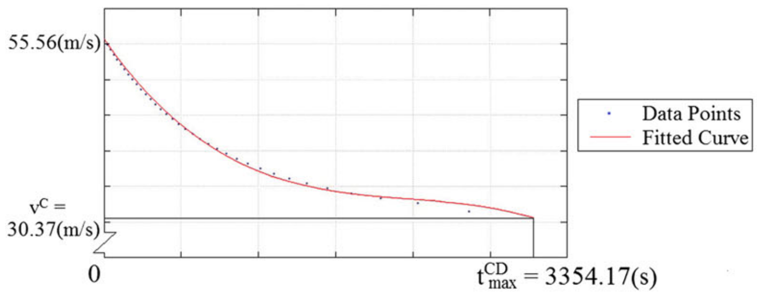
$f(x) = -1.045e^{-7}x^3 - 6.269e^{-4}x^2 + 0.3919x \qquad R^2 = 1$

(a)



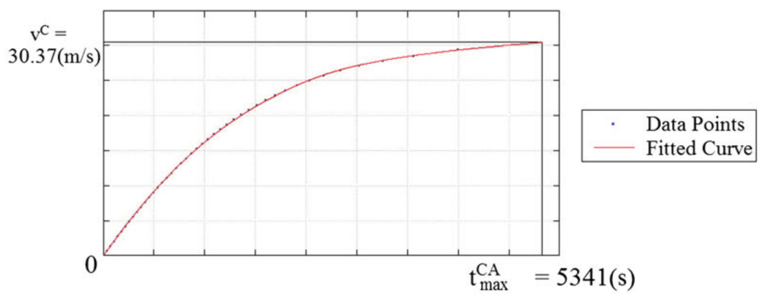
$f(x) = -1.154e^{-6}x^3 - 1.264e^{-3}x^2 - 0.6597x + 55.56 \qquad R^2 = 1$

(b)



$f(x) = -2.472e^{-9}x^3 - 1.462e^{-5}x^2 - 0.0305x + 55.56 \qquad R^2 = 0.998$

(c)



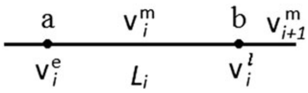
$f(x) = 4.458e^{-10}x^3 - 5.137e^{-6}x^2 + 0.0209x \qquad R^2 = 1$

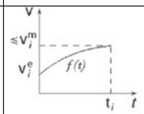
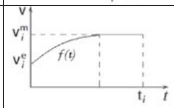
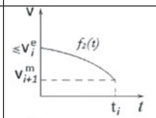
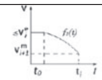
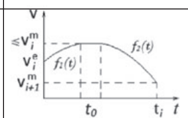
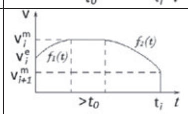
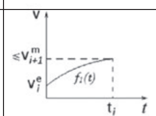
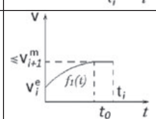
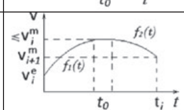
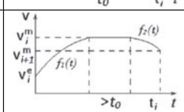
(d)

Figure 5. Fitting curves of velocity variance. (a) *v–t* curve of powering (0‰), (b) *v–t* curve of braking (5‰), (c) *v–t* curve of coasting to decelerate(−3‰), and (d) *v–t* curve of coasting to accelerate(−3‰).

DOI: 10.1002/atr

Table I. Category of elements and the corresponding algorithms (default: node **a** is on the left, and node **b** is on the right)



Given	v_i^e, L_i, $v_i^m = v_i^D$, $v_{i+1}^m = v_{i+1}^D$, v_{ac}^c, v_{de}^c (v_i^D has been revised by v_i^s and v_i^b)			
Group	Element type, critical length, and choose functions	Case	v–t graph	Solution process
I (speed up by powering or coasting on a steep slope)	$v_i^e \le v_i^m \le v_{i+1}^m$ Function needed: $f(t) = f_T(t)$ If $v_i^e \ge v_{ac}^c$, $v_i^m \le v_i^c$, then $f(t) = f_{CA}(t)$ (speed up by coasting) Calculate critical length: $v_i^e, v_i^m, 0, f(t) \xrightarrow{Algorithm1^*} S, \Delta t_1$ $L_{cr} = S$	$L_i \le L_{cr}$		$v_i^e, v_i^m, L_i, 0, f(t) \xrightarrow{Algorithm2^{**}}$ ① $v', \Delta t$ ② $v_i^D = v'$ (for the iteration)**** $v_i^l = v_2, t_i = \Delta t$
		$L_i > L_{cr}$		$\Delta t = \Delta t_1 + \frac{L_i - L_{cr}}{v_i^m}$ ① ② $v_i^l = v_i^m, t_i = \Delta t$
II (speed up by powering or coasting on a steep slope. speed down by coasting or braking)	$v_{i+1}^m \le v_i^e \le v_i^m$ Function needed: ① $f_1(t) = f_T(t)$ ② $f_2(t) = f_B(t)$ If $v_i^e \ge v_{ac}^c$ and $v_i^m \le v_i^c$, then $f_1(t) = f_{CA}(t)$ (speed up by coasting) If $v_{i+1}^m \ge v_{de}^c$ and $v_{i+1}^m \ge v_i^c$, then $f_2(t) = f_{CD}(t)$ (speed down by coasting) Calculate critical length: $v_i^e, v_{i+1}^m, 0, f_2(t) \xrightarrow{Algorithm1} S, \Delta t$ $L_{cr1} = S$ $v_i^e, v_{i+1}^m, t_0, f_2(t) \xrightarrow{Algorithm1} S, \Delta t$ $L_{cr2} = S$ $v_i^e, v_i^m, 0, f_1(t) \xrightarrow{Algorithm1} S_1, \Delta t_1$ $v_i^m, v_{i+1}^m, 0, f_2(t) \xrightarrow{Algorithm1} S_2, \Delta t_2$ $L_{cr3} = S_1 + v_i^m t_0 + S_2$	$L_i \le L_{cr1}$		$v_i^e, v_{i+1}^m, L_i, 0, f_2(t) \xrightarrow{Algorithm2}$ ① $v', \Delta t$ ② $v_i^D = v'$ (for the iteration) $v_i^l = v_{i+1}^m, t_i = \Delta t$
		$L_{cr1} < L_i \le L_{cr2}$		$v_i^e, v_{i+1}^m, L_i, t_0, f_2(t) \xrightarrow{Algorithm2}$ ① $v', \Delta t$ ② $v_i^D = v'$ (for the iteration) $v_i^l = v_{i+1}^m, t_i = \Delta t$
		$L_{cr2} < L_i \le L_{cr3}$		$v_i^e, v_{i+1}^m, v_i^m, t_0, L_i, f_1(t), f_2(t)$ ① $\xrightarrow{Algorithm3^{***}} v', \Delta t$ ② $v_i^l = v_{i+1}^m, t_i = \Delta t$
		$L_i > L_{cr3}$		$\Delta t = \Delta t_1 + \Delta t_2 + t_0 + \frac{L_i - L_{cr3}}{v_i^m}$ ① ② $v_i^l = v_{i+1}^m, t_i = \Delta t$
III (speed up by powering or coasting on a steep slope. speed down by coasting or braking)	$v_i^e \le v_{i+1}^m \le v_i^m$ Function needed: ① $f_1(t) = f_T(t)$ ② $f_2(t) = f_B(t)$ If $v_i^e \ge v_{ac}^c$ and $v_i^m \le v_i^c$, then $f_1(t) = f_{CA}(t)$ (speed up by coasting) If $v_{i+1}^m \ge v_{de}^c$ and $v_{i+1}^m \ge v_i^c$, then $f_2(t) = f_{CD}(t)$ (speed down by coasting) Calculate critical length: $v_i^e, v_{i+1}^m, 0, f_1(t) \xrightarrow{Algorithm1} S, \Delta t$ $L_{cr1} = S$ $v_i^e, v_{i+1}^m, t_0, f_1(t) \xrightarrow{Algorithm1} S, \Delta t$ $L_{cr2} = S$ $v_i^e, v_i^m, 0, f_1(t) \xrightarrow{Algorithm1} S_1, \Delta t_1$ $v_i^m, v_{i+1}^m, 0, f_2(t) \xrightarrow{Algorithm1} S_2, \Delta t_2$ $L_{cr3} = S_1 + v_i^m t_0 + S_2$	$L_i \le L_{cr1}$		$v_i^e, v_{i+1}^m, L_i, 0, f_1(t)$ ① $\xrightarrow{Algorithm2} v', \Delta t$ ② $v_i^D = v'$ (for the iteration) $v_i^l = v', t_i = \Delta t$
		$L_{cr1} < L_i \le L_{cr2}$		$v_i^e, v_{i+1}^m, L_i, t_0, f_1(t) \xrightarrow{Algorithm2}$ ① $v', \Delta t$ ② $v_i^D = v'$ (for the iteration) $v_i^l = v', t_i = \Delta t$
		$L_{cr2} < L_i \le L_{cr3}$		$v_i^e, v_{i+1}^m, v_i^m, t_0, L_i, f_1(t), f_2(t)$ ① $\xrightarrow{Algorithm3} v', \Delta t$ ② $v_i^l = v_{i+1}^m, t_i = \Delta t$
		$L_i > L_{cr3}$		$\Delta t = \Delta t_1 + \Delta t_2 + \frac{L_i - L_{cr3}}{v_i^m}$ ① ② $v_i^l = v_{i+1}^m, t_i = \Delta t$

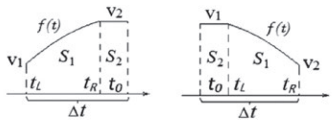
(*Continues*)

DOI: 10.1002/atr

Table I. (Continued)

[*] **Algorithm 1:**

Suitable v-t Graph:



Input: v_1, v_2, t_0, $f(t)\ t \in [0, t_{max}]$ Output: S (= $S_1 + S_2$), Δt

Solution Process:

① $F(t) = \int f(t)dt$

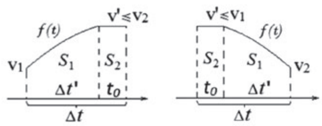
② $v_1 \xrightarrow{v=f(t)} t_L \in [0, t_{max}] \Rightarrow v_2 \xrightarrow{v=f(t)} t_R \in [0, t_{max}] \Rightarrow \Delta t = t_R - t_L + t_0$

③ If $v_1 \le v_2$ then $S = F(t_R) - F(t_L) + v_2 t_0$

If $v_1 > v_2$ then $S = F(t_R) - F(t_L) + v_1 t_0$

[**] Algorithm 2:

Suitable v-t Graph:



Input: v_1, v_2, S (= $S_1 + S_2$), t_0, $f(t)\ t \in [0, t_{max}]$ Output: v', Δt

Solution Process:

① $v_L = v_1, v_R = v_2$

② $v_c = (v_L + v_R)/2$

③ $v_1, v_c, 0, f_1(t) \xrightarrow{Algorithm1} S_1, \Delta t'$

$S_2 = v_c t_0$

⑤ If $S_1 + S_2 < S$, $v_L = v_C$

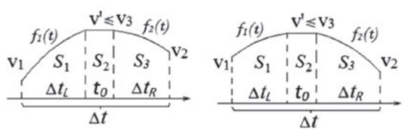
If $S_1 + S_2 > S$, $v_R = v_C$

⑥ If abs (($S_1 + S_2$)-S) > ε, go back to ②

⑦ v' = ($v_L + v_R$)/2, $\Delta t = \Delta t' + t_0$

[***] Algorithm 3:

Suitable v-t Graph:



Input: v_1,v_2,v_3, t_0, S (= $S_1 + S_2 + S_3$), $f_1(t)\ t \in [0, t_{max1}]$, $f_2(t)\ t \in [0, t_{max2}]$ Output: v', Δt

Solution Process:

① $v_L = v_1, v_R = v_3$

② $v_c = (v_L + v_R)/2$

③ If $v_C \le \max(v_1, v_2)$, then $v_c = (v_2 + v_R)/2$

④ $v_1, v_c, 0, f_1(t) \xrightarrow{Algorithm1} \Delta t_L, S_L$

$v_c, v_2, 0, f_2(t) \xrightarrow{Algorithm1} \Delta t_R, S_R$

⑤ If $S_L + S_R + v_c t_0 < S$, $v_L = v_C$

If $S_L + S_R + v_c t_0 > S$, $v_R = v_C$

⑥ If abs (($S_L + S_R$)-S) > ε, go back to ②

⑦ v' = ($v_L + v_R$)/2, $\Delta t = \Delta t_L + \Delta t_R + t_0$

[****] We need iteration for the first case of the Group I & III is for the requirement of logic of the programming.

(*Continues*)

Table I. (Continued)

$$v_c, v_2, 0, f_2(t) \xrightarrow{Algorithm1} \Delta t_R, S_R$$

⑤ If $S_L + S_R + v_c t_0 < S$, $v_L = v_C$

If $S_L + S_R + v_c t_0 > S$, $v_R = v_C$

⑥ If abs $((S_L + S_R)\text{-}S) > \varepsilon$, go back to ②

⑦ $v' = (v_L + v_R)/2$, $\Delta t = \Delta t_L + \Delta t_R + t_0$

**** We need iteration for the first case of the Group I & III is for the requirement of logic of the programming.

speed of coasting (denote as v_{de}^{c}) to be on a differentiated basis and set new nodes if the v–S curve crosses these velocity values in modeling, and finally achieve the result by iteration (Table I).

As shown by the tables earlier, the results of the polynomial equation can be calculated by means that solve the eigenvalue of Frobenius companion matrix or other similar ways (see Figure 6) [36]. Because the fitted dynamic characteristic curves are monotonic, at any point in $f(t)$, there must be a corresponding t value between 0 and $t_{\max}$. Because $F(t)$ can be gotten from $f(t)$ through integral computation and $f(t)$ is a monotonic function, which is equal to and greater than 0, $F(t)$ must monotonically increase. Thus, similarly, there must be a corresponding t value in $F(t)$ between 0 and $t_{\max}$.

5. EXAMPLE

5.1. Background

The profile diagram of a certain railway route with no tunnel is given next (as shown in Figure 7).The operation route of a certain locomotive is from A to B to C, and the train will stop for 2 minutes at station B. The route is controlled by a moving block signal system. For simplifying the calculation, we put dynamic braking and air braking together for consideration, and ignore the speed limit when

$$p(z) = a_0 + a_1 z + \cdots + a_{n-1} z^{n-1} + z^n$$

$$\Big\downarrow \ \det(zI - C) = p(z)$$

$$C = \begin{pmatrix} 0 & & & & -a_0 \\ 1 & 0 & & & -a_1 \\ & 1 & 0 & & -a_2 \\ & & \ddots & & \vdots \\ & & & 1 & -a_{n-1} \end{pmatrix} \xrightarrow{\text{eigen values of } \mathbf{C}} \text{roots of } p(z) = 0$$

Figure 6. Find roots of a polynomial by calculating eigenvalues of its Frobenious companion matrix [30].

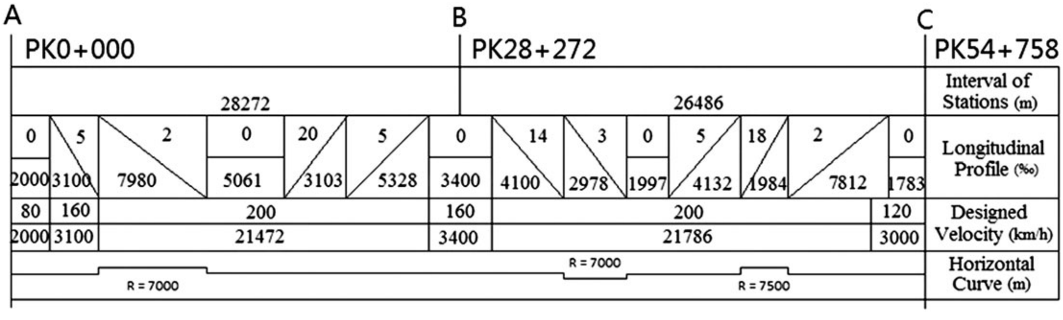


Figure 7. A railway route.

DOI: 10.1002/atr

Table II. Coefficients of fitting curves and related parameters.

		$f_T(t) = a_1t^3 + a_2t^2 + a_3t + a_4$								
i_s(‰)		−14	−5	−3	−2	0	2	5	18	20
R(m)		—	—	7000	7000	—	—	—	7500	—
i_s'(‰)		—	—	−2.91	−1.91	—	—	—	18.08	—
$f_T(t)$	**a1**	−1.36E − 06	−4.80E − 07	−3.20E − 07	−2.45E − 07	−1.05E − 07	2.21E − 08	1.84E − 07	2.12E − 07	1.74E − 07
	a2	−7.21E − 04	−6.58E − 04	−6.45E − 04	−6.39E − 04	−6.27E − 04	−6.15E − 04	−5.99E − 04	−3.76E − 04	−3.26E − 04
	a3	0.52	0.44	0.42	0.41	0.39	0.37	0.35	0.23	0.21
	a4	0	0	0	0	0	0	0	0	0
$t^T_{\max}$		148.32	190.83	204.32	211.92	229.25	255.26	302.36	744.22	763.40
$f_B(t)$	**b1**	−1.30E − 06	−1.26E − 06	−1.25E − 06	−1.24E − 06	−1.22E − 06	−1.19E − 06	−1.15E − 06	−9.00E − 07	−8.50E − 07
	b2	−8.81E − 04	−1.06E − 03	−1.10E − 03	−1.12E − 03	−1.16E − 03	−1.20E − 03	−1.26E − 03	−1.53E − 03	−1.57E − 03
	b3	−0.49	−0.57	−0.59	−0.60	−0.61	−0.63	−0.66	−0.78	−0.80
	b4	55.56	55.56	55.56	55.56	55.56	55.56	55.56	55.56	55.56
$t^B_{\max}$		96.04	83.88	81.59	80.49	78.37	76.37	73.55	63.41	62.10
$f_{CA}(t)$	**c1**	−2.84E − 08	1.40E − 09	4.01E − 10	1.38E − 10	0	0	0	0	0
	c2	−2.76E − 05	−1.30E − 05	−4.95E − 06	−2.08E − 06	0	0	0	0	0
	c3	0.12	0.04	0.02	0.01	0	0	0	0	0
	c4	0	0	0	0	0	0	0	0	0
$t^{CA}_{\max}$		571.83	3781.10	5341.01	6220.75	0	0	0	0	0
$f_{CD}(t)$	**d1**	0	−1.90E − 09	−2.01E − 09	−1.73E − 09	−3.90E − 09	−1.38E − 08	−3.42E − 08	−2.02E − 07	−2.39E − 07
	d2	0	8.92E − 06	1.33E − 05	1.36E − 05	2.29E − 05	4.47E − 05	7.16E − 05	1.79E − 04	1.95E − 04
	d3	0	−0.02	−0.03	−0.04	−0.05	−0.08	−0.11	−0.23	−0.25
	d4	0	55.56	55.56	55.56	55.56	55.56	55.56	55.56	55.56
$t^{CD}_{\max}$		0	2166.02	3354.17	4157.15	2805.66	1324.59	776.70	286.86	261.72
v^f		55.56	55.56	55.56	55.56	55.56	55.56	55.56	47.69	44.77
v^b		53.20	54.72	55.05	55.22	55.56	55.56	55.56	55.56	55.56
v^c		55.56	45.09	30.37	20.34	0	0	0	0	0

DOI: 10.1002/atr

entering or leaving the side track at stations. And the F–v curves are given through the following calculation (refer to China Railway High-speed 2).

$$F = 176 - 0.36v \quad (0 \le v \le 125 \ \text{km/hour})$$
$$F = 16\ 250/v \quad (v > 125 \ \text{km/hour}) \quad F\text{—kN}$$

Resistance of the locomotive:

$$W_0 = 8.63 + 0.07295v + 0.00112v^2 \quad W_0\text{—N}/t, v\text{—km/hour}$$

Braking force of the locomotive (N7 Gear):

$$F_B = 334 \quad (0 \le v \le 70 \ \text{km/hour})$$
$$F_B = 334 - 0.638(v - 70) \quad (70 \le v \le 200 \ \text{km/hour}) \quad F_B\text{—kN}$$

Additionally, when the train speed is over 160 km/hour, it accelerates by coasting when possible (i.e., v_{ac}^c = 160 km/hour). Also, when the train speed is over 120 km/hour, it decelerates by coasting when possible (i.e., v_{de}^c = 120 km/hour). From pulling operation mode to braking operation mode, the train needs 10-second transition time in coasting operation mode (i.e., t_0 = 10 seconds). The additional resistance of the train running on the horizontal curvature is [37]:

$$w_r = \frac{600}{R} (\text{N/kN})$$

where R is the radius of the circular curve.

5.2. *Encode elements*

We divide the whole route into 16 elements.

Element no.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
L (m)	2000	3100	7980	5061	3103	5328	1700	1700	4100	2978	1997	4132	1984	6595	1217	1783
v^D (km/hour)	80	160	200	200	200	200	160	160	200	200	200	200	200	200	120	120
i_s (‰)	0	−5	−2	0	20	5	0	0	−14	−3	0	5	18	2	2	0

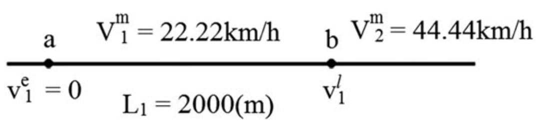


Figure 8. The element no. 1.

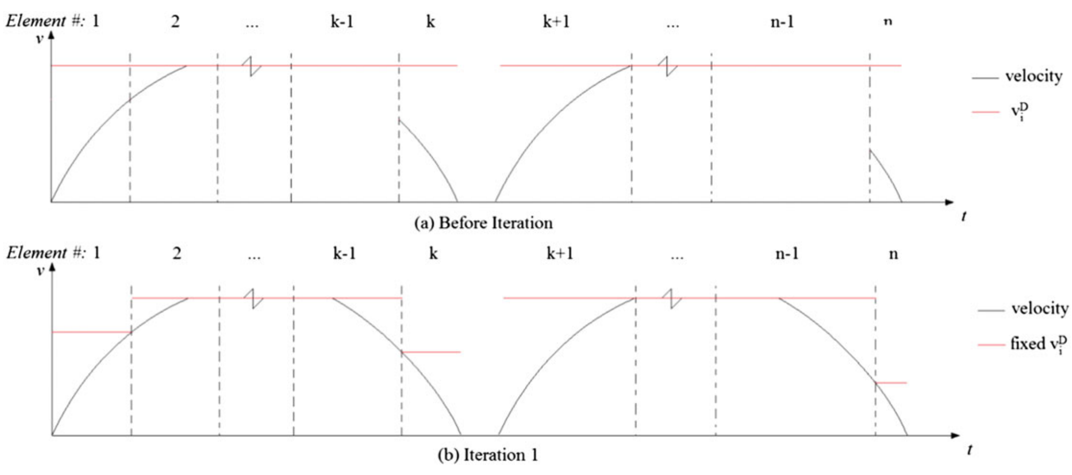


Figure 9. The effect of iteration.

DOI: 10.1002/atr

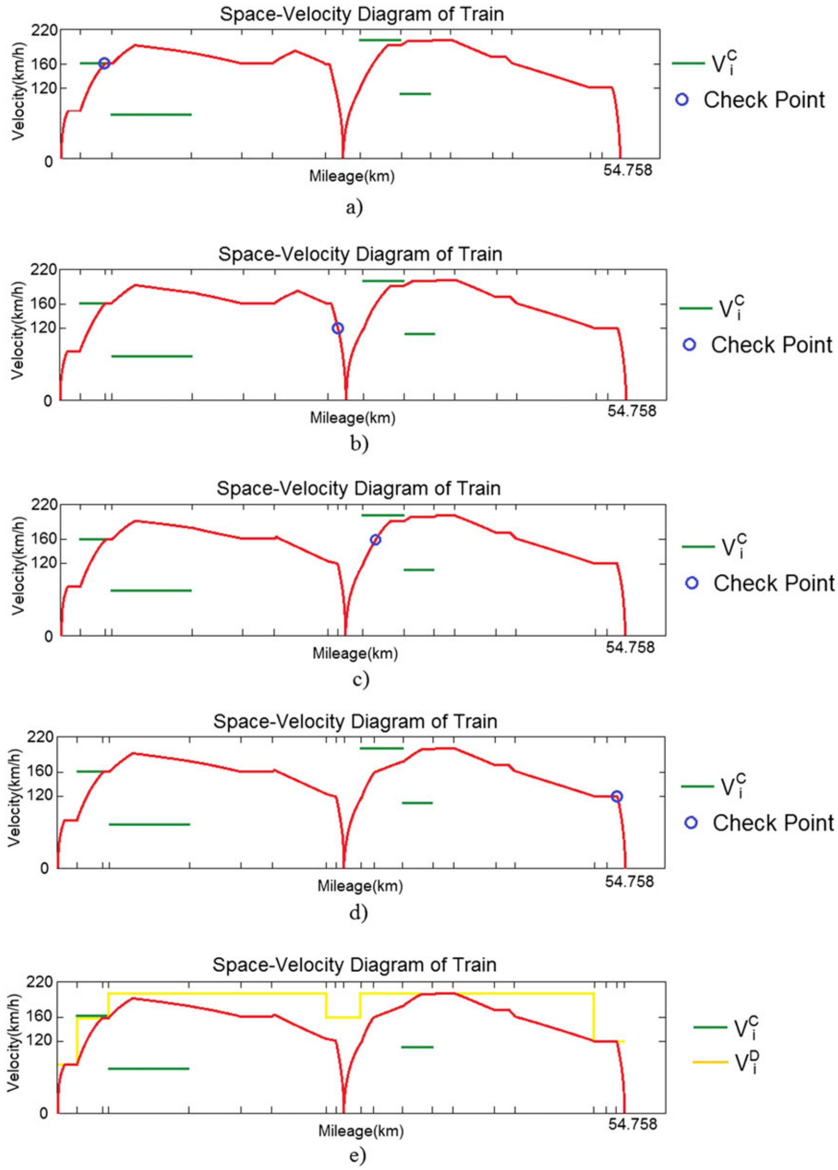


Figure 10. Schedule of the trains running in the interval of A - C. (a) iteration no. 1, (b) iteration no. 2, (c) iteration no. 3, (d) iteration no. 4, and (e) final curve.

5.3. Fitting curves

Because the figure of gradient permillage equals alignment resistance, we put the additional resistance when the locomotive is on circular curve w_r and the additional resistance when locomotive pass the tunnel (denote as w_t) together. In this case, based on the conditions of the longitudinal gradient and the radius of the circular curve, we can calculate the sum of w_r and w_t, and the result equals the new alignment resistance as well as the new gradient permillage [38]. Thus, we extract the parameters of traction, coasting, braking v–t matching curves whose gradients are −14, −5, −2.91, −1.91, 0, 2, 5, 18.08, and 20 and show them in Table II. Nevertheless, gradient braking limited velocity (denote as v^b) should be formulated by the realistic gradient.

5.4. Assembling by elements

We analyze the performance of each train in each element and calculate the time cost. All conditions have been enumerated in Table I. For example, take the first element to do calculation (Figure 8).

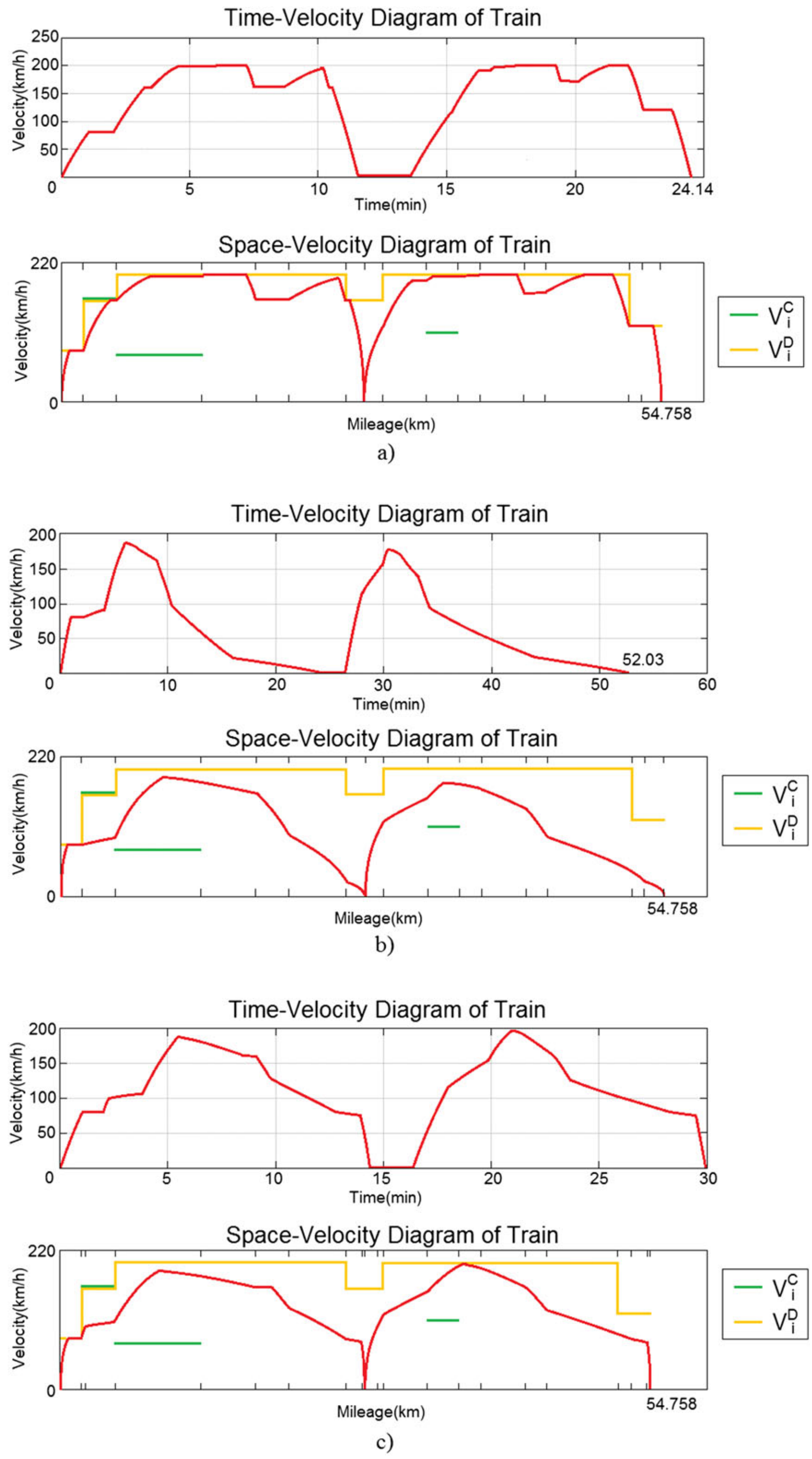


Figure 11. v–t and v–S curves for different situations of train operation. (a) Train running through the route as fast as possible, (b) train running through the route with absolutely energy saving, and (c) train running through the route within 30 minutes.

Given that:
$v_1^e = 0$ (m/s) $v_1^m = 22.22$ (m/s) $v_2^m = 44.44$ (m/s) $L_1 = 2000$ (m)
$\because v_1^e < v_1^m < v_2^m$ $\therefore$ It belongs to **Group I**, $L_{cr} = 728$(m).
$\because L_1 > L_{cr}$, belongs to **Case 2.**
$\therefore t_1 = 120.37$(s), $v_1^l = 22.22$ (m/s)

We need to pay attention to the possible requirement of iteration when calculating other elements using the similar method earlier. Based on the calculation result of each element, we choose the related *v–t* curve in Table II. Then, the curve parts are jointed to obtain the total curve relevant to the train operation, and the *v–S* curve is worked out by integrating.

The effect of iteration is to guarantee the continuity between two adjacent elements, which makes the whole simulation curve continuous. For example, a *v–t* curve of a train running on a route is showed in Figure 9, the curve formed by the first calculation is not continuous, but we make it so after doing iteration work. Generally, it only needs a few iterations to achieve, so it is a highly efficient process.

5.5. *Energy saving*

To solve the train energy-saving issue, many algorithms have been presented [39–45]. Yet, here is a tool that is suitable for EIM. After drawing a space–velocity diagram (*v–S* curve) from element nos. 1 to 16 by element increment, we search for the point whose value is equal to v_{ac}^c, v_{de}^c, and v_i^c, label the position as check points, and set a node here to cut the corresponding element. Then, we use EIM to calculate from the beginning again in order to find out whether there is another element that needs to be divided. Repeat this procedure until no further elements need to be divided. The iteration process is shown in Figure 10.

According to the value of v_{ac}^c, v_{de}^c, and v_i^c, classify the operation route through different efficiency. If $v_{ac}^c = v_{de}^c = 55.56$ km/hour, there is no coasting in the whole operation process, but the train does not rush up to upslope with full power, either. If so, the train will pass this route segment by the fastest way, yet it will spend more energy. If $v_{ac}^c = v_{de}^c = 0$ km/hour, the train tries the most to apply coasting to accelerate and decelerate. If so, the train operates in the most energy-saving way, yet it will spend more time. Therefore, in the realistic operation process, trains normally operate with a constant time cost required [46]. This operation method is more energy saving compared with the first one, and is more time saving compared with the second one. In the EIM algorithm, try to match the operational plan by adjusting $v_{ac}^c\, v_{de}^c$ and v_i^D. And Figure 11 shows the simulation curve in the fastest way, the most energy-saving way, and operation with exact 30 minutes. (In this example, the matching values are $v_{ac}^c = 100$ km/hour, $v_{de}^c = 75$ km/hour, and v_i^D is fixed.)

6. CONCLUSION

Element increment method is a novel method to do train movement simulation and work out simulation curves of train movement. It takes advantage of long-distance intervals (elements) as increment steps, which is different from the traditional train traction calculation that uses time intervals as increment steps. Thus, firstly, the calculation will create higher calculation efficiency, especially for railway routes that are straighter with higher design speed. Take the example earlier as illustration, it only needs to calculate in total 202 times ($16 \times 2 + 17 \times 2 + 18 + 19 \times 3 + 20 \times 2 + 21$) in EIM, instead of at least 1440 times in time increment method (set time increment as 1 second). Also, the EIM needs less room for data storage, the time increment method needs 1440 data points to form the *v–t* curve, but, in EIM, it only needs $21 \times 8 = 168$ data points ("8" refers to the eight parameters of each element: $v_i^e, v_i^m, v_i^l, L_i, t_0, \Delta t, f_1(t)$, and $f_2(t)$), and then the *v–t* and *v–S* curves can be drawn by on-time calculation. Secondly, the EIM can also search for all operation transmission points automatically with only a little of calculation demand. Thirdly, through adjusting v_{ac}^c, v_{de}^c, and v_i^c, it is available for users to adjust locomotives' operational plans to satisfy the demand of operation with a constant time cost required and achieve rational energy conservation. Therefore, compared with the current train traction

DOI: 10.1002/atr

calculation method, EIM works faster and is able to establish further optimized suggestion proposed by human-computer interaction.

However, EIM is limited in some aspects. Firstly, because of the assumption of a model of single particle, the result for short-length train will be better than that for long-length train. Secondly, classification of operation mode transforms need to be improved. Thirdly, in our case, the simulation is for moving block operational routes, so some extra factors will need to be considered if the simulation is for the automatic block operational route. Finally, EIM is more suitable to calculate the railway routes whose profiles are smooth and on which trains can operate a higher velocity.

In the future, further research of EIM should focus on how to use the model with multi-particle assumption and make the calculation results more accurate. Moreover, more indexes should be introduced, like the comfort index of passengers to consider the different feeling of passengers when we change the operational plan. And we can also develop the aided modeling system to build a calculation model of a complicated railway system. Additionally, the further research on improvement of automatic train operation, automatic train protection, or other train control systems could be developed by EIM.

7. LIST OF SYMBOLS AND ABBREVIATIONS

EIM	Element Increment Method
EMU	Electric Multiple Unit
CRH	China Railway High-speed
DC	Direct Current
ATP	Automatic Train Protection
t_i	the time cost for a train to pass the element i
$t_{\max}$	the maximum time of fitted v–t curve
t_0	the time cost for a train to switch operation mode from pulling to brake
L_i	the real length of the element i
L_{cr}	critical element length of categorizing cases for each element group
v_i^e	entrance velocity of the train on the element i
v_i^l	leaving velocity of the train on the element i
v_i^m	maximum velocity of the train on the element i
v_i^D	design velocity of the element i
v^t	top speed of a train on a slope
i_S	gradient of the element i
v_i^s	the maximum velocity on the slope of the element i
v_i^b	limit velocity for general braking on the slope of the element i
v_i^c	balanced velocity for coasting on the slope of the element i. If the velocity of a train is lower than v_i^c, it is able to speed up to v_i^c. If the train's velocity is greater than v_i^c, it is able to decelerate to v_i^c.
v_{ac}^c	critical velocity for coasting to accelerate, the train should take coasting if the velocity is greater than this value
v_{de}^c	critical velocity for coasting to decelerate, the train should take coasting if the velocity is greater than this value

ACKNOWLEDGEMENTS

This work was financially supported by the Foundation for Innovative Research Groups of the National Natural Science Foundation of China (no. 71621001), and the National Natural Science Foundation of China (nos. 71571018 and 71401008), the State key Laboratory of Rail Traffic Control and Safety (no. RCS2016ZQ001), and the Fundamental Research Funds for the Central Universities (no. 2016JBZ007).The authors wish to thank the anonymous referees and the editor for their comments and suggestions. The authors would also like to thank Mr. Jiandong Xu and Mr. Kishan Pai for their helpful revisions to the earlier version of this paper.

REFERENCES

1. Rao Z. *Train Traction Calculation*. China Railway: Beijing, China, 2005.
2. Martin P. Train performance simulation. In *2008 IET Professional Development Course on Electric Traction Systems*. IET: Manchester, 2008; 215–230.
3. American Railway Engineering Association. *Manual for Railway Engineering*. Washington DC, 1994.
4. Goodman C, Mellitt B, Rambukwella N. CAE for the electrical design of urban rail transit systems. In *Computers in Railway Operations*. Springer-Verlag New York, Inc, 1987.
5. Capillas Hedo E, Vadillo Vallejo J. Computer simulation of the basic parameters for designing an underground railway line. In *Computers in Railway Management*. Springer-Verlag New York, Inc, 1987.
6. Yinping F, Ziyou G, Keping L. Modeling study for tracking operation of subway trains based on cellular automata. *Journal of Transportation Systems Engineering and Information Technology* 2008; **8**(4):89–95.
7. Geng Z, Li X, Zhang B. Simulation study of heavy haul train operation on Datong–Qinhuangdao Railway. In *China Railway Science*. 2008; **29**(2):88–93.
8. Shi H, Guo H. CRH train traction calculation model and algorithm based on automatic constant speed. In *ICLEM 2012: Logistics for Sustained Economic Development–Technology and Management for Efficiency* 2012; 506–512.
9. Hwang H-S. Control strategy for optimal compromise between trip time and energy consumption in a high-speed railway. *IEEE Transactions on, Systems, Man and Cybernetics, Part A: Systems and Humans* 1998; **28**(6):791–802.
10. Bocharnikov Y, Tobias A, Roberts C, Hillmansen S, Goodman C. Optimal driving strategy for traction energy saving on DC suburban railways. *IET, Electric Power Applications* 2007; **1**(5):675–682.
11. Shirai Y, Ishihara Y. Teito rapid transit authority's automatic train operation. *Proceedings of the IEEE* 1968; **56**(4):605–615.
12. Malvezzi M, Allotta B, Rinchi M. Odometric estimation for automatic train protection and control systems. *Vehicle System Dynamics* 2011; **49**(5):723–739.
13. Dong H, Ning B, Cai B, Hou Z. Automatic train control system development and simulation for high-speed railways. *IEEE Circuits and Systems Magazine* 2010; **10**(2):6–18.
14. Kim DN, Schonfeld PM. Benefits of dipped vertical alignments for rail transit routes. *Journal of Transportation Engineering* 1997; **123**(1):20–27.
15. Kim K, Chien SI-J. Simulation-based analysis of train controls under various track alignments. *Journal of Transportation Engineering* 2010; **136**(11):937–948.
16. Dessouky MM, Leachman RC. A simulation modeling methodology for analyzing large complex rail networks. *SIMULATION* 1995; **65**(2):131–142.
17. Lai Y-C, Huang P-W. High-speed route improvement optimizer. *Transportation Research Record: Journal of the Transportation Research Board* 2012; (2289):18–23.
18. Ning Z, Roberts C, Hillmansen S, Nicholson G. A multiple train trajectory optimization to minimize energy consumption and delay. *IEEE Transactions on, Intelligent Transportation Systems* 2015; **16**(5):2363–2372.
19. Dorfman M, Medanic J. Scheduling trains on a railway network using a discrete event model of railway traffic. *Transportation Research Part B: Methodological* 2004; **38**(1):81–98.
20. Yang L, Li K, Gao Z, Li X. Optimizing trains movement on a railway network. *Omega* 2012; **40**(5):619–633.
21. Li F, Gao Z, Li K, Yang L. Efficient scheduling of railway traffic based on global information of train. *Transportation Research Part B: Methodological* 2008; **42**(10):1008–1030.
22. Mu S, Dessouky M. Efficient dispatching rules on double tracks with heterogeneous train traffic. *Transportation Research Part B: Methodological* 2013; **51**:45–64.
23. Yang L, Li S, Gao Y, Gao Z. A coordinated routing model with optimized velocity for train scheduling on a single-track railway line. *International Journal of Intelligent Systems* 2015; **30**(1):3–22.
24. Howard SM, Gill LC, Wong PJ. Review and assessment of train performance simulation models. *Transportation Research Record* 1983; (917): 1–6.
25. Uher RA, Disk DR. A train operations computer model. In *Computers in Railway Operations*. Springer-Verlag New York, Inc., 1987.
26. Kikuchi S. A simulation model of train travel on a rail transit line. *Journal of Advanced Transportation* 1991; **25**(2):211–224.
27. Zhang J, Ni S, Ge L, Lv M. Electric multiple units train traction calculating system. In *ICTE 2013: Safety, Speediness, Intelligence, Low-Carbon, Innovation* 2013; 1759–1764.
28. Nash A, Huerlimann D. Railroad simulation using OpenTrack. In *Computers in Railways IX*. WIT Press, 2004; 45–54.
29. Radtke A, Bendfeldt J. Handling of railway operation problems with RailSys. In *Proceedings of the 5th World Congress on Rail Research*. 2001.
30. Jong J-C, Chang S. Algorithms for generating train speed profiles. *Journal of the Eastern ASIA Society for Transportation Studies* 2005; **6**:356–371.
31. Ho TK, Mao B, Yuan Z, Liu H, Fung Y. Computer simulation and modeling in railway applications. *Computer Physics Communications* 2002; **143**(1):1–10.
32. Ho TK, Goodman C, Norton J. An event-based traffic flow model for traffic control at railway junctions. In *Proceedings of IRSE International Conference on Advanced Railway Control (ASPECT'95)*. IRSE, 1995; 163–170.
33. Goodman C, Siu L, Ho TK. A review of simulation models for railway systems. In *International Conference on Developments in Mass Transit Systems 1998*. IET, 1998: 80–85.

DOI: 10.1002/atr

34. Li K, Gao Z, Ning B. Cellular automaton model for railway traffic. *Journal of Computational Physics* 2005; **209**(1):179–192.
35. Vincze B, Tarnai G. Development and analysis of train brake curve calculation methods with complex simulation. *Advances in Electrical and Electronic Engineering* 2011; **5**(1–2):174–177.
36. Edelman A, Murakami H. Polynomial roots from companion matrix eigenvalues. *Mathematics of Computation* 1995; **64**(210):763–776.
37. *TB/T 1407–1998 Train Traction Calculation Procedures*. China Railway Publishing House: Beijing, 1998.
38. Rakha H, Lucic I, Demarchi S, Setti J, Aerde M. Vehicle dynamics model for predicting maximum truck acceleration levels. *Journal of Transportation Engineering* 2001; **127**(5):418–425.
39. Howlett PG, Milroy IPPudney PJ. Energy-efficient train control. *Control Engineering Practice* 1994; **2**(2):193–200.
40. Liu RF, Golovitcher IM. Energy-efficient operation of rail vehicles. *Transportation Research Part A-Policy and Practice* 2003; **37**(10):917–932.
41. Su S, Tang T, Chen L, Liu B. Energy-efficient train control in urban rail transit systems. *Proceedings of the Institution of Mechanical Engineers Part F-Journal of Rail and Rapid Transit* 2015; **229**(4):446–454.
42. Ciccarelli F, Iannuzzi D, Tricoli P. Control of metro-trains equipped with onboard supercapacitors for energy saving and reduction of power peak demand. *Transportation Research Part C: Emerging Technologies* 2012; **24**:36–49.
43. Howlett P. Optimal strategies for the control of a train. *Automatica* 1996; **32**(4):519–532.
44. Xu X, Li K, Yang L. Rescheduling subway trains by a discrete event model considering service balance performance. *Applied Mathematical Modelling* 2016; **40**(2):1446–1466.
45. Yin J, Tang T, Yang L, Gao Z, Ran B. Energy-efficient metro train rescheduling with uncertain time-variant passenger demands: an approximate dynamic programming approach. *Transportation Research Part B: Methodological* 2016; **91**:178–210.
46. Zhang J, Guo X, Chen T, Jin J. Optimized algorithm for train traction calculation under fixed-time mode. In *International Conference on Transportation Engineering 2009*. 2009; 3760–3765.

DOI: 10.1002/atr