Transportation Research Part C 93 (2018) 544–565

Contents lists available at ScienceDirect

Transportation Research Part C

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

Position synchronization for track geometry inspection data via big-data fusion and incremental learning

Yuan Wang[a,b,c], Ping Wang[a,b,*], Xin Wang[a,b], Xiang Liu[c]

[a] School of Civil Engineering, Southwest Jiaotong University, Chengdu, China
[b] Key Laboratory of High-speed Railway Engineering, Ministry of Education, Chengdu, China
[c] Department of Civil and Environmental Engineering, Rutgers, The State University of New Jersey, NJ, USA

ARTICLE INFO

Keywords:
Track geometry inspection
Railway
Big data
Information fusion
Position synchronization
Incremental learning

ABSTRACT

Track geometry inspection data is important for managing railway infrastructure integrity and operational safety. In order to use track geometry inspection data, having accurate and reliable position information is a prerequisite. Due to various issues identified in this research, the positions of different track geometry inspections need to be aligned and synchronized to the same location before being used for track degradation modeling and maintenance planning. This is referred to as "position synchronization", a long-standing important research problem in the area of track data analytics. With the aim of advancing the state of the art in research on this subject, we propose a novel approach to more accurately and expediently synchronize track geometry inspection positions via big-data fusion and incremental learning algorithms. Distinguishing it from other relevant studies in the literature, our proposed approach can simultaneously address data exceptions, channel offsets and local position offsets between any two inspections. To solve the Position Synchronization Model (PS-Model), an Incremental Learning Algorithm (IL-Algorithm) is developed to handle the "lack of memory" challenge for the fast computation of massive data. A case study is developed based on a dataset with data size of 18 GB, including 58 inspections between February 2014 and July 2016 over 323 km (200 miles) of tracks belonging to China High Speed Railways. The results show that our proposed model performs robustly against data exceptions via the use of multi-channel information fusion. Also, the position synchronization error using our proposed approach is within 0.15 meters (0.5 feet). Our proposed data-driven, incremental learning algorithm can quickly solve the complex, data-extensive, position synchronization problem, using an average of 0.1 s for processing one additional kilometer of track. In general, the data analysis methodology and algorithm presented in this paper are also suitable to address other relevant position synchronization problems in transportation engineering, especially when the dataset contains multiple channels of sensors and abnormal data outliers.

1. Introduction

Track geometry defects are considered one of the most important factors in operational stability and safety (Esveld, 2001; Higgins and Liu, 2017; Liu et al., 2013; Quiroga and Schnieder, 2012). Track geometry data from track inspection cars is useful for railway maintenance. There are multiple inspection channels corresponding to different types of track geometry, and each channel relates to a

* Corresponding author at: School of Civil Engineering, Southwest Jiaotong University, Chengdu, China.
E-mail address: wping@home.swjtu.edu.cn (P. Wang).

https://doi.org/10.1016/j.trc.2018.06.018
Received 27 September 2017; Received in revised form 22 June 2018; Accepted 27 June 2018

Table 1
Selected track inspection parameters and methods.

Channel	Type	Sketch Map	Measurement Method	Sensors
1	Gauge	Fig. 1 (1)	Laser ranging	Laser and displacement transducer
2	Longitudinal profile (two sides)	Fig. 1 (4)	Inertial method	Accelerometer and displacement transducer
3	Alignment (two sides)	Fig. 1 (3)		
4	Crosslevel	Fig. 1 (2)	Automatic acceleration compensation	Accelerometer
5	Warp (twist)	–	Difference of crosslevel with a distance of 3 meters	Calculated from crosslevel

specific type of sensor. Taking the GJ-4 track inspection car of the Chinese Ministry of Railways as an example, some track inspection parameters are listed in Table 1 (Ren et al., 2010). The illustrative sketches of different types of track geometry are shown in Fig. 1. Each channel corresponds to a specific type of track geometry.

There have been many studies based on track inspection data, including data measurement (Haigermoser et al., 2015; Weston et al., 2007; Bocciolone et al., 2007; Tsunashima, 2008), track condition evaluation (Tsunashima, 2008; Alfelor et al., 2001; Sadeghi, 2010; Sadeghi and Askarinejad, 2011) and track degradation prediction (Kawaguchi et al., 2005; Bartram et al., 2008; Liu et al., 2010; Xu et al., 2011, 2012; Xu, 2012; Selig et al., 2008). Nearly all the methods and models require high quality inspection data. The use of raw track geometry inspection data from the track geometry car is not always valid due to various data issues, such as measurement errors, abnormal data outliers and positional errors. Among these errors, milepost positional error is one common issue, requiring extensive effort to match and align the positions of the same inspected location from multiple inspections (Xu, 2012; Selig et al., 2008; Qu, 2012; Xu et al., 2013). This effort is not trivial because of the need for estimating and predicting location-specific track geometry deterioration in railroad track maintenance planning. This paper aims to address position synchronization problem from different inspection runs, with a high precision and computational efficiency. The research outcomes can be used for all types of railway systems, particularly high-speed railways, whose track asset management demands a high accuracy in positional information.

In practice, an initial milepost can be manually selected. The subsequent mileage information is obtained according to the rotation angles (by counting the grating encoder impulse number) and the wheel radius (Allotta et al., 2002), as illustrated in Fig. 2a. However, there are inevitable positional errors caused by radial errors of the wheels, faulty encoder output (Qu, 2012), degraded adhesive conditions (Soleimani and Moavenian, 2017, Liu and Bruni, 2015) or track geometry irregularities (Fig. 2d). Due to these factors, the positional error accumulates. To address these issues, the Global Positioning System (GPS) (Specht et al., 2017; Tsunashima, 2008; Allotta et al., 2002), Differential GPS (DGPS) (Allotta et al., 2002; Hanreich et al., 2002) and radio-frequency identification (RFID) (Yang, 2009) are introduced as an absolute reference to control the accumulation of positional errors.

Even though many advanced techniques and devices are used, the positional errors cannot be eliminated and can sometimes reach 100 meters (328 feet). Furthermore, other environmental conditions could lead to abnormal data points. For example, a film of water from rain on the rail-head can cause laser sensor malfunction (Fig. 2c). This kind of abnormal data outlier may influence the performance of the position synchronization method. In Section 2, we review the related work in the literature that addresses this research problem, the respective merit and limitations of each method and the intended contributions of our proposed new approach to the body of knowledge.

2. Related prior work

Positional errors can be classified into three categories, which are (1) absolute position errors (APE); (2) relative position errors (RPE); and (3) channel-inside position offset (CPO). Since our study focuses on position synchronization of data from different runs with multiple measurement channels, our review focuses on RPE and CPO. It should be noted that position synchronization is only focus on RPE and CPO. The track inspection dataset used in this paper has undergone a preliminary processing based on the Key Equipment Identification (KEI) model proposed in Xu et al. (2013).

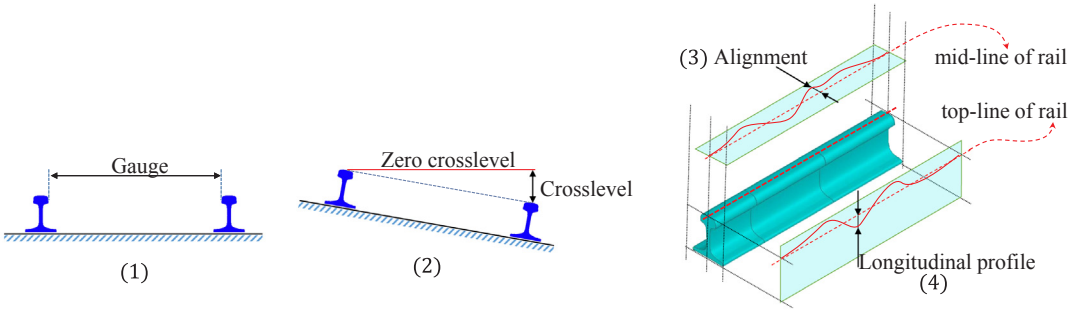

Fig. 1. Schematic diagrams of track gauge, crosslevel, alignment and.

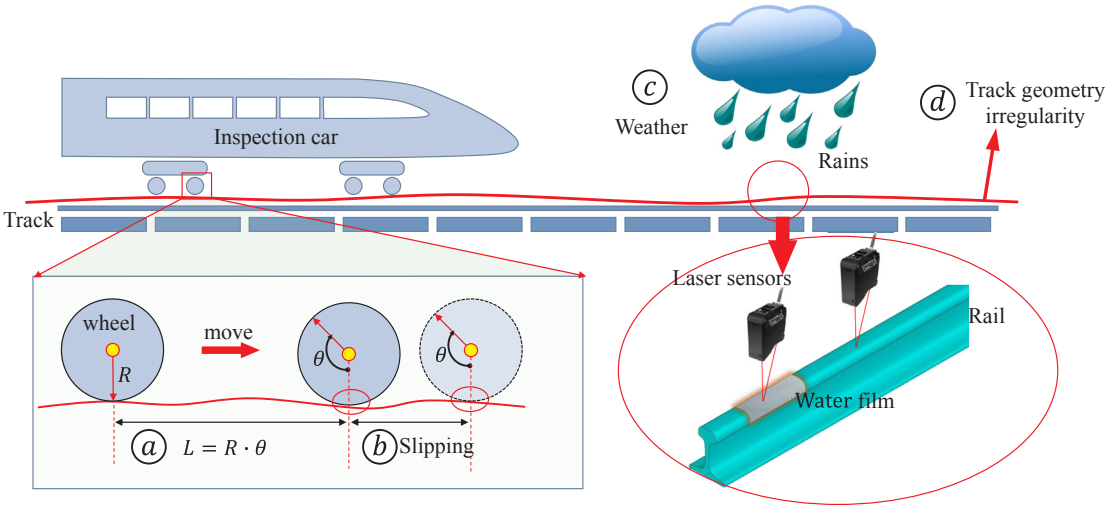

Fig. 2. Positioning principle of the inspection car and the causes of the positional errors. (*a*) shows the mileage measuring principle; (*b*) shows the degraded adhesive conditions that leads to relative slippage between rail and wheel; (*c*) shows unforeseen influences such as weather; (*d*) track geometry irregularity.

2.1. Absolute Position Error (APE)

The positional difference of the inspection data compared to its actual position is called the Absolute Position Error (APE). The absolute positional accuracy is important when a worker attempts to locate a track defect that is observed in the inspection. The APE magnitude is determined by absolute reference along the track. Since there are inevitable errors in the selected reference, the APE cannot be eliminated. In addition to using better inspection technologies, researchers have developed mathematical models to deal with APE. For example, optimization models were established to minimize the sum of the squares of the difference between two sets of selected track geometry data from a certain section (Sui, 2009). Models are established to correct the local offset of curve sections based on the least square method and correlation coefficient (Li and Xu, 2010; Pedanekar, 2006). More recently, a key equipment identification model was proposed to correct the APE of track inspection data by combining the real mileage information of track equipment, which can reduce the APE to under 5 meters (Xu, 2012; Xu et al., 2013). There are more details regarding APE in Vu et al. (2012).

2.2. Relative position error (RPE)

The mileage difference in inspection data between different runs is known as the Relative Position Error (RPE). The RPE exists because of uncertain rail-wheel contact profiles for different inspection cars running along the same track section. The magnitude of RPE is determined by comparing the data from multiple inspections. In the scope of this paper, the process to correct RPE is treated as a problem of position synchronization. The literature concerning RPE is summarized in Table 2. Addressing RPE is one focus of this paper.

Some common issues with the methods mentioned in Table 2 can be summarized as follows:

- A quantitative assessment model of the RPE is lacking. A general approach to address the position error is to observe the coincidence of waveforms from graphs (Xu et al., 2013; Sui, 2009; Li and Xu, 2010). Xu (2012) applies two indirect assessments, including the correlation coefficient and summation of gauge change between the measured waveforms of two inspections, to address the performance of mileage corrections. Xu et al. (2015) and Xu et al. (2016) present an indirect measurement by using standard deviation of the inspection data of different runs. A smaller standard deviation indicates better performance of the model.
- The reference for position synchronization is determined both subjectively and empirically. Xu et al. (2015), Sui (2009) and Xu et al. (2016) use the latest set of the previous inspection data as a reference to synchronize the current inspection data. Li and Xu (2010) and Pedanekar (2006) use a reference data library or some static files, which are generated by railway operators and will be updated when maintenance is carried out.
- All the aforementioned methods are based on a default assumption that the inspection data has no data exception issues, or the dataset is thought to be exception-free after preprocessing. However, that is not always the case. Some models may become invalid or even lead to erroneous positioning results, since the position error due to data exceptions in the reference data will spread to current processed data.
- Only a single measurement channel is used for position synchronization in the above literatures. Abnormal data points may exist in one channel, but the probability of data exceptions in all channels at the same position is low. The performance can be enhanced by fusing data from multiple channels in one unified model (Section 5.3).

Table 2
Previous studies related to RPE.

Reference	Application scenarios	Quantitative assessment model of RPE	Main techniques	Reference for position synchronization	Consideration of data exceptions	Channels in use
Xu (2012)	Railway	Indirect assessment	Dynamic time warping	The latest inspection run	Preprocessing	Track gauge
Xu et al. (2015)	Railway	Indirect assessment	Dynamic time warping	The latest inspection run	Preprocessing	Track gauge
Sui (2009)	Railway	Not mentioned	Optimization	The latest inspection run	Not mentioned	Profile, alignment or gauge (one channel)
Li and Xu (2010)	Railway	Not mentioned	Least square method	Reference data library	Not mentioned	Profile, alignment or gauge (one channel)
Pedanekar (2006)	Railway	Not mentioned	Correlation coefficient	Static files	Not mentioned	–
Xu et al. (2016)	Subway	Indirect assessment	Dynamic Programming and Cross-correlation	The latest inspection run	Preprocessing	Profile, alignment, gauge, twist, and cross-level (separately)

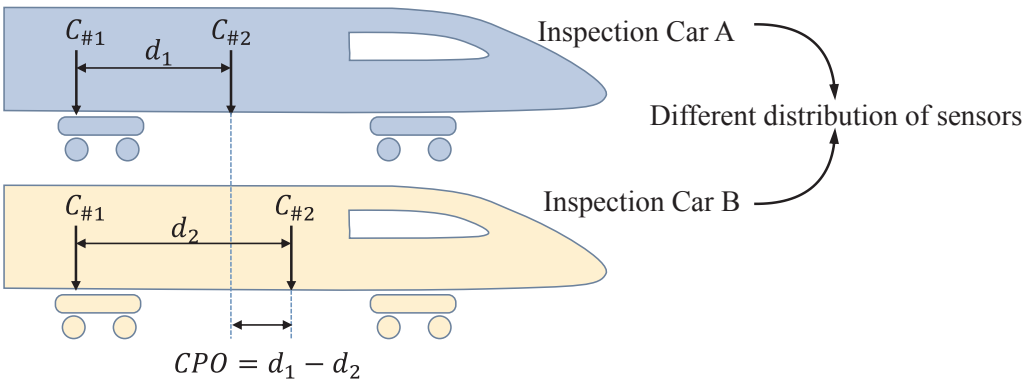

Fig. 3. Sketch map of the CPO and track geometries. $C_{\#1}$ and $C_{\#2}$ represent two different types of track geometry; d_1 and d_2 are the relative distances between the measurement locations of $C_{\#1}$ and $C_{\#2}$, for inspection Car A and Car B, respectively.

2.3. Channel-inside Position Offset (CPO)

The Channel-inside Position Offset (CPO) is the position difference of inspection channels between different inspection runs. It is derived from differences in the distribution of sensors between different inspection cars. Generally, the sensors are not mounted at the same location of the inspection car. As illustrated in Fig. 3, two different types of inspection car, Car A and Car B, measure the same types of track geometry, $C_{\#1}$ and $C_{\#2}$. The relative distances between the measurement locations of sensors, d_1 and d_2, can be different. The CPO is the difference between d_1 and d_2. The existence of CPO between inspection Car A and Car B is caused because the two inspection cars are designed differently.

To our knowledge, there are no published studies concerning CPO. The concept of CPO in the aforementioned Xu (2012), Xu et al. (2015), Sui (2009), Li and Xu (2010), Pedanekar (2006) and Xu et al. (2016) does not exist since they only consider the data from a single measurement channel. On the contrary, in this paper we attempt to deal with data exception issues by fusing data from multiple sensors into one unified model. Because it is very rare that all sensors malfunction at the same time (unless there are inspection-car-specific failures), multi-sensor fusion can make our proposed algorithm more robust against sensor data outliers in aligning track inspection data positions.

3. Contributions and organization of this paper

Considering all the limitations of the prior research presented in Sections 2.2 and 2.3, this paper develops a novel approach to addressing the problem of position synchronization via big data fusion and incremental learning algorithms.

Firstly, the local milepost offset and waveform similarity between every two inspection runs are estimated based on definitions given in Section 5.2. Secondly, to deal with data exceptions, the inspection data of multiple measurement channels are fused by MCF-model in Section 5.3. The channel fusion process improves the precision of position synchronization and enhances the robustness against abnormal data. The side-effect is that the required memory and computation time increase a lot comparing to previous methods (Xu et al., 2015, 2016; Sui, 2009; Li and Xu, 2010; Pedanekar, 2006), where only the data of one single channel is used.

As a countermeasure, in Section 6, we propose the concept of *knowledge library*, which represents the minimal information refined from the *overall information* with a high-dimensional structure. The refining process compresses the overall information into several low-dimensional vectors and matrices by statistical approaches. Whenever a new dataset is obtained from the track inspection car, the RPE and the CPO of the newly measured data are estimated as referring to historical knowledge, and then position synchronization is carried out to reduce the RPE and CPO. In return, the knowledge library will be updated. The implement of the updating process of knowledge library is called IL-algorithm.

For a better understanding of this paper's work, its organization is illustrated in Fig. 4. The contents with a red border are the main contributions, which are also summarized as the following six points.

- The inspection data from multiple measurement channels are fused and synchronized through a multi-channel fusion model established in Section 5.3. A quantitative assessment of RPE is achieved through an optimization model proposed in Section 5.4.
- This paper presents a novel approach to dealing with data exceptions through the establishment of three matching criteria with the thresholds determined through statistical methods presented in Section 5.5.
- The position synchronization operation is achieved through a two-phase interpolation approach presented in Section 5.6.
- An incremental learning algorithm is developed to execute position error estimating, multi-channel fusing and position synchronizing processes in Section 6. The required computational time is minimized via these advanced data processing and analytic algorithms.
- A case study is developed based on real-life data from part of the China High Speed Railway to demonstrate the practical values of this research for industrial practice (Section 7).
- The tradeoff between the computation efficiency and the accuracy of position synchronization is discussed in Section 8.

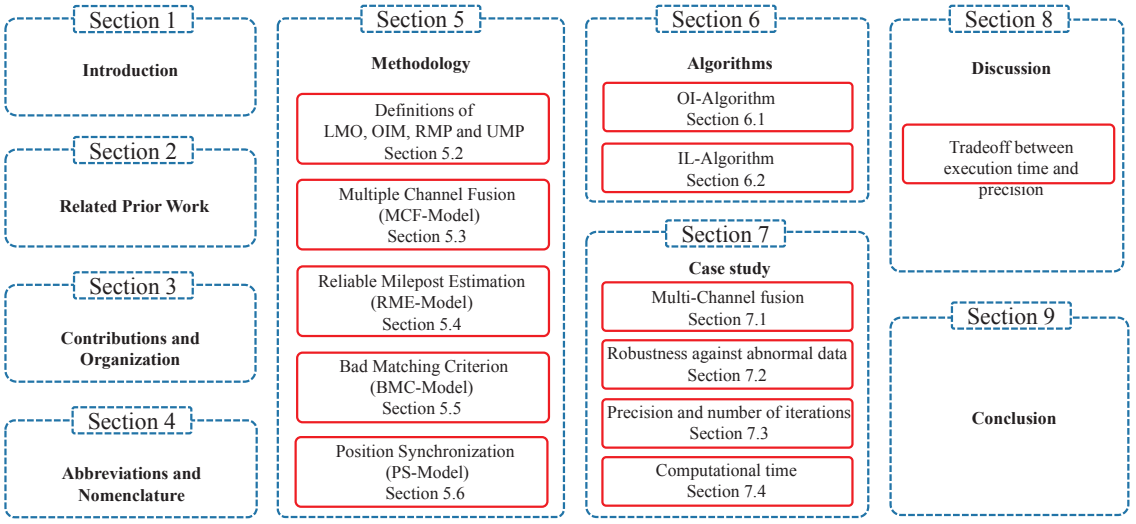

Fig. 4. Organization of this paper.

4. Abbreviations and nomenclature

See Table 3.

5. Methodology

5.1. Technical framework

The purpose of this section is to introduce the overall model framework, which contains five parts, one part is model preparation with some definitions and four sub-models, as illustrated in Fig. 5. The four models, Multiple Channel Fusion Model (MCF-Model), Reliable Milepost Estimate Model (RME-Model), Bad Matching Criterion Model (BMC-Model) and Position Synchronization Model (PS-Model), are established in order of the dependency relationship. The purposes of each process are presented below.

- The first process is aimed at estimating the local offsets and waveform similarities between every two runs of inspection data. (Section 5.2)
- The MCF-Model is established to estimate and fuse the CPO among different inspection channels. (Section 5.3)
- The RME-Model is established to estimate the RPE by fusing the output results of the MCF-Model. (Section 5.4)
- The BMC-Model is established to deal with data exceptions by filtering the CPO and RPE. Three criteria are defined and the thresholds are given according to probability distributions of local offsets and waveform similarities. (Section 5.5)
- The PS-Model is established to conduct position synchronization operation according to a two-phase interpolation. (Section 5.6)

5.2. Fundamentals of the models

This section presents some basic definitions concerning the local offsets between every two runs, including Local Milepost Offset (LMO), Overall Information Matrix (OIM), Reliable Matching Point (RMP) and Unreliable Matching Point (UMP).

5.2.1. Local Milepost Offset

Local Milepost Offset (LMO) is defined to describe the milepost difference between two data samples, data #1 and data #2 in Fig. 6. If there is no milepost offset between the two sets of data, this situation is referred to as an ideal mapping, illustrated in Fig. 6a. Real mapping is different from ideal mapping because of RPE. For example, the data sample R_1 in data #1 and R_2 in data #2 share the same milepost range, as in curve A-B in Fig. 6a. When R_1 is longer than R_2, it indicates data #1 is compressed in comparison to data #2 within the A-B range. Therefore, the LMO is defined as the difference between real mapping and ideal mapping, see Fig. 6b.

The following presents a mathematical definition of LMO. For discrete signal $X = \{x_i | i = 1, 2, ..., N\}$ and $Y = \{y_i | i = 1, 2, ..., N\}$, in considering a local waveform matching scale s, the local samples of X and Y at location k are defined as

$$\begin{cases} X^{(s,k)} = \left\{x_i | k - \frac{s}{2} + 1 \leqslant i < k + \frac{s}{2}\right\} \\ Y^{(s,k)} = \left\{y_i | k - \frac{s}{2} + 1 \leqslant i < k + \frac{s}{2}\right\} \end{cases} \tag{1}$$

For $i\langle 1 \text{ or } i\rangle N$, $x_i, y_i = 0$ the local offset $\delta^{(s,k)}$ and waveform similarity $\rho^{(s,k)}$ of Y to X are defined as follows

Table 3
Abbreviations and Nomenclature used in this Paper.

Abbreviations	Explanation
APE	Absolute Position Error
RPE	Relative Position Error
CPO	Channel-inside Position Offset
RMP	Reliable Matching Point
UMP	Unreliable Matching Point
RME	Reliable Milepost Estimate
LMO	Local Milepost Offset
OIM	Overall Information Matrix
MCF-Model	Multiple Channel Fusion Model
RME-Model	Reliable Milepost Estimate Model
BMC-Model	Bad Matching Criterion Model
PS-Model	Position Synchronization Model
OI-Algorithm	Overall-Iterative Algorithm
IL-Algorithm	Incremental-Learning Algorithm

Notation	Explanation
s	The scale parameter for waveform matching
d_s	The step for waveform matching, 0.25 m per point; point by point if $d_s = 1$.
δ	Local mileage offset
ρ	Local similarity, ranging from $[-1, 1]$.
k	Refers to a mileage position
c, c^*	A data channel, c^* indicates the best channel to be selected.
$\delta^{(s,k)}(X, Y)$	The mileage offset of X to Y at location k under scale s.
$\rho^{(s,k)}(X, Y)$	The similarity of X to Y at location k under scale s.
$\delta_{ij,c}^{(k)}$	The mileage offset of the jth run to the ith run at location k based on data from channel c.
$\rho_{ij,c}^{(k)}$	The similarity of the jth run to the ith run at location k based on data from channel c.
$\boldsymbol{\Delta}_c^{(k)}(\boldsymbol{X})$, $\Delta_{\boldsymbol{c}}(\boldsymbol{X})$	The local offset matrix and overall offset matrix at location k based on data from channel c. Where X is a matrix containing data of multiple runs.
$\Upsilon_c^{(k)}(\boldsymbol{X})$, $\boldsymbol{\Upsilon}_c(\boldsymbol{X})$	The local similarity matrix and overall similarity matrix at location k based on data from channel c. Where X is a matrix containing data of multiple runs.
$\boldsymbol{U}$	$= \{(\delta_{i,j,c}^{(k)}, \rho_{i,j,c}^{(k)}) \mid i, j \in [1, n]; k \in [1, N/d_s]; c \in [1, t]\}$, the Overall Information Matrix. Also written as $\{(\boldsymbol{\Delta}_c(\boldsymbol{X}), \boldsymbol{\Upsilon}_c(\boldsymbol{X})) \mid c = 1, 2, \ldots, t\}$.
$C_{i,j,c}$	The overall channel offset of the jth run to the ith run for channel c.
$D_{i,c}$	The estimation of overall channel offset for the ith run.
$d_i^{(k)}$, $d_i^{(k)*}$	The RME of the ith run at the location k. * indicates the best selected channel.
$\dot{d}_i^{(k)*}$	The first-order derivative of $d_i^{(k)*}$.
δ_0, ρ_0, δ_1	The thresholds of$\delta_{ij,c}^{(k)}$, $\rho_{ij,c}^{(k)}$ and $\dot{d}_i^{(k)*}$ for the BMC.
r_{UMP}	$= \frac{\mathbf{card}(\{\mathbf{bo}(\delta_{i,j}^{(k)}, \rho_{i,j}^{(k)}) = 0 \mid i, j \in [1, n]; k \in [1, N/d_s]; c \in [1, t]\})}{\mathbf{card}(\boldsymbol{U})}$, the proportion of the UMPs.
$u_i^{(k,n)}$	The average of the LMO of the ith run at location k, considering a total of n runs of inspection data.
$n_i^{(k,n)}$	The count of the RMPs of the ith run at location k, considering a total of n runs of inspection data.
$\boldsymbol{K}^{(k,n)}$	$= \{(u_i^{(k,n)}, n_i^{(k,n)}) \mid i, j \in [1, n]; k \in [1, N/d_s]\}$, local information matrix.
f_r	The distribution of variable r, where r can be δ, ρ or $\dot{d}$.
M	A parameter of IL-Algorithm, which represents the maximal number of inspection runs that are to be cached in memory.
$\boldsymbol{P_r}(p)$	The precision of position synchronization at a given confidence level p.

Operator	Explanation
$\mathbf{norm}(\boldsymbol{x})$	$= \|\boldsymbol{x}\|_2 = \sqrt{\sum \|x_i\|^2}$, vector norm.
$\mathbf{cov}(\boldsymbol{x}, \boldsymbol{y})$	$= \sum (x_i - \bar{x})(y_i - \bar{y})$, covariance.
$\mathbf{Percentile}(\boldsymbol{x}, p)$	The percentile value of $\boldsymbol{x}$ at a percentage of p.
$\mathbf{corr}(\boldsymbol{x}, \boldsymbol{y})$	$= \frac{\mathbf{cov}(\boldsymbol{x}, \boldsymbol{y})}{\mathbf{norm}(\boldsymbol{x} - \bar{x}) \cdot \mathbf{norm}(\boldsymbol{y} - \bar{y})}$, normalized correlation coefficient.
$\mathbf{xcorr}2(\boldsymbol{A}, \boldsymbol{B})$	2D cross-correlation of matrix **A** and **B**.
$\mathbf{card}(\{\cdot\})$	Number of elements in set $\{\cdot\}$.
$\mathbf{interp}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{x}')$	Resampling the sequence $(\boldsymbol{x}, \boldsymbol{y})$ by a new $\boldsymbol{x}'$ through interpolation.
$\mathbf{bo}(\delta, \rho)$	$= \begin{cases} 1, & \text{if } \rho \geqslant \rho_0 \text{ and } \lvert\delta\rvert \leqslant \delta_0 \\ 0, & \text{otherwise} \end{cases}$, exception judgment operation.

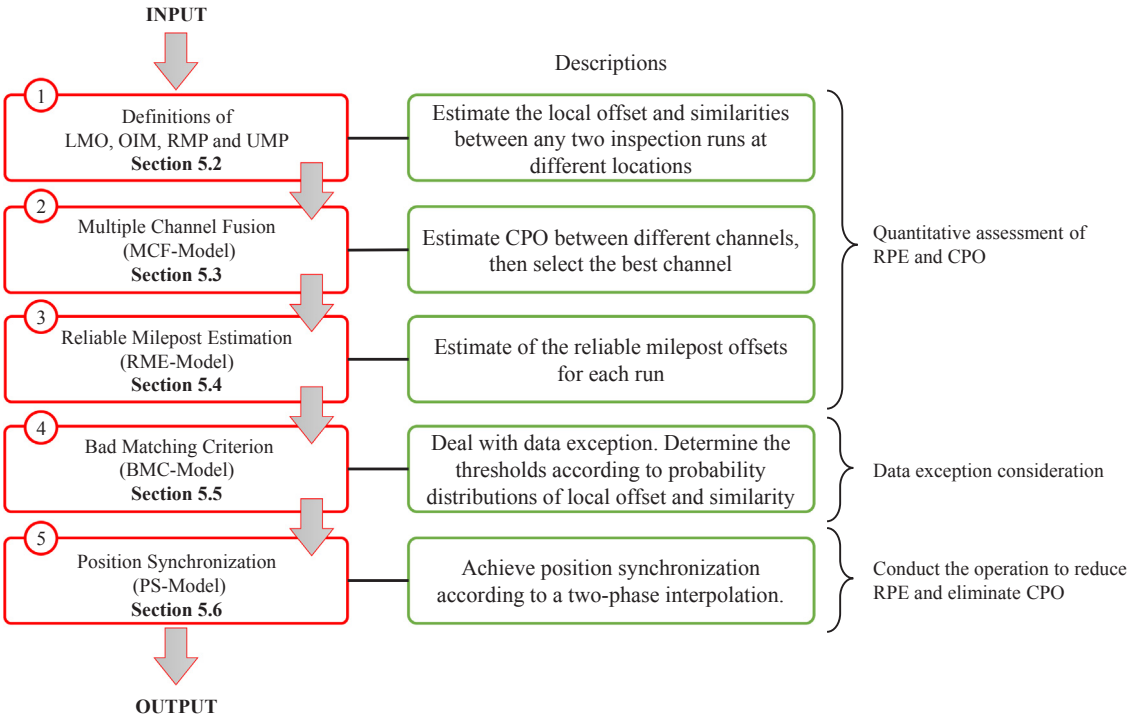

Fig. 5. Overall model framework.

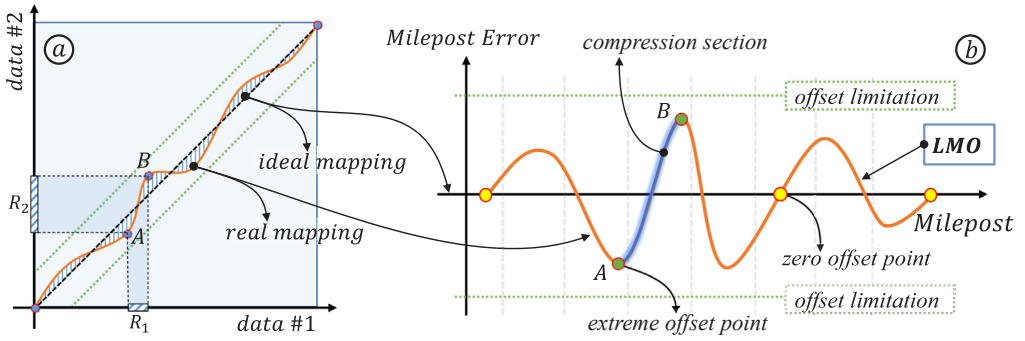

Fig. 6. LMO between data #1 and data #2.

$$\begin{cases} \delta^{(s,k)}(X, Y) \triangleq = \operatorname{argmax}_{-\Delta \leqslant \theta \leqslant \Delta} |\mathbf{cov}(X^{(s,k+\theta)}, Y^{(s,k)})| \\ \rho^{(s,k)}(X, Y) \triangleq = \mathbf{corr}(X^{(s,k+\delta^{(s,k)})}, Y^{(s,k)}) \end{cases} \tag{2}$$

where Δ is a limitation of possible offset range.

It can be interpreted that the local offset $\delta^{(s,k)}$ is defined as a local shift θ that causes $X^{(s,k+\theta)}$ to experience the strongest covariance with $Y^{(s,k)}$ within the range of $[-\Delta, \Delta]$. The similarity $\rho^{(s,k)}$ is defined as the normalized correlation coefficient of $X^{(s,k+\delta(s,k))}$ and $Y^{(s,k)}$.

Introducing a matching step $d_s (d_s \geqslant 1)$ to sample LMO discretely, the location parameter k is specified as a natural number within the range of $[1, N/d_s]$. It should be noted that $\delta^{(s,k)}(X, Y)$ may not equal $\delta^{(s,k)}(Y, X)$ when X and Y are unevenly stretched or compressed around the location k.

5.2.2. Overall information matrix

When considering a dataset of $n (n \geqslant 2)$ runs $\{X_1, X_2, \ldots, X_n\}$, with each containing t channels, all intermediate results, including local offset and waveform similarity, can be gathered into a high-dimension matrix $\boldsymbol{U} = \{(\delta_{i,j,c}^{(k)}, \rho_{i,j,c}^{(k)}) | i, j \in [1, n]; k \in [1, N/d_s]; c \in [1, t]\}$, which is named the Overall Information Matrix (OIM). The detailed structure of $\boldsymbol{U}$ can be found in Appendix A. The formation process of $\boldsymbol{U}$ is highly parallel since the elements in $\boldsymbol{U}$ are independent.

5.2.3. Reliable and unreliable matching point

In practice, for some reasons, such as abnormal data, maintenance carried out by heavy machinery or track condition degradation, a bad correlation will exist at some milepost points, namely $\rho^{(s,k)}(X, Y) \ll 1$. Therefore, the similarity threshold ρ_0 is introduced to judge whether a matching exception exists at location k. Another situation is also taken as a matching exception when the estimated local offset exceeds a given threshold δ_0. The determination of similarity threshold is based on the probability distribution, which is to be addressed in Section 5.5.4. As a result, the exception judgment operation $\mathbf{bo}(\delta, \rho)$ is defined as follows:

$$\mathbf{bo}(\delta, \rho) \triangleq = \begin{cases} 1, & \text{if } \rho \geqslant \rho_0 \text{ and } |\delta| \leqslant \delta_0 \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

If $\mathbf{bo}(\delta_{i,j}^{(k)}, \rho_{i,j}^{(k)}) = 1$, it indicates that the milepost point at location k is reliable according to the inspection data from the ith and jth runs; we call this kind of matching point a Reliable Matching Point (**RMP**). Otherwise it is an Unreliable Matching Point (**UMP**), and should be ignored. The ratio between the number of UMPs and the number of all elements in $\boldsymbol{U}$ is denoted as r_{UMP}. The ratio r_{UMP} represents the proportion of bad matching points. Moreover, the larger the ratio r_{UMP} is, the poorer the repeatability of track geometry is, so that the ratio r_{UMP} is an important index to describe the reliability of position synchronization.

$$r_{UMP} \triangleq = \frac{\mathbf{card}(\{\mathbf{bo}(\delta_{i,j}^{(k)}, \rho_{i,j}^{(k)}) = 0 | i, j \in [1, n]; k \in [1, N/d_s]; c \in [1, t]\})}{\mathbf{card}(\boldsymbol{U})} \tag{4}$$

5.3. Multiple Channel Fusion Model (MCF-Model)

The purpose of this section is to estimate the CPO of each inspection run and establish the model for fusing multiple inspection channels. All inspection channels provide information for position synchronization after the corresponding CPO is corrected. The fusion of multiple channels is the essential part of the models in this paper to deal with data exception.

The estimation of CPO according to the overall information matrix $\boldsymbol{U}$ is presented in Fig. 7. The cube on the left side of Fig. 7 represents the overall offset matrix $\boldsymbol{\Delta}_{c1}(\boldsymbol{X})$, please refer to ①. Subscript $c1$ indicates the channel of gauge. The right cube represents matrix $\boldsymbol{\Delta}_{ct}(\boldsymbol{X})$, see ②. Subscript ct indicates any other channel but that of gauge. The matrixes $\Delta_{c1,k}(\boldsymbol{X})$ and $\Delta_{ct,k}(\boldsymbol{X})$ are slices at location k of $\boldsymbol{\Delta}_{c1}(\boldsymbol{X})$ and $\boldsymbol{\Delta}_{ct}(\boldsymbol{X})$, respectively, please refer to ③ and ④. The two vectors $\delta_{i,j,c1}$ and $\delta_{i,j,ct}$ are extracted from $\boldsymbol{\Delta}_{c1}(\boldsymbol{X})$ and $\boldsymbol{\Delta}_{ct}(\boldsymbol{X})$, respectively, please refer to ⑤ and ⑥. In an ideal situation, $\delta_{i,j,c1}$ and $\delta_{i,j,ct}$ should be coincident, while in actuality there may exist a constant deviation. The difference between $\delta_{i,j,c1}$ and $\delta_{i,j,ct}$ is the CPO between the ith and jth run for channel ct, that is denoted as $C_{i,j,ct}.C_{i,j,ct}$, the constant offset of channel ct between the ith and jth run, can be obtained by solving Eq. (5).

$$\text{argmin}_{C_{i,j,ct}} = \|\boldsymbol{\delta}_{i,j,c1} - \boldsymbol{\delta}_{i,j,ct} - C_{i,j,c}\|^2;\ c = 2, 3, \ldots, t \tag{5}$$

The best estimation of the channel offset of the ith run, denoted as $D_{i,c}$, can be obtained by solving Eq. (6).

$$\text{argmin}_{D_{i,c}} = \|\boldsymbol{C}_{i,j,ct} - D_{i,c}\|^2;\ i = 1, 2, \ldots, n;\ c = 2, 3, \ldots, t \tag{6}$$

Finally, the final local mileage offset $\delta_{ij}^{(k)*}$ between the ith and jth runs at location k can be obtained by subtracting D_{i,c^*} from $\delta_{ij,c^*}^{(k)}$, and the final waveform similarity $\rho_{ij}^{(k)*}$ equals to the one estimated within the channel c^*, as presented in Eq. (7).

$$\begin{cases} \delta_{ij}^{(k)*} = \delta_{ij,c^*}^{(k)} - D_{i,c^*} \\ \rho_{ij}^{(k)*} = \rho_{ij,c^*}^{(k)} \end{cases} \tag{7}$$

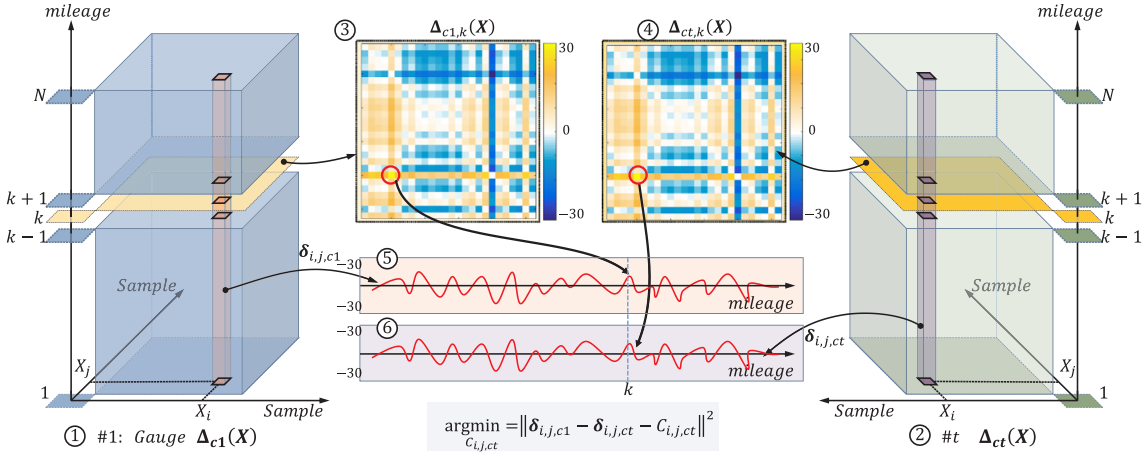

Fig. 7. Estimation of channel offset.

5.4. Reliable Milepost Estimation Model (RME-Model)

The purpose of this section is to establish a model to estimate the reliable milepost offsets for each run of inspection data. All data from other inspection runs are taken as a reference to determine the reliable milepost offset for each run. The reliable milepost estimation (RME) of the ith run at location k is denoted as $d_i^{(k)}$. The RME-Model relies on two principles:

- The RME minimizes the sum of squared differences between the ith and other inspection runs;
- The sum of RME of all runs at each location k equals zero.

Therefore, the RME-Model can be described as an optimization model given in Eq. (8).

$$\begin{cases} \min = \sum_{i=1}^{n} \sum_{j=1}^{n} (\delta_{ij}^{(k)*} - d_i^{(k)})^2 \\ \sum_{i=1}^{n} d_i^{(k)} = 0 \end{cases} \tag{8}$$

Eq. (8) contains an optimization objective and a constrain, which are corresponding to the two principles given above. Eq. (8) is a constrained least squares problem, which can be solved using the Augmented Lagrangian method, please refer to Appendix B.

5.5. Bad Matching Criterion Model (BMC-Model)

Bad matching points, such as the UMPs, lead to invalid estimations of local offsets and have a great influence on the position synchronization of inspection data. Especially in the event of a large erroneous estimation of local offset which is not rectified, the waveform will distort, the effects of which are all but irreversible. The purpose of this section is to minimize the probability for UMPs to be treated as RMPs through the establishment of the BMC-Model. In this paper, the BMC-Model includes three parts: (1) amplitude criterion, (2) similarity criterion, and (3) 1st derivative criterion. Multiple criteria can achieve better performance against the fault matching points.

5.5.1. Amplitude criterion

The amplitude criterion is used for the local offsets, and it works when the magnitude of a local offset is beyond the given threshold δ_0, a limitation not likely to be exceeded. It is especially useful when the inspection data waveform is strongly periodic. The amplitude criterion is expressed by Eq. (9).

$$|\delta_{i,j}^{(k)}| \leqslant \delta_0 \tag{9}$$

5.5.2. Similarity criterion

The similarity criterion is used for the local waveform similarity, and it works when the similarity of two waveforms is significantly less than 1. A low similarity indicates unreliable waveform matching. The similarity criterion is expressed by Eq. (10).

$$|\rho_{i,j}^{(k)}| \geqslant \rho_0 \tag{10}$$

5.5.3. 1st derivative criterion

Unlike the above two criteria, the 1st derivative criterion is used for the RME-Model. It works when the change rate of the RME along the railway is faster than a given threshold δ_1, a limitation not likely to be exceeded. The 1st derivative criterion is expressed by Eq. (11).

$$|\dot{d}_i^{(k)*}| \leqslant \delta_1 \tag{11}$$

where $\dot{d}_i^{(k)*}$ is the first order derivative of $d_i^{(k)*}$ along the railway, namely the change rate of the RME. $\dot{d}_i^{(k)*}$ is defined by Eq. (12). It should be noted that, when $\delta_1 < 1$, the 1st derivative criterion is able to guarantee the monotonicity of the newly generated position coordinates in Section 5.6.

$$\dot{d}_i^{(k)*} = \frac{\mathrm{d}}{\mathrm{d}k}(d_i^{(k)*}) \tag{12}$$

5.5.4. Determination of δ_0, ρ_0 and δ_1

The amplitude threshold δ_0 and similarity threshold ρ_0 are determined based on the joint probability distribution of δ and ρ. As the δ_0 becomes larger, it is more likely for a UMP to be mistakenly taken as an RMP. Generally, for a lager local milepost offset, the corresponding similarity will also be larger. In practice, the aim to reduce the probability of false matching can always be achieved by increasing the values of the thresholds δ_0 and ρ_0.

The joint probability distribution of the local milepost offset and similarity is illustrated in Fig. 8. The log-normal distribution of ρ

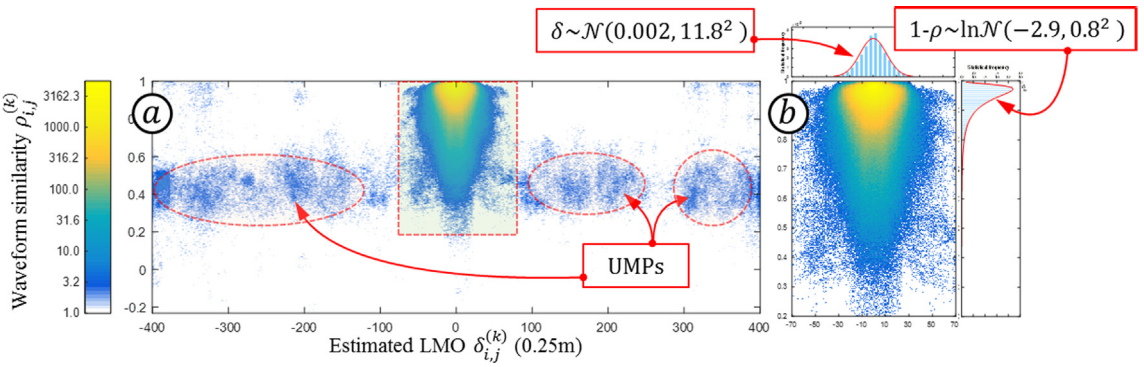

Fig. 8. The joint probability distribution of local milepost offset and waveform similarity. (b) Is an enlarged view of the range in the highlighted square of (a). The scattered points in the highlighted circles indicate the UMPs.

is given in $(1-\rho) \sim \ln\mathcal{N}(-2.9, 0.8^2)$. In this paper, the thresholds δ_0 and ρ_0 are given according to the 99% confidence level, or stricter 95% confidence. As for the threshold δ_1, the distribution of $\dot{d}_i^{(k)*}$ can also be obtained from the overall information matrix $\boldsymbol{U}$. Similarly, the value of δ_1 can be obtained according to the 99% confidence level, or stricter 95% confidence.

It should be noted that the confidence level can be taken as a percentage threshold, which is given empirically according to subjective experience. Higher confidence level leads to more misjudgments of Reliable Matching Points (RMPs) and fewer misjudgments of Unreliable Matching Points (UMPs), and vice versa. The percentage should not be too large or too small. In this paper, two percentages are suggested, 95% and 99%.

5.6. Position Synchronization Model (PS-Model)

The purpose of this section is to establish a model to conduct position synchronization according to the estimated RPE and CPO. Assuming $d_i^{(k)*}$ is the best estimated mileage offset of the ith run at location k, the position synchronization can be achieved by moving the data point from location k to a new location of $k + d_i^{(k)*}$. The channel offset $D_{i,c}$ for different runs needs to be included to ensure channel synchronization. The process can be described as a two-phase interpolation approach that can be written as Eq. (13).

$$X_{i,c}^{*} = \mathbf{interp}(\mathbf{interp}(d_m,\ d_m + d_i^{(k)*} + D_{i,c},\ d_v),\ X_{i,c},\ d_v);\ i = 1,\ 2,\ \ldots,n;\ c = 1,\ 2,\ \ldots,t \tag{13}$$

where $X_{i,c}^{*}$ is the synchronized data, and $d_v = (1, 2, \ldots, N)^T$. $d_m = \{(i-1)\cdot d_s + 1 | i = 1, 2, \ldots, N/ds\}$, is the matching position sequence with a step of d_s.

The first interpolation (the inside interpolation in Eq. (13) is aimed at generating a new position coordinate according to the mapping of $(d_{ori},\ d_{new})$. Since the new position coordinate does not share the same sampling rate with that of the original, the second interpolation (the outside interpolation in Eq. (13)) is conducted to achieve data resampling by d_v. Piecewise linear interpolation is adopted in the first interpolation, while both piecewise linear interpolation and cubic spline interpolation can be used in the second interpolation.

The PS-Model process is illustrated in Fig. 9. The solid line in Fig. 9(a) represents the estimated RME of one run of inspection data. The estimated $d_i^{(k)*}$ are taken as a discrete sampling of RME. The piecewise linear interpolation for the first interpolation of Eq. (13) is equivalent to a linear approximation of the RME. The second interpolation of Eq. (13) can reduce the RME, as shown in Fig. 9(b). The residual RME is the interpolation remainder of the first interpolation of Eq. (13).

For $x \in [x_0, x_1]$, the linear interpolation remainder is expressed as:

$$R(x) = \frac{f'(\xi)}{2}(x-x_0)(x-x_1) \tag{14}$$

It can be seen from Eq. (14) that, theoretically, the residual RME will converge towards 0 as $|x_1-x_0|$ approaches 0. In this paper, the value of $|x_1-x_0|$ equals the matching step d_s. However, the smaller $|x_1-x_0|$ is, the more calculations there will be. The optimization of computation efficiency and precision will be discussed further in Section 8.

6. Computational algorithms

The purpose of this section is to develop algorithms to achieve the estimating, fusing and synchronizing processes. A direct solution is proposed in this paper through an Overall-Iteration Algorithm (OI-Algorithm). As an improvement, the Incremental-Learning Algorithm (IL-Algorithm) is developed to handle the “lack of memory” and massive computation challenges.

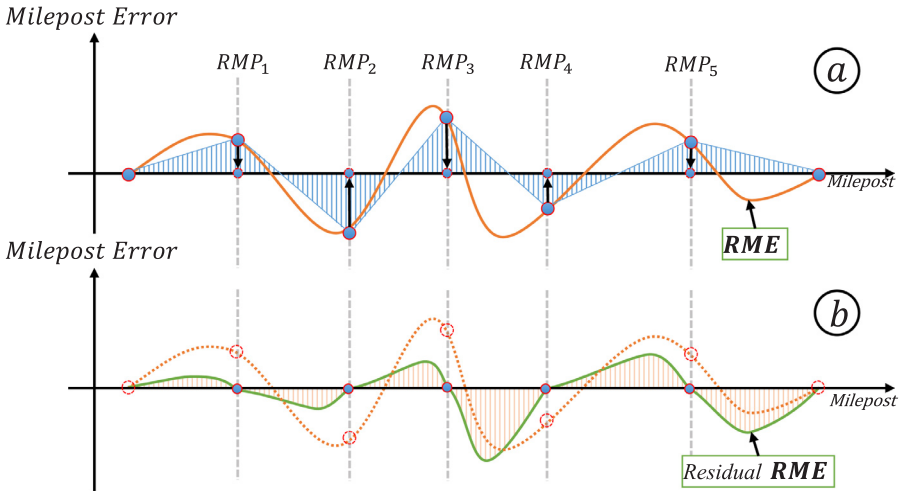

Fig. 9. Illustration of the PS-Model process.

6.1. OI-Algorithm

Considering that deviations may exist in the RME values due to distortions of the original waveform in the inspection data, instead of simply decreasing the matching step d_s, further iterations of the PS-Model on the newly processed results can more efficiently enhance the position synchronization performance. According to Eq. (14), the farther the position is to the interpolation points, the larger the residual RME becomes. A new iteration for position synchronization can be better when it avoids the previous RMPs. It is achieved in this paper by changing the beginning position of the matching sequence d_m according to the iteration number, as in Eq. (15).

$$d_m^* = \left\{ (i-1)\cdot d_s + (j-1)\cdot \frac{s}{N_o} + 1 \;\middle|\; i = 1, 2, ..., \frac{N}{ds};\ j = 1, 2, ..., N_o \right\} \tag{15}$$

where d_m^* represents the matching sequence with multiple iterations; N_o is the overall number of iterations; j is the iteration number.

As a result, the OI-Algorithm, which takes the simultaneous input of data from all inspection runs, contains several iterations and in each iteration the four models are solved in order. A new iteration is carried out on the results of the previous iteration (or the original data for the first iteration). The flow chart of the OI-Algorithm is presented in Appendix C.

The implement of OI-Algorithm is exactly corresponding to the data processing flow provided in Fig. 5, the overall model framework. From the calculation steps illustrated in Fig. 17, it can be found that the input dataset will undergo the five sub-procedures given in Sections 5.2-5.6, one after the other until the final result is obtained.

Though the OI-Algorithm is a complete implementation of the four models presented in Section 5, there are two main challenges for practical application of the algorithm.

- ***Challenge 1*: lack of memory**. Take a 300 km track section for example, the required buffer memory reaches 12 GB to cache the data of 50 inspection runs. More memory is needed to carry out the OI-Algorithm. The size of the overall information matrix $\boldsymbol{U}$ is $50 \times 50 \times 10{,}000 \times t$ for a matching step of 30 m, where t is the number of inspection channels in use. $\boldsymbol{U}$ is larger when processing data from more inspection runs.
- ***Challenge 2*: massive computations**. Each time a new inspection run is conducted, the OI-Algorithm needs to be re-executed with massive repeated calculations. A direct countermeasure can be helpful to reduce repeated computation, such as storing the matrix $\boldsymbol{U}$. Nevertheless, with an increasing number of inspection runs, it is uneconomical to store such a quantity of intermediary data.

An incremental algorithm is of great importance to achieve the models instead of the OI-Algorithm. The aim is not simply to transform the OI-Algorithm into an incremental style, but to minimize the required memory, and optimize computations through incremental probability estimation and the establishment of a “Knowledge Library”.

6.2. IL-Algorithm to improve the OI-Algorithm

6.2.1. Knowledge library

In fact, it is not necessary to store all elements in the overall information matrix $\boldsymbol{U}$. The concept of a “knowledge library” represents the minimal information refined from the high-dimensional matrix $\boldsymbol{U}$. The refining process compresses $\boldsymbol{U}$ into several low-dimensional matrices or vectors through statistical approaches. Each time a new set of data is obtained from the track inspection car,

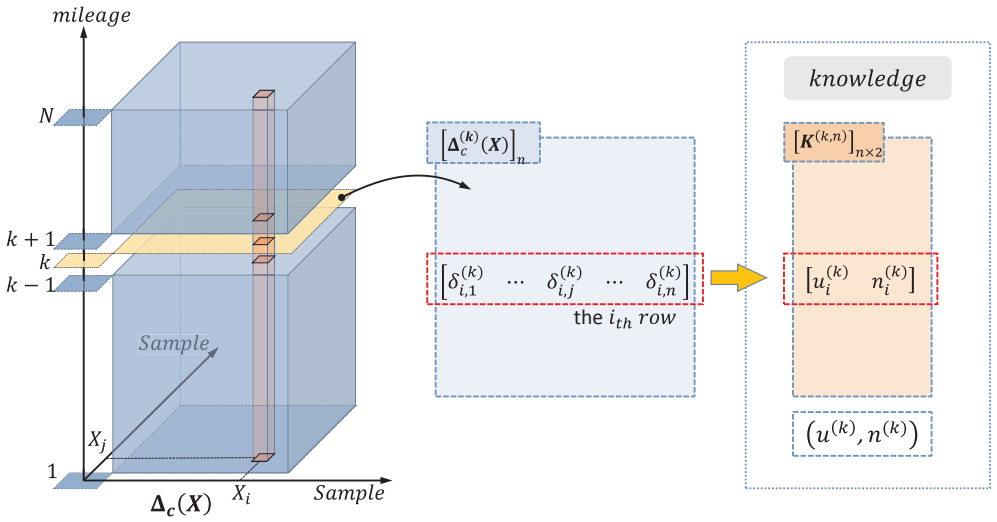

Fig. 10. The refining processing for local knowledge.

the RPE and CPO of the newly measured data is estimated by reference to the historical knowledge. In return, the knowledge will be updated.

The knowledge library contains two parts: local knowledge and global knowledge. Local knowledge is the refined information from the overall information matrix $\boldsymbol{U}$ for different inspection runs at different locations; this process is shown in Fig. 10.

$\boldsymbol{\Delta}_c^{(k)}(\boldsymbol{X})$ represents the slice of $\Delta_c(\boldsymbol{X})$ at location k. It should be noted that the UMPs should not be included, so the values $\delta_{ij}^{(k)}$ are excluded when $\mathbf{bo}(\delta_{ij}^{(k)}, \rho_{ij}^{(k)}) = 0$. The average and the number of elements in the ith row of $\boldsymbol{\Delta}_c^{(k)}(\boldsymbol{X})$, denoted as $u_i^{(k)}$ and $n_i^{(k)}$ respectively, are taken as local knowledge, as shown in Eq. (16).

$$\begin{cases} u_i^{(k)} = \frac{1}{n_i^{(k)}} \sum_{j=1}^{n} \mathbf{bo}(\delta_{ij}^{(k)*}, \rho_{ij}^{(k)*}) \cdot \delta_{ij}^{(k)*} \\ n_i^{(k)} = \sum_{j=1}^{n} \mathbf{bo}(\delta_{ij}^{(k)*}, \rho_{ij}^{(k)*}) \end{cases} \tag{16}$$

The value of $\mathbf{bo}(\delta_{ij}^{(k)*}, \rho_{ij}^{(k)*})$ equals 1 for a RMP, and 0 for a UMP, so $n_i^{(k)}$ is calculated by summing up $\mathbf{bo}(\delta_{ij}^{(k)*}, \rho_{ij}^{(k)*})$ directly for $j = 1, 2, \ldots, n$.

Global knowledge retains the overall features of the matrix $\boldsymbol{U}$, including the statistical distributions of δ, ρ and $\dot{d}$, that are denoted as $f_\delta[p]$, $f_\rho[p]$ and $f_{\dot{d}}[p]$, respectively. The range $L_r (r = \delta, \rho, \dot{d})$ can be divided into $2N'$ parts. The number of parameters in each part p are counted according to Eq. (17).

$$N_{r,p} = \mathbf{card}\left(\left\{r \;\middle|\; \frac{p}{N'} L_{r,1} \leqslant r < \frac{p+1}{N'} L_{r,2}\right\}\right); \; p = -N', 1-N', \ldots, N'-1 \tag{17}$$

where the boundaries of δ, ρ, and $\dot{d}$ are given as:

$$\begin{cases} L_{\delta,(1,2)} = [-\Delta, \Delta] \\ L_{\rho,(1,2)} = [-1, 1] \\ L_{\dot{d},(1,2)} = [-\Delta_1, \Delta_1] \end{cases} \tag{18}$$

The total number of elements in $\boldsymbol{U}$ is $N_r = \mathbf{card}(\{r | L_{r,1} \leqslant r < L_{r,2}\})$. Global knowledge $f_r[p]$ can be calculated according to Eq. (19).

$$f_r[p] = \frac{N_{r,p}}{N_r}; \tag{19}$$

6.2.2. Incremental implementation of the RME $d_i^{(k)*}$

According to Appendix B, the result $d_i^{(k)*}$ consists of two parts, $d_i'^{(k)}$ and $d^{(k)*}$, the incremental implementation of $d_i^{(k)*}$ can therefore be derived from two parts, as expressed in Eq. (20).

$$d_i^{(k)*} = d_i'^{(k)} - d^{(k)*} = u_i^{(k,n+1)} - \frac{1}{n+1} \sum_{i=1}^{n+1} u_i^{(k,n+1)} \tag{20}$$

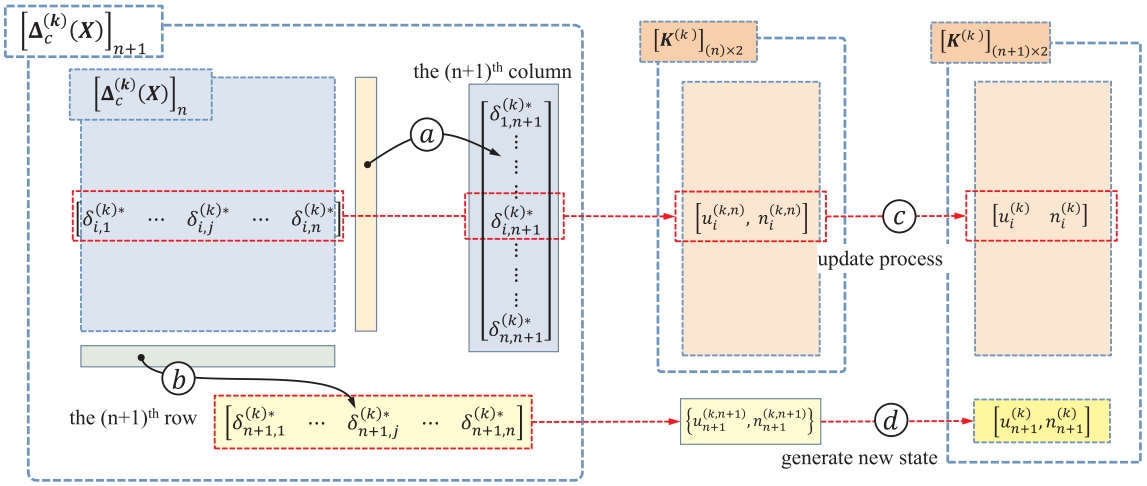

Fig. 11. Local knowledge updating process.

It should be noted that the $d_i^{(k)*}$ in Eq. (20) is the total estimation of reliable mileage offset based on the historical inspection data. When the data of the $(n+1)$th inspection run is included, the adjustment value is given in Eq. (21).

$$d_i^{(k,n+1)*} = d_i^{(k)*} - d_i^{(k,n)*} = (u_i^{(k,n+1)} - u_i^{(k,n)}) - \frac{1}{n+1}\left(\sum_{i=1}^{n+1} u_i^{(k,n+1)} - u_i^{(k,n)}\right) \tag{21}$$

where $u_{n+1}^{(k,n)}$ is defined as zero because there is no previous information for the $(n+1)$th run.

6.2.3. Local knowledge updating

The updating process of local knowledge contains four parts, as presented in Fig. 11.

Process A. Obtain the local milepost offset $\{(\delta_{i,n+1}^{(k)*}, \rho_{i,n+1}^{(k)*}) | i = 1, 2, ..., N\}$ by performing the waveform matching for data of the (n + 1)th run to each of the previous n runs, respectively, using the model presented in Section 5.2.1.
Process B. Obtain the local milepost offset $\{(\delta_{n+1,i}^{(k)*}, \rho_{n+1,i}^{(k)*}) | i = 1, 2, ..., N\}$ by performing the waveform matching for each of the previous n to $(n+1)$th runs, respectively, using the model presented in Section 5.2.1.

It should be noted that the aforementioned data from the previous n runs are the latest version after mileage synchronization.

Process C. The matrix $\boldsymbol{K}^{(k,n)}$ is updated into $\boldsymbol{K}^{(k,n+1)}$ with $\{(\delta_{i,n+1}^{(k)*}, \rho_{i,n+1}^{(k)*}) | i = 1, 2, ..., N\}$. The formula is as follows:

$$\begin{cases} u_i^{(k,n+1)} = \frac{1}{n_i^{(k,n+1)}}(n_i^{(k,n)} \cdot u_i^{(k,n)} + \mathbf{bo}(\delta_{i,n+1}^{(k)}, \rho_{i,n+1}^{(k)}) \cdot \delta_{i,n+1}^{(k)}) \\ n_i^{(k,n+1)} = n_i^{(k,n)} + bo(\delta_{i,n+1}^{(k)}, \rho_{i,n+1}^{(k)}) \end{cases} \tag{22}$$

Eq. (22) is an incremental form of Eq. (16). The mean value $u_i^{(k,n+1)}$ is updated based on the previous value $u_i^{(k,n)}$ and the newly input $\delta_{i,n+1}^{(k)}$.

Process D. The (n + 1)th row of $\boldsymbol{K}^{(k,n+1)}$ is generated with $\{(\delta_{n+1,i}^{(k)*}, \rho_{n+1,i}^{(k)*}) | i = 1, 2, ..., N\}$. The formula is as follows:

$$\begin{cases} u_{n+1}^{(k,n+1)} = \frac{1}{n_{n+1}^{(k,n+1)}} \sum_{j=1}^{n} \mathbf{bo}(\delta_{n+1,j}^{(k)*}, \rho_{n+1,j}^{(k)*}) \cdot \delta_{ij}^{(k)} \\ n_{n+1}^{(k,n+1)} = \sum_{j=1}^{n} \mathbf{bo}(\delta_{n+1,j}^{(k)*}, \rho_{n+1,j}^{(k)*}) \end{cases} \tag{23}$$

Eq. (23) is similar to Eq. (16), because there is no previous knowledge for the (n + 1)th inspection dataset.

6.2.4. Global knowledge updating

The global knowledge updating process can be achieved as Eq. (24).

$$f_r^{(n+1)}[p] = \frac{1}{N_r^{(n+1)}} \cdot (f_r^{(n)}[p] \cdot N_r^{(n)} + N_{r,p}^{(n+1)}); r = \delta, \rho, \dot{d} \tag{24}$$

where

$$N_{r,p}^{(n+1)} = \mathbf{card}\left(\left\{r_{ij} | \frac{p}{N'} L_{r,1} \leqslant r_{ij} < \frac{p+1}{N'} L_{r,2}; iorj = n+1\right\}\right) \tag{25}$$

$$N_r^{(n+1)} = N_r^{(n)} + \mathbf{card}(\{r_{ij} | L_{r,1} \leqslant r_{ij} < L_{r,2}; iorj = n+1\}) \tag{26}$$

6.2.5. IL-Algorithm

The flow chart of the IL-Algorithm is presented in Fig. 18 in Appendix D. The initialization of the knowledge library can be achieved by applying the OI-Algorithm on M inspection runs. After the data of the (n + 1)th inspection run is processed ($n \geqslant M$), the data of the ($n + 1 - M$)th inspection run is saved to the disk and released from memory, so there are only $M + 1$ inspection runs of data in memory. Each time data from a new inspection run is imported, the four models are to be solved using the newly processed historical data, rather than the original measured data. It should be noted that in this paper the data arrival order is arranged according to the inspection date. The early inspection data is processed before the late one.

There are two main improvements in the IL-Algorithm compared to the OI-Algorithm:

- The IL-Algorithm only needs to cache data from M inspection runs, rather than all historical inspection data as in the OI-Algorithm. The parameter M is determined according to the required precision and hardware environment, such as memory capacity.
- Instead of caching the entire matrix $\boldsymbol{U}$, only the knowledge library needs to be cached in memory. The knowledge library is updated incrementally with few redundant calculations during the execution process of the algorithm.

7. China high speed railway case study

The purpose of this section is to demonstrate the performance and contribution of the proposed track inspection data position synchronization methodology. It should be noted that there are a lot of results to be presented corresponding to the models given in Section 5. However, this section only presents the most important and interesting points due to limited space. This section focuses on four aspects, that are (1) multi-channel information fusion, (2) robustness against data exception, (3) precision and number of iterations, and (4) computational time. The precision of position synchronization is defined as the percentile value of the estimated LMO ($\delta_{i,j}^{(k)}$) at a given confidence level p, as given in Section 5.5.4.

A case study was carried out based on a dataset with 58 inspection runs (17.7 GB) between February/2014 and July/2016 on the China High Speed Railway. Each inspection run contained 13 measurement channels, including milepost, gauge, crosslevel, longitudinal profile (left and right side), alignment (left and right side), superelevation, curvature, carbody acceleration (vertical and lateral direction), speed and ALD (please refer to Xu et al., 2013 for a description of ALD). The length of the railway line was 323 km. The sampling distance of the inspection car was 0.25 m. Some default parameters of the IL-Algorithm are: matching scale $s = 50$ m; matching step ds = 20 m; matching range $\Delta = 40$ m and percentage threshold for BMC-Model is 99%. The dataset has undergone preliminary processing using the Key Equipment Identification (KEI) model proposed in Xu et al. (2013).

7.1. Multi-channel information fusion

Fig. 12 illustrates the estimated CPO of the original and processed inspection data. It can be found that the CPO of the original channel offset is within the range of [−1.25, 1.25] meters by reference to the gauge channel, as illustrated in Fig. 12(a). There are differences between different runs and channels. As for the processed data, the CPO drops to less than 0.025 m, which can nevertheless be disregarded, as found in Fig. 12(b).

7.2. Robustness against abnormal data

This section presents a comparison between the waveform before and after position synchronization. The inspection data from two channels, Channel #1 for track gauge and Channel #2 for crosslevel, within the range of K641.6-K641.8 are taken as an example, please refer to Fig. 13.

It can be seen that Channel #1 (gauge) has some abnormal data points, which randomly occurred in different inspection runs, as highlighted in Fig. 14(a), (b). And there are data exceptions in the measured gauge of the inspection run colored red, see Fig. 13(a), (b). By contrast, Channel #2 (crosslevel) does not have any abnormal data points within the same range of positions. As a result, the UMPs in Channel #1 are substituted by the RMPs in Channel #2 through multi-channel fusion. That explains the robustness of the methods in this paper against abnormal data for the synchronized gauge.

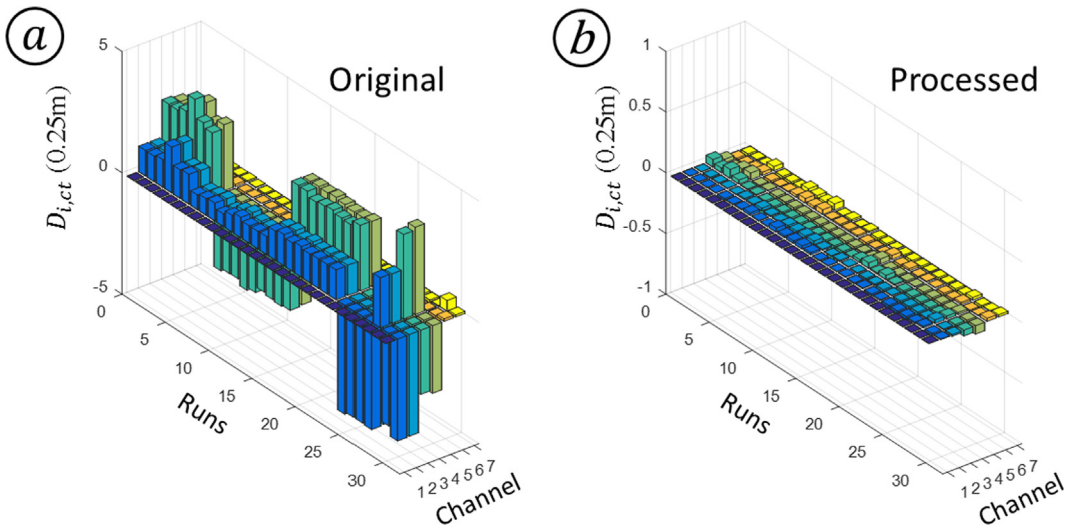

Fig. 12. The estimation of CPO; (a) shows the CPO of original data; (b) illustrates the CPO of the processed data.

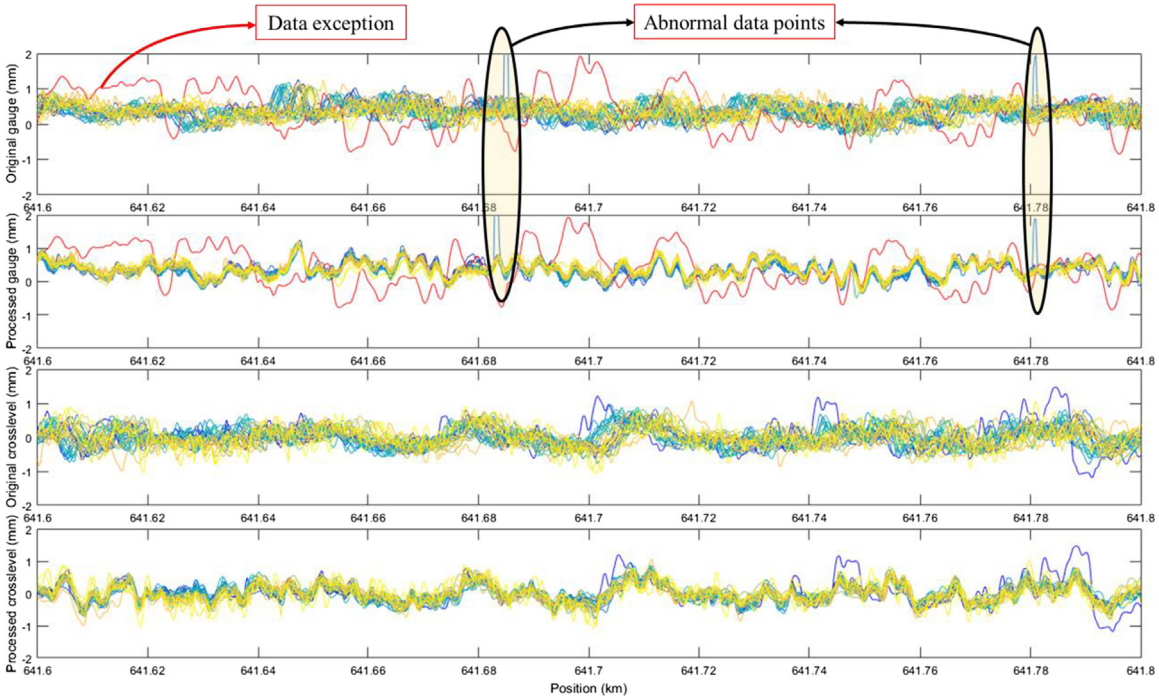

Fig. 13. Comparison of the waveform before and after position synchronization (Channel #1 gauge). (a and b) illustrate the waveforms of the original and processed gauge, respectively; (c and d) illustrate the waveforms of the original and processed crosslevel, respectively.

7.3. Precision and number of iterations

The purpose of this section is to analyze the influence of iterations on the precision of position synchronization. Fig. 14 illustrates the joint distribution of local milepost offset and waveform similarity after different iterations. It can be observed that the standard deviation of local milepost offset drops quickly to 0.73 (0.19 m) after the first iteration, then 0.26 (0.25 m) and 0.22 (0.25 m) after the second and third iteration. The distribution of waveform similarity, however, changes slightly with different iterations. The quantitative relationship among the iterations, distributions and thresholds are summarized in Table 4. It can be observed that after the first iteration, the threshold of δ is reduced from 6.58 m to 0.42 m with 99% confidence, or 0.3 m for 95% confidence. And after the

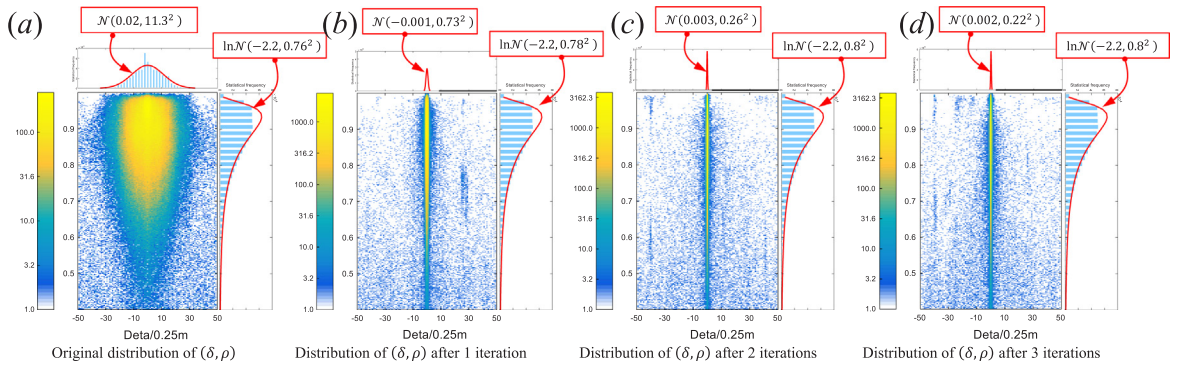

Fig. 14. The joint distribution of value pairs (δ, ρ). (a–d) show the distributions of the original data, and the data after one, two and three iterations respectively.

Table 4
Iterations, distribution and the precision $P_r(p)$.

Iterations	Distribution of the estimated local offset (0.25 m)	Precision $P_r(p)$ (m)	
		p = 99%	p = 95%
Original	$\mathcal{N}(0.020, 11.3^2)$	6.58	4.65
After 1 iteration	$\mathcal{N}(-0.001, 0.73^2)$	0.42	0.30
After 2 iterations	$\mathcal{N}(0.003, 0.26^2)$	0.15	0.10
After 3 iterations	$\mathcal{N}(0.002, 0.22^2)$	0.13	0.09

second iteration, the threshold of δ drops to 0.15 m and 0.1 m for 99% and 95% confidences, respectively. Here, the value of 0.15 m is less than the sampling distance of the inspection car, which is 0.25 m in China. However, the third iteration does not seem to show any significant improvement, so it is suggested that two iterations are necessary for position synchronization.

7.4. Computational time

This section analyzes the execution efficiency of OI- and IL-Algorithms. The algorithms are implemented with the MATLAB R2016b platform, in a CPU/I7-5820 k (6CPUs × 3.3 GHz) and RAM/16 GB hardware environment. The percentage threshold for the BMC-Model is 95% and the iteration was performed twice. The polynomial fitting of the execution time for both algorithms is performed, quadratic fitting for OI-Algorithm and linear fitting for IL-Algorithm. The results are presented in Fig. 15. It can be observed that the execution time for the OI-Algorithm (developed in Section 6.1 is $0.03n^2 + 0.05n - 0.03$ sec/km, where n represents the number of inspection times. In contrast, the execution time for the IL-Algorithm is $0.1n + 0.43$ sec/km.

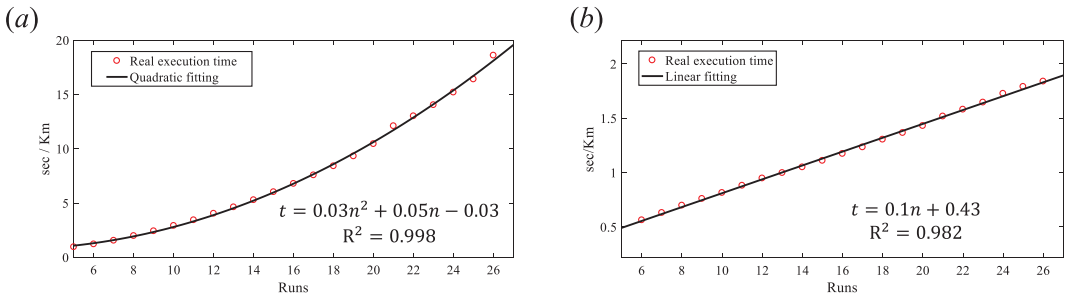

Fig. 15. The execution time per kilometer for OI-Algorithm (a) and IL-Algorithm (b).

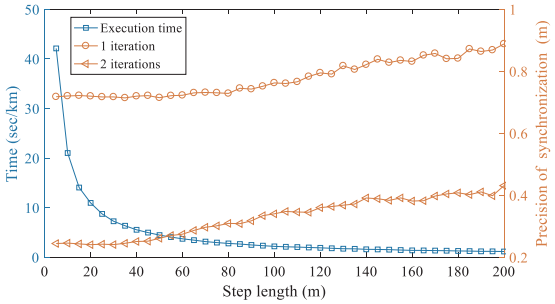

Fig. 16. The execution time and precision $P_r(p = 99.7\%)$ of position synchronization considering different matching step *ds*.

8. Discussion

The purpose of this section is to discuss the computation efficiency and precision of position synchronization. According to Eq. (14), the interpolation remainder is closely related to the distance between two adjacent interpolation points, which is determined by the matching step (*ds*), a parameter introduced in Section 5.2.1. The smaller the matching step *ds* is, the higher the interpolation precision that can be achieved, but the more execution time the IL-Algorithm will require. As a result, execution time and precision are two conflicting objectives. There are two essential parameters, the matching step *ds* and the number of total runs of inspection data cached in memory (the parameter M in the IL-Algorithm), which control the execution time and precision of position synchronization.

To optimize the execution time and precision, the IL-Algorithm is executed with different values of *ds*, from 5 m to 200 m. The results are illustrated in Fig. 16. There is an inversely proportional relationship between the execution time and matching step *ds*. When the matching step *ds* is less than 40 m, the execution time is seen to increase dramatically. The precision P_r (confidential level $p = 99.7\%$) of position synchronization is illustrated in Fig. 16. Fig. 16 shows that the precision of two iterations is much better than that of only one iteration, with an improvement of approximately 0.45 m on average. As the matching step becomes larger, the precision gradually drops. However, when the matching step is less than 40 m, the precision shows no improvement with smaller values. For a given requirement of precision according to the sampling distance of inspection cars, namely 0.25 m, the minimal matching step should be 50 m.

9. Conclusion

Position synchronization of track inspection data is crucial for track degradation modeling and maintenance scheduling. This paper develops a novel data-driven methodological framework for addressing the accurate and computationally efficient position synchronization of track infrastructure inspection data via big data fusion and an incremental learning algorithm. A number of data analysis models are established to mine, fuse and synchronize the inspection data of multiple runs with multiple sensors. The model also accounts for possible measurement data exceptions. An incremental learning algorithm (IL-Algorithm) is developed to facilitate the fast computation of enormously large inspection data volumes.

The proposed method has been applied to a section of track on the China High Speed Railway network. The results show that (1) our proposed method is robust against measurement data exceptions via multi-sensor data fusion; (2) the proposed algorithm can sufficiently synchronize positions within a small number of iterations (in some cases, only two iterations are sufficient); (3) because of these methodological merits, our approach can reduce the relative position error (RPE) within 0.15 meters at a 99 percent confidence level. This is a significant improvement in positioning accuracy, considering the fact that the RPE error is much smaller than the sampling interval (0.25 meters in our dataset); (4) for processing one additional kilometer of track, the proposed algorithm would take an extra $0.1n + 0.43$ s, where n is the number of inspection runs.

Finally, we discuss the relationship between the computational efficiency and the accuracy of position synchronization, and the proper number of inspection runs that should be cached in memory in the IL-algorithm. In summary, the proposed position synchronization methodology considers both positioning accuracy and computational time, with a promising application to data-rich railroad inspection information processing problems, in support of a wide array of track safety and maintenance research activities.

Acknowledgements

The present work has been supported by the National Natural Science Foundation of China (51425804, 51,608,459 and 51378439) and the United Fund for Key Projects of China's High-Speed Railway (U1334203 and U1234201). Yuan Wang is funded by China Scholarship Council CSC (No. 201707000036). Dr. Xiang Liu is funded by the US Federal Railroad Administration at the time of writing this paper. However, the authors are solely responsible for all views and analyses in this research.

Appendix A. The structure of OIM

$\delta_{i,j,c}^{(k)}$ and $\rho_{i,j,c}^{(k)}$ represent the local offset and similarity of the jth run to the ith run at location k based on the data of the cth channel, as shown in Eq. (28). $X_{i,c}$ represents the data of the cth channel of X_i.

$$\delta_{i,j,c}^{(k)} = \delta^{(s,(k-1)\cdot N/d_s)}(X_{i,c}, X_{j,c}) \tag{28a}$$

$$\rho_{i,j,c}^{(k)} = \rho^{(s,(k-1)\cdot N/d_s)}(X_{i,c}, X_{j,c}) \tag{28b}$$

Slicing up the matrix $\boldsymbol{U}$ at a given location k and channel c, we get the local offset matrix $\boldsymbol{\Delta}_c^{(k)}(\boldsymbol{X})$ and local similarity matrix $\Upsilon_c^{(k)}(\boldsymbol{X})$:

$$\Delta_c^{(k)}(\boldsymbol{X}) = \begin{bmatrix} \delta_{1,1,c}^{(k)} & \cdots & \delta_{1,n,c}^{(k)} \\ \vdots & \ddots & \vdots \\ \delta_{n,1,c}^{(k)} & \cdots & \delta_{n,n,c}^{(k)} \end{bmatrix}; \boldsymbol{\Upsilon}_c^{(k)}(\boldsymbol{X}) = \begin{bmatrix} \rho_{1,1,c}^{(k)} & \cdots & \rho_{1,n,c}^{(k)} \\ \vdots & \ddots & \vdots \\ \rho_{n,1,c}^{(k)} & \cdots & \rho_{n,n,c}^{(k)} \end{bmatrix} \tag{29}$$

Then the overall offset matrix $\boldsymbol{\Delta}_c(\boldsymbol{X})$ and the overall similarity matrix $\boldsymbol{\Upsilon}_c(\boldsymbol{X})$ is denoted by Eq. (30). The Overall Information Matrix $\boldsymbol{U}$ can be rewritten as Eq. (31).

$$\boldsymbol{\Delta}_c(\boldsymbol{X}) = [\boldsymbol{\Delta}_c^{(1)}(\boldsymbol{X}),\ \boldsymbol{\Delta}_c^{(2)}(\boldsymbol{X}),\ \cdots\ \boldsymbol{\Delta}_c^{(N/ds)}(\boldsymbol{X})] \tag{30a}$$

$$\boldsymbol{\Upsilon}_c(\boldsymbol{X}) = [\boldsymbol{\Upsilon}_c^{(1)}(\boldsymbol{X}),\ \boldsymbol{\Upsilon}_c^{(2)}(\boldsymbol{X})\ \cdots\ \boldsymbol{\Upsilon}_c^{(N/ds)}(\boldsymbol{X})] \tag{30b}$$

$$\boldsymbol{U} = \{(\boldsymbol{\Delta}_c(\boldsymbol{X}), \boldsymbol{\Upsilon}_c(\boldsymbol{X}))|c = 1, 2, ..., t\} \tag{31}$$

Appendix B. Solution of RME-Model using the Augmented Lagrangian method

Eq. (8) is a least squares problem with a constraint, which can be solved through the Augmented Lagrangian method. Introduce variable λ, and Eq. (8) can be transferred into Eq. (32).

$$\min = \sum_{i=1}^{n}\sum_{j=1}^{n}(\delta_{ij}^{(k)*} - d_i^{(k)})^2 + \lambda\sum_{i=1}^{n} d_i^{(k)} \tag{32}$$

Through conducting partial derivative operations for each variable and solving the resultant equations, we can obtain the best estimation of $d_i^{(k)}$, as presented in Eq. (33).

$$d_i^{(k)*} = \frac{1}{n}\sum_{j=1}^{n}\delta_{ij}^{(k)*} - \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\delta_{ij}^{(k)*} \quad (i = 1, 2, ..., n) \tag{33}$$

The $d_i^{(k)*}$ can be divided into two components, $d_i'^{(k)}$ and $d^{(k)*}$.

$$d_i^{(k)*} = d_i'^{(k)} - d^{(k)*} \tag{34}$$

where,

$$d_i'^{(k)} = \frac{1}{n}\sum_{j=1}^{n}\delta_{ij}^{(k)*}; d^{(k)*} = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\delta_{ij}^{(k)*} \tag{35}$$

The first component $d_i'^{(k)}$ represents the average of the local offsets between the data from other inspection runs and the ith run at location k. The second component $d^{(k)*}$ represents the average of all the local offsets between inspection data from every two runs at location k, namely the average of $\Delta_{c*}^{(k)}(\boldsymbol{X})$. The magnitude of $d^{(k)*}$ reflects the degree of waveform irregularity resulting from rarefaction and compression, in which the smaller value is better. When $\Delta_{c*}^{(k)}(\boldsymbol{X})$ is an antisymmetric matrix, satisfying Eq. (36), we get $d^{(k)*} = 0$ and $d_i^{(k)*} = d_i'^{(k)}$.

$$\delta_{ij}^{(k)*} = -\delta_{ji}^{(k)*} \tag{36}$$

Appendix C. The OI-Algorithm Flow Chart

See Fig. 17.

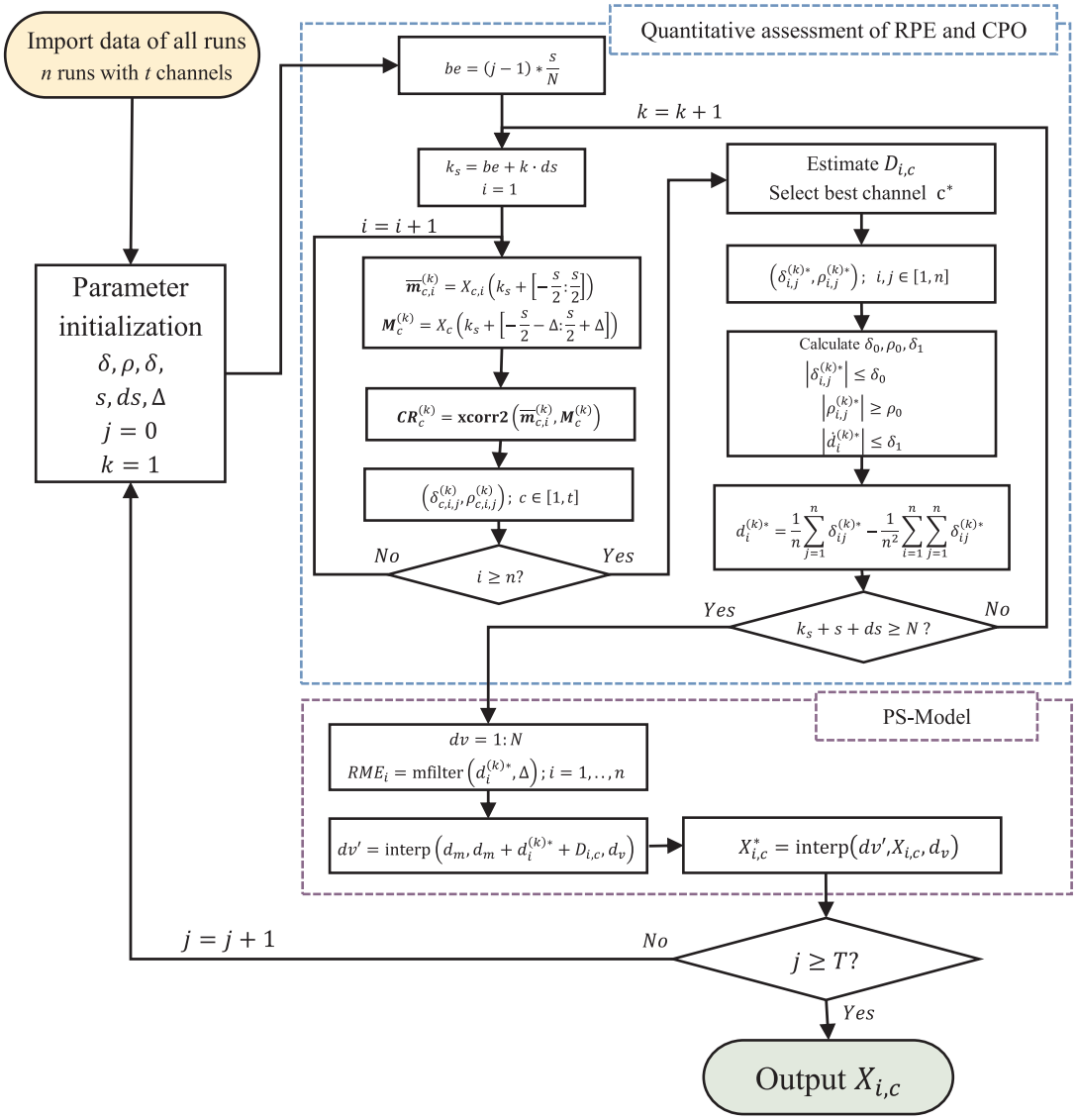

Fig. 17. The OI-Algorithm flow chart.

Appendix D. The IL-Algorithm Flow Chart

See Fig. 18.

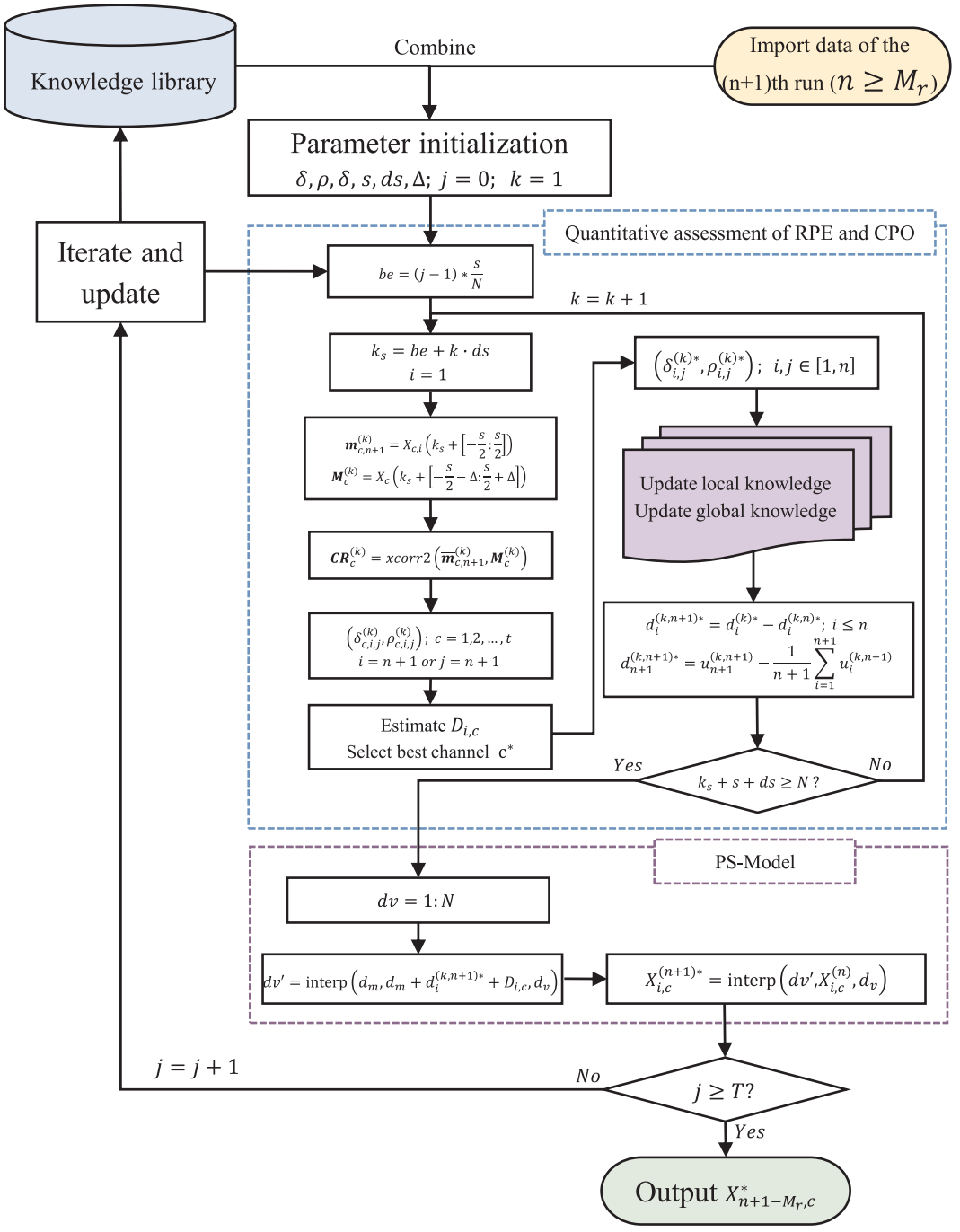

Fig. 18. The IL-Algorithm Flow Chart.

References

Alfelor, R.M., Carr, G.A., Fateh, M., 2001. Track degradation assessment using gauge restraint measurements. Transp. Res. Rec. 1, 68–77.

Allotta, B., Colla, V., Malvezzi, M., 2002. Train position and speed estimation using wheel velocity measurements. Proc. Inst. Mech. Eng. Part F J. Rail Rapid Transit 216 (3), 207–225.

Bartram, D., Burrow, M., Yao, X., 2008. A Computational Intelligence Approach to Railway Track Intervention Planning. Evolutionary Computation in Practice. Springer, Berlin Heidelberg.

Bocciolone, M., Caprioli, A., Cigada, A., Collina, A., 2007. A measurement system for quick rail inspection and effective track maintenance strategy. Mech. Syst. Sig. Process. 21 (3), 1242–1254.

Esveld, C., 2001. Modern Railway Track. Delft University of Technology Publishing Service, The Netherlands, pp. 349–406.

Haigermoser, A., Luber, B., Rauh, J., Gräfe, G., 2015. Road and track irregularities: measurement, assessment and simulation. Veh. Syst. Dyn. 53 (7), 878–957.

Hanreich, D.W., Mittermayr, P., Presle, G., 2002. Track geometry measurement database and calculation of equivalent conicities of the OBB network. In: Proceedings of the AREMA 2002 Annual Conference, Washington, DC.

Higgins, C., Liu, X., 2017. Modeling of track geometry degradation and decisions on safety and maintenance: a literature review and possible future research directions. Proc. Inst. Mech. Eng. Part F J. Rail Rapid Transit 095440971772187.

Kawaguchi, A., Miwa, M., Terada, K., 2005. Actual data analysis of alignment irregularity growth and its prediction model. Quart. Report of Rtri 46 (4), 262–268.

Li, H., Xu, Y., 2010. A method to correct the mileage error in railway track geometry data and its usage. In: 2010 International Conference on Traffic and Transportation Studies. pp. 1130–1135.

Liu, R., Xu, P., Wang, F., 2010. Research on a short-range prediction model for track irregularity over small track lengths. J. Transp. Eng. 136 (12), 1085–1091.

Liu, B., Bruni, S., 2015. Analysis of wheel-roller contact and comparison with the wheel-rail case. Urban Rail Transit 1 (4), 215–226.

Liu, X., Saat, M.R., Qin, X., Barkan, C.P., 2013. Analysis of U.S. freight-train derailment severity using zero-truncated negative binomial regression and quantile regression. Accid. Anal. Prev. 59 (59C), 87–93.

Pedanekar, N.R., 2006. Methods for aligning measured data taken from specific rail track sections of a railroad with the correct geographic location of the sections, US, US7130753[P].

Qu, J., 2012. Study on Track Irregularity Prediction and Decision-aiding Technology based on TQI of Raised Speed Lines. Ph.D. thesis. Beijing Jiaotong University, Beijing, China.

Yang, A., 2009. Automatic correction milepost system of geometry inspection car based on RFID. Railway Comput. Appl. 18 (10), 39–41 (In Chinese).

Quiroga, L.M., Schnieder, E., 2012. Monte Carlo simulation of railway track geometry deterioration and restoration. Proc. Inst. Mech. Eng. Part O J. Risk Reliab. 226 (3), 274–282.

Ren, S., Gu, S., Xu, G., Gao, Z., Feng, Q., 2010. Development of GJ-4G track inspection car. Proc. Spie 7544 75440Q–75440Q-6.

Sadeghi, J., 2010. Development of railway track geometry indexes based on statistical distribution of geometry data. J. Transp. Eng. 136 (8), 693–700.

Sadeghi, J., Askarinejad, H., 2011. Development of track condition assessment model based on visual inspection. Struct. Infrastruct. Eng. 7 (12), 895–905.

Selig, E.T., Cardillo, G.M., Stephens, E., Smith, A., 2008. Analyzing and forecasting railway data using linear data analysis. Comput. Railways XI.

Soleimani, H., Moavenian, M., 2017. Tribological aspects of wheel-rail contact: a review of wear mechanisms and effective factors on rolling contact fatigue. Urban Rail Transit 3 (4), 227–237.

Specht, C., Koc, W., Chrostowski, P., et al., 2017. The analysis of tram tracks geometric layout based on mobile satellite measurements. Urban Rail Transit 3 (4), 214–226.

Sui, G., 2009. Mileage calibration algorithm of track geometry data. J. Transport Inf. Saf.

Tsunashima, H., 2008. Condition monitoring of railway tracks using in-service vehicles: development of probe system for track condition monitoring. J. Mech. Syst. Transport. Log. 3 (1), 154–165.

Vu, A., Ramanandan, A., Chen, A., Farrell, J.A., Barth, M., 2012. Real-time computer vision/DGPS-aided inertial navigation system for lane-level vehicle navigation. IEEE Trans. Intell. Transp. Syst. 13 (2), 899–913.

Weston, P., Ling, C., Goodman, C.J., Li, P., Goodall, R.M., 2007. Monitoring vertical track irregularity from in-service railway vehicles. J. Rail Rapid Transit 221, 75–88.

Xu, P., 2012. Mileage Correction Model for Track Geometry Data from Track Geometry Car & Track Irregularity Prediction Model. Ph.D. thesis. Beijing Jiaotong University, Beijing, China.

Xu, P., Liu, R., Sun, Q., Wang, F., 2012. A novel short-range prediction model for railway track irregularity. Discrete Dyn. Nat. Soc. 2012 (2012), 1951–1965.

Xu, P., Liu, R., Sun, Q., Jiang, L., 2015. Dynamic-time-warping-based measurement data alignment model for condition-based railroad track maintenance. IEEE Trans. Intell. Transp. Syst. 16 (2), 799–812.

Xu, P., Sun, Q., Liu, R., Souleyrette, R.R., 2016. Optimal match method for milepoint postprocessing of track condition data from subway track geometry cars. J. Transp. Eng. 142 (8), 04016028.

Xu, P., Sun, Q.X., Liu, R.K., Wang, F.T., 2011. A short-range prediction model for track quality index. Proc. Inst. Mech. Eng. Part F J. Rail Rapid Transit 225 (3), 277–285.

Xu, P., Sun, Q.X., Liu, R.K., Wang, F.T., 2013. Key equipment identification model for correcting milepost errors of track geometry data from track inspection cars. Transport. Res. Part C Emerging Technol. 35 (9), 85–103.