[8] P. Viola and M. Jones, "Robust real-time face detection," *Int. J. Comput. Vis.*, vol. 57, no. 2, pp. 137–154, May 2004.
[9] M. Carrasco, L. Pizarro, and D. Mery, *Bimodal Biometric Person Identification System Under Perturbations*. Berlin, Germany: Springer-Verlag, 2007, p. 114.
[10] W. Zhao, R. Chellappa, P. Phillips, and A. Rosenfeld, "Face recognition: A literature survey," *ACM Comput. Surv. (CSUR)*, vol. 35, no. 4, pp. 399–458, Dec. 2003.
[11] G. García-Busnter and M. Torres-Torriti, "Effective pedestrian detection and counting at bus stops," in *Proc. IEEE Latin Amer. Robot. Symp.*, Oct. 2008, pp. 158–163.
[12] F. Delgado, J. C. Muñoz, R. Giesen, and A. Cipriano, "Real-time control of buses in a transit corridor based on vehicle holding and boarding limits," in *Proc. 88th Annu. Meeting Transp. Res. Board*, Jan. 2009, pp. 59–67.
[13] C. Cortés, D. Sáez, E. Sáez, A. Núñez, and A. Tirachini, "Hybrid predictive control strategy for a public transport system with uncertain demand," in *Proc. 6th TRISTAN*, Jun. 2007, 6 p.
[14] Poynting, RFID UHF Patch Antenna (PATCH-A0025), 2008. [Online]. Available: http://www.poynting.co.za/
[15] *INfinity 510 Reader Datasheet*, Sirit, Toronto, ON, Canada, 2008.
[16] Altelicon, CA-240 Coaxial Cable. [Online]. Available: http://www.altelicon.com/
[17] Laitan Holding Corp. [Online]. Available: http://www.laitan.ca/
[18] *ALN-9534 2 × 2 Inlay—Product Overview*, Alien Technology, Morgan Hill, CA, 2008. [Online]. Available: http://www.alientechnology.com/docs/products/
[19] S. M. Ross, *Introduction to Probability and Statistics for Engineers and Scientists*. Amsterdam, The Netherlands: Elsevier, 2004.

Modeling and Algorithms of GPS Data Reduction for the Qinghai–Tibet Railway

Dewang Chen, *Member, IEEE*, Yun-Shan Fu, Baigen Cai, and Ya-Xiang Yuan

***Abstract*—Satellites are currently being used to track the positions of trains. Positioning systems using satellites can help reduce the cost of installing and maintaining trackside equipment. This paper develops a nonlinear combinatorial data reduction model for a large amount of railway Global Positioning System (GPS) data to decrease the memory space and, thus, speed up train positioning. Three algorithms are proposed by employing the concept of looking ahead, using the dichotomy idea, or adopting the breadth-first strategy after changing the problem into a shortest path problem to obtain an optimal solution. Two techniques are developed to substantially cut down the computing time for the optimal algorithm. The surveyed GPS data of the Qinghai–Tibet railway (QTR) are used to compare the performance of the algorithms. Results show that the algorithms can extract a few data points from the large amount of GPS data points, thus enabling a simpler representation of the train tracks. Furthermore, these proposed algorithms show a tradeoff between the solution quality and computation time of the algorithms.**

***Index Terms*—Data reduction, Global Positioning System (GPS), heuristic algorithms, Qinghai–Tibet railway (QTR), shortest path problem.**

I. Introduction

As satellite positioning has many advantages (e.g., low cost, real time, and no cumulative errors) [1], it is often used in car-navigation systems [2] or in generating electronic maps [3]. Furthermore, satellites are currently used to track the positions of trains instead of using radio frequency systems [4] or track circuits. Positioning systems using satellites can help in reducing the cost of installing and maintaining trackside equipment [5]. Recently, the European Union has launched many projects (e.g., GADEROS [6] and RUNE [7]) using satellite positioning for low-density railways. In the United States, an incremental train control system using GPS positioning was employed in a Michigan railway [8]. The results of these projects showed that satellite positioning has a better performance–cost ratio for low-density railways. In China, GPS positioning was first adopted in the train-control system for the Qinghai–Tibet railway (QTR) in

Manuscript received March 7, 2008; revised January 1, 2009 and June 22, 2009; accepted March 30, 2010. Date of publication May 24, 2010; date of current version September 3, 2010. This work was supported in part by the National Natural Science Foundation (NSFC) of China under Grant NSFC 60776833, Grant NSFC 60634010, and Grant NSFC 60736047. Some of the work presented in this paper was completed during the visit of D. Chen with the Institute of Computational Mathematics, Chinese Academy of Sciences, Beijing, China, in 2008 and with the Berkeley Initiative in Soft Computing Program, Department of Electrical Engineering and Computer Science, University of California, Berkeley, in 2009. The Associate Editor for this paper was S. Sun.

D. Chen and B. Cai are with the State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing 100044, China (e-mail: dwchen@bjtu.edu.cn; bgcai@bjtu.edu.cn).

Y.-S. Fu and Y.-X. Yuan are with the State Key Laboratory of Scientific and Engineering Computing, Chinese Academy of Sciences, Beijing 100080, China (e-mail: fuys@lsec.cc.ac.cn; yyx@lsec.cc.ac.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2010.2048030

2006, where no track circuits are present. It greatly reduced trackside equipment and maintenance costs in the world's highest railway [9].

Digital track maps (DTMs) with high precision are the basis for accurate train positioning. With the help of DTMs, positioning error is reduced, and positioning reliability is enhanced [10]. Apparently, DTMs generated from GPS track data can easily be used in train positioning. After a large amount of satellite location data points with a high precision for railway tracks has been obtained, effective data reduction algorithms should be developed to select as few data points as possible to describe railway tracks approximately within the allowable error.

Railway tracks can be divided into straight tracks and curved tracks. Using curve representation methods [11] to represent curved tracks will result in the increase of the data storage; in particular, the corresponding map-matching algorithm will be complicated. As the curvature radii of the curved tracks are very big and the connecting parts between curved tracks and straight tracks are smooth, the line segments formed by a few points in the curved track can represent the curved track itself, and the resulting error can be controlled within a certain range [12]. Therefore, some key points in the railway tracks can be selected to form the line segments to approximate the railway tracks.

In Section II, a mathematical model of GPS data reduction is developed based on the combinatorial aspects of the problem. In Section III, two heuristic algorithms are proposed, an optimal algorithm is developed after changing this problem into a shortest path problem, and five performance measures are defined to evaluate the algorithms. In Section IV, the performance of the three algorithms are compared and analyzed using two surveyed GPS data sets of a QTR section. The results show that the algorithms are highly effective and that there is a tradeoff between solution quality and computational time. Conclusions and further research are given in Section V.

II. Data Reduction Modeling for Global Positioning System Data Set

As aforementioned, the data reduction problem for railway GPS data is solved by selecting a minimal number of data points from the surveyed data set to form the connected lines that represent railway tracks in a simpler way. The constraint condition for the selected data points is that the orthogonal distance from each data point to the corresponding line is less than an error bound. A mathematical model based on combinatorics [13] for this problem can be described by

$$\min \quad \sum_{i=1}^{N} t_i \tag{1}$$

s.t.

$$J = \{i_1, \ldots, i_m\} \subseteq \{1, \ldots, N\} \tag{2}$$

$$1 = i_1 < i_2 < \cdots < i_{m-1} < i_m = N \tag{3}$$

$$dist\,(L(i_j, i_{j+1}), k) \le \delta, \quad \text{if } i_j < k < i_{j+1} \tag{4}$$

$$t_i = 1, \quad \text{if } i \in J; t_i = 0, \quad \text{if } i \notin J \tag{5}$$

$$dist\,(L(i,j), k) = \sqrt{\|p_k - p_i\|_2^2 - \frac{\langle p_k - p_i, p_j - p_i \rangle^2}{\|p_j - p_i\|_2^2}} \tag{6}$$

where (1) is the objective function, N is the number of all data points in the data set for reduction, J is the set of sequence numbers of the key points selected, δ is the error bound, p_i is the ith data point in the data set, $L(i,j)$ is the line segment from p_i to p_j, and $dist(L(i,j),k)$ is the orthogonal distance from point p_k to the line segment $L(i,j)$.

In addition to the first point p_1 and the last point p_N, there are $N-2$ points in the data set that can be selected into the set J. Among the $N-2$ data, t_i is equal to 1 if p_i is selected; otherwise, t_i is equal to 0. Thus, there are 2^{N-2} possible choices in the feasible set, which makes it very hard to get an optimal solution for a large-scale problem only by enumerating.

Taking a QTR railway section as an example, N will be several thousands. As (6) is nonlinear, the data reduction problem is a large-scale nonlinear combinatorial problem. To make the descriptions of the algorithms clear, two definitions are given.

1) $\overline{p_i p_j}$, which is the line segment from p_i to p_j, is said to satisfy the δ condition if $dist(L(i,j),k)) \le \delta$, for all $i < k < j$.
2) $\overline{p_i p_j}$ is said to have a deviation of γ if $\max_{i<k<j} dist\,(L(i,j),k)) = \gamma$, denoting the deviation by $D(i,j) = \gamma$.

III. Algorithms and Performance Measures

A. Algorithm 1—Employing the Concept of Looking Ahead

The basic idea of this algorithm is to search from one end of the railway to the other end, employing the concept of looking ahead. After the first point is selected (which means that $i_1 = 1$), the second point i_2, which is chosen from $\{2, 3, \ldots, N\}$, becomes as large as possible until the last point N is reached or until the next point after i_2 will make the deviation larger than δ. Once i_2 is chosen, i_3 can similarly be chosen. The algorithm stops when point N is reached. The description of this algorithm is given as follows.

Step 0 Given N points, given $\delta > 0$.
Step 1 $k = 1$, $i_k = 1$.
Step 2 If $D(i_k, N) <= \delta$, then $k = k+1$, $i_k = N$, stop. Otherwise, for $j = i_k + 1 : N$ find the first j that makes $D(i_k, j) > \delta$, $k = k+1$, $i_k = j-1$.
Step 3 Repeat Step 2.

Let $J = \{i_1, i_2, \ldots, i_m\}$ be the solution obtained from Algorithm 1. From the process of the algorithm, it is not difficult to know that the computation time of the algorithm is at most quadratic. Define $s = \max_j (i_{j+1} - i_j)$; the number of computational steps required for Algorithm 1 is

$$o(sN) \approx o(N^2/m) < o(N^2). \tag{7}$$

B. Algorithm 2—Based on Dichotomy

As Algorithm 1 only considers the local data, another heuristic algorithm based on dichotomy is proposed so that the holistic characteristics of the data set are taken into consideration. The basic idea of the algorithm is to judge whether the data set needs to be divided into two smaller sets or not.

Starting with the whole data set, if $D(1, N)$ is greater than δ, it is necessary to divide it into two subsets. For each subset, the same procedure can be continued until all subsets do not need to be divided. This is like the generation of a binary tree if we view the whole data set as the root node and the subsets as the leaf nodes. The description of this algorithm is given as follows.

Step 0 Given N points, given $\delta > 0$.
Step 1 $J = \{1, N\}, m = 2$.
Step 2 Let the elements of J be ordered as

$$1 = i_1 < i_2 < \cdots < i_{m-1} < i_m = N.$$

Step 3 If $D(i_j, i_{j+1}) \le \delta$ holds for all $j = 1 : m$ then stop. Otherwise, for $j = 1 : m$, if there exists a point j which makes $D(i_j, i_{j+1}) > \delta$, then point k is added into J, where point k satisfies $dist(L(i_j, i_{j+1}), k) = D(i_j, i_{j+1})$, $m = m+1$, go to Step 2.

Let $J = \{i_1, i_2, \ldots, i_m\}$ be the solution obtained from Algorithm 2. Define $s = \min_j(i_{j+1} - i_j)$, the estimation for the computation cost of Algorithm 2 is

$$O\left(N \times \log_2(N/s)\right) \approx O(N \times \log_2 q). \tag{8}$$

As $q = N/s$ is much less than N, the computational complexity of Algorithm 2 is almost linear.

C. Algorithm 3—Employing Breadth-First Strategy

As neither of the aforementioned two algorithms optimally solves the data-reduction problem, we present a reformulation of the data-reduction problem as a shortest path problem. If $\overline{p_i p_j}$ satisfies the δ condition, we connect p_i and p_j with a directed arc from p_i to p_j. For this specific problem, we can define that every arc has the same weight of 1. A weighted directed graph is generated when all such line segments are found.

Let A be the graph's adjacency matrix, then $A(i,j) = 1$ if $\overline{p_i p_j}$ satisfies the δ condition. Apparently, finding the shortest path from p_1 to p_N is equivalent to minimizing the number of points that satisfy (2)–(6). Therefore, the data-reduction problem is converted into a shortest path problem of a weighted directed graph having the same weight of 1.

In the worst case, all the possible line segments need to be checked in the process of generating the directed graph. For point p_i, we should check $\overline{p_i p_j}$ with k $(k = j - i - 1)$ projection computations in (6). As a result, we need $(N - i - 1)(N - i)/2$ computations for all $\overline{p_i p_j}$ $(j = i + 2 : N)$. Calculating all $\overline{p_i p_j}$ $(i = 1 : N - 2, j = i + 2 : N)$, the total computations are $\sum_{i=1}^{N-2}((N - i - 1)(N - i)/2)$, which is equal to $N^3/6 - N^2/2 + N/3$. As N is about 5000 in this paper, it will take too much time if all the line segments are checked.

Therefore, we propose a 2δ technique to reduce the computing time and memory space. If $\overline{p_i p_j}$ does not satisfy the δ condition, the violation of the deviation restriction will be observed. We sequentially check $\overline{p_i p_{j+k}}$ until $D(i, j + k)$ is greater than 2δ. If $D(i, j + k)$ is greater than 2δ, then we turn to check $\overline{p_{i+1} p_{i+3}}$. This will preserve the possibility of finding a point beyond p_j that can satisfy the δ condition. Because the railway tracks are usually straight or curved with big curvature radii, it is almost impossible to find a point p_{m+k} that makes $D(i, m + k)(k \geq 1)$ less than δ when $D(i, m)$ is greater than 2δ. See Appendix A for the mathematical proof of this proposition.

Many classical algorithms (e.g., the Dijkstra algorithm and the Flord algorithm) solve different kinds of shortest path problems [14]. Among them, the breadth-first algorithm specializes in the directed graph whose arcs have the same weight. Based on the breadth-first strategy, Algorithm 3 is proposed as follows.

Step 0 Set $k = 0$, $i = 1$, $V(0) = \{p_1\}$.

Step 1 If $i = N$, go to Step 5, else go to Step 2.

Step 2 If $p_i \notin V(k)$, then set $i = i + 1$ and go to Step 1, else set $j = i + 1$ and go to Step 3.

Step 3 If $p_j \notin V(k) \bigcup V(k-1) \bigcup \cdots \bigcup V(0)$ and $D(i,j) \leq \delta$, then set $V(k+1) = V(k+1) \bigcup p_j$ and set p_i as p_j's previous point, go to Step 6, else set $j = j + 1$ and go to Step 4.

Step 4 If $j = N$, then $i = i + 1$ and go to Step 1, else set $j = j + 1$ and go to Step 3.

Step 5 Set $k = k + 1$, $V(k + 1) = \emptyset$ and $i = 1$, go to Step 2.

Step 6 If $j = N$, then stop, else set $j = j + 1$ and go to Step 3.

Finally, the shortest path is obtained by reversely finding the previous points from p_N to p_1.

Just as the 2δ technique, the proposed V technique employed in Step 3 is also to reduce projection computations. As every set $V(k)$ is a level set, the points in each $V(k)$ have the same shortest distance to the original point p_1. Evidently, if p_j is already in some previous level set $V(k)$, then it is not necessary to consider p_j when generating the newest level set $V(k+1)$. To be specific, we do not check $\overline{p_i p_j}$ when p_i is in the current level set $V(k)$ and p_j is already in some previous level set. Thus, the projection computations to check $\overline{p_i p_j}$ are reduced.

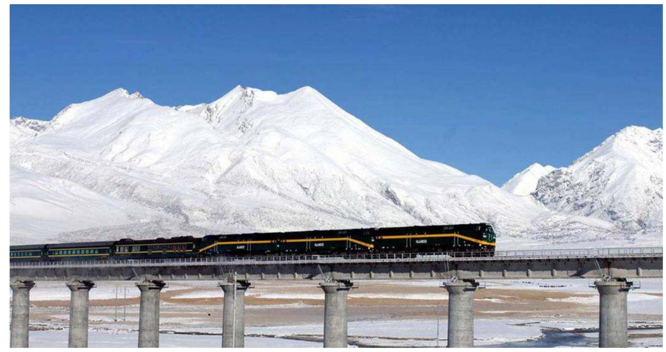

Fig. 1. Train on the QTR.

D. Performance Measures for Algorithms

The distance between the middle lines of the two adjacent railways is about 5 m. Theoretically, if δ is less than 2.5 m, it can satisfy the requirement for distinguishing the adjacent railways. In this paper, δ is set as 0.5 or 1 m to ensure safety. In this section, five performance measures are defined as follows to evaluate the performance of different algorithms.

1) $r(\%)$ is defined as

$$r = 100 \times \frac{(N - m)}{N} \tag{9}$$

where m is the number of key points found by the algorithms, and N is the number of all data points. Therefore, r reflects the efficiency of data reduction.

2) $e_l(‱)$ is defined as

$$e_l = 10\,000 \times \left(1 - \frac{\sum_{j=1}^{m-1} \left\| p_{i_{j+1}} - p_{i_j} \right\|_2}{\sum_{i=1}^{N-1} \left\| p_{i+1} - p_i \right\|_2} \right). \tag{10}$$

It reflects the loss in length.

3) $e_a(m)$ is defined as the average of all orthogonal distances from data points to the corresponding line segments.

4) $e_m(m)$ is defined as the maximum of all orthogonal distances from data points to the corresponding line segments. It is used to check whether (4) is satisfied or not.

5) $t(s)$ is the running time of the algorithms. If it is not too long, it is acceptable as the algorithms are run offline.

IV. Qinghai–Tibet Railway Case Study

A. Data Description

The QTR mentioned in this paper is a high-altitude railway that connects Golmud in Qinghai Province to Lhasa in the Tibet Autonomous Region, which was inaugurated on July 1, 2006. Fig. 1 illustrates a train that ran on the QTR after the inauguration. Fig. 2 shows the major railway stations on the QTR and the main mountains alongside the QTR. The QTR includes the Tanggula railway station at 5072 m above sea level, which is the world's highest railway station, and the Fenghuoshan tunnel at 4905 m above sea level, which is the highest rail tunnel in the world.

Fig. 2. Vertical sectional diagram of the QTR.

TABLE I
PROJECTION COMPUTATIONS REDUCED WITH OR WITHOUT THE 2δ AND V TECHNIQUES

Projection computations	Without 2δ technique		With 2δ technique			
			Without V technique		With V technique	
	$\delta=0.5$	$\delta=1$	$\delta=0.5$	$\delta=1$	$\delta=0.5$	$\delta=1$
data set 1	1.6446e+10	1.6446e+10	2.1402e+08	3.1271e+08	0.8638e+08	1.4150e+08
data set 2	2.0399e+10	2.0399e+10	1.5130e+09	2.6602e+09	8.0326e+08	8.1796e+08

Approximately 500 000 GPS data points with 1-cm precision were obtained from differential GPS technology along the QTR with a length of 1142 km. The distance of the adjacent points is between 1.5 and 3 m, and the average distance is about 2.5 m.

The GPS data points from the Tanggula railway section, which includes two parts extending from the Tanggula railway station but in the opposite directions, are selected to validate the performance of the algorithms. Latitude, longitude, and height GPS data are transformed into 3-D coordinates at first, and then, the beginning point of the railway section is set as the origin of the reference coordinates. The length of the selected railway section is 24 km, including 9587 GPS data points.

In the operation of the QTR, the direction from the Golmud railway station to a certain railway station along the QTR is defined as the in-station direction, and the direction from that certain station to the Lhasa railway station is defined as the out-station direction, respectively. According to the direction definition, the 9587 data points are divided into two data sets. One refers to the in-station direction section of the Tanggula station, which includes 4622 data points; the other refers to the out-station direction section of the Tanggula station, which includes 4966 data points.

TABLE II
PERFORMANCE COMPARISON OF THREE ALGORITHMS FOR DATA SET 1

δ	0.5m			1m		
Algorithm	1	2	3	1	2	3
$r\%$	97.86	96.97	97.94	98.53	98.05	98.55
$e_l(‰)$	0.79	0.39	0.77	1.60	0.99	1.61
$e_a(m)$	0.30	0.15	0.31	0.57	0.34	0.61
$e_m(m)$	0.50	0.50	0.50	1.00	0.97	1.00
$t(s)$	50.5	16.4	3684	63.5	13.7	5998

TABLE III
PERFORMANCE COMPARISON OF THREE ALGORITHMS FOR DATA SET 2

δ	0.5m			1m		
Algorithm	1	2	3	1	2	3
$r(\%)$	99.54	99.34	99.58	99.68	99.60	99.70
$e_l(‰)$	0.092	0.051	0.092	0.18	0.098	0.17
$e_a(m)$	0.29	0.13	0.29	0.56	0.33	0.49
$e_m(m)$	0.50	0.45	0.50	1.00	0.87	1.00
$t(s)$	149.4	16.1	23131	253.5	13.9	23532

B. Comparison and Analysis of Computational Results

Before comparing the performance of the three algorithms, some results are given about projection computations, which are reduced by using the 2δ technique in creating a directed graph and the V technique in computing the shortest path.

Table I demonstrates how the 2δ technique can effectively cut down projection computations. By singly using the 2δ technique, more than 95% of the computations are reduced for data set 1 and more than 85% for data set 2. After that, about 50% of the computations are reduced for both data set 1 and data set 2 by the V technique. Therefore, the two techniques can greatly cut down the running time of Algorithm 3.

The computation results are obtained by a laptop (Pentium 1.6 GHZ, 512 MB) using Matlab 7.0. The following conclusions are drawn from Tables II and III.

1) r is very close to 1 for all the three algorithms, which illustrates that the proposed algorithms are highly effective for the GPS data reduction problem. In other words, there are too many redundant data points among the large amount of surveyed GPS data points. Algorithms 3, 1, and 2 rank first, second, and third, respectively, with regard to the reduction rate.
2) Algorithm 3 runs much longer than algorithm1 1 and 2, although algorithm 2 runs a little quicker than algorithm 1. For algorithm 3, the running time rapidly increases for data set 2 as it contains long straight lines. In this case, there are much more feasible paths.
3) As the shape of data set 1 is more curved than that of data set 2, the r of data set 1 is smaller than that of data set 2 under the same δ. With the increase of δ, the reduction rate r will accordingly increase for all the three algorithms.
4) e_l is very small for all the algorithms. For the selected railway station, the loss in length is only about 1 or 3 m if δ is 0.5 or 1 m, respectively. Therefore, using straight lines to describe railway tracks is applicable.
5) e_m and e_a show that the constraint equation (4) holds true for all the three algorithms. Algorithms 1 and 3 have made full use of the constraint; however, Algorithm 2 has not.

Due to the page limitations, only some running results are chosen for the purpose of demonstration. As the changes in height are very small for the two data sets, only the running results in the xy plane are illustrated in Figs. 3–6, where the dots represent the key points found by the algorithms. The four figures all demonstrate that the density of

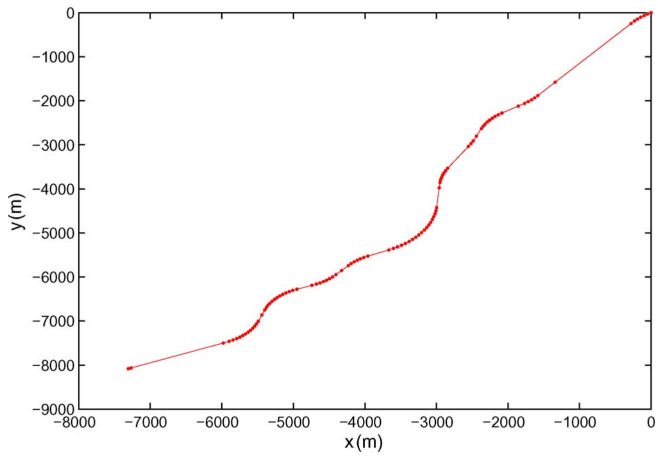

Fig. 3. Key points in data set 1 found by Algorithm 1 for a 0.5-m error bound.

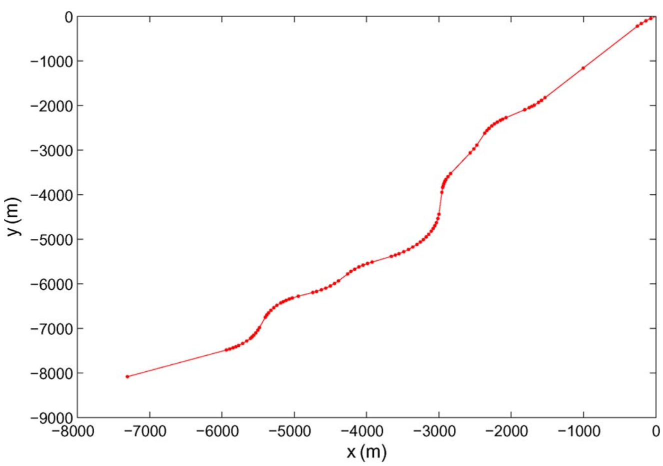

Fig. 4. Key points in data set 1 found by Algorithm 2 for a 1-m error bound.

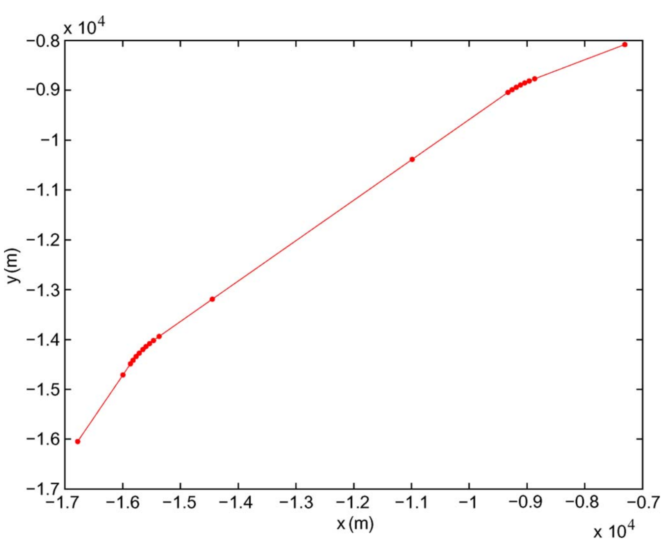

Fig. 5. Key points in data set 2 found by Algorithm 3 for a 0.5-m error bound.

dots is relatively higher in the curved railway track than in the straight railway track, which means that fewer points are needed to describe the straight railway, but more points are needed in describing the curved railway. It is worth pointing out that the research results in this paper has been put into engineering applications after many field tests in the QTR, as illustrated in Fig. 7.

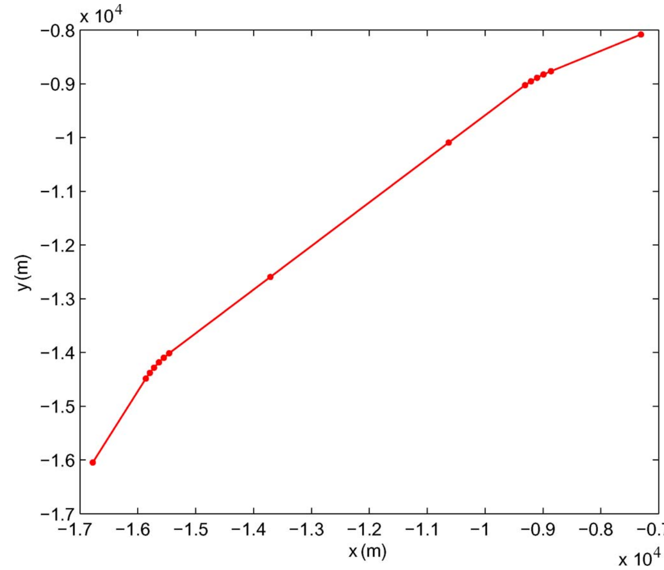

Fig. 6. Key points in data set 2 found by Algorithm 1 for a 1-m error bound.

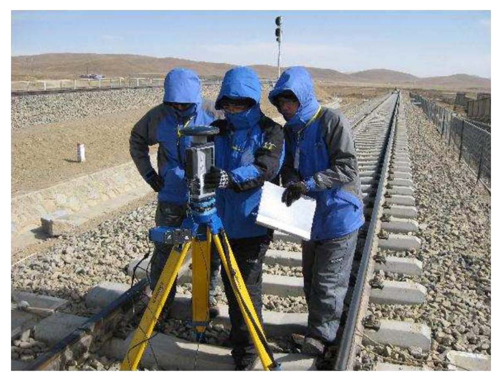

Fig. 7. Field test for algorithms on the QTR.

V. Conclusion

Through modeling and analysis, we have found that the railway GPS data reduction is a large-scale combinatorial problem. Two heuristic algorithms and an optimal algorithm have been proposed in this paper. Two techniques have been developed to reduce the computing time for the optimal algorithm. Two data sets from the surveyed GPS data of a QTR railway section have been used to compare and analyze the performance of the algorithms.

Results show that the heuristic algorithms run much quicker than the optimal one. After considering the reduction rate and the running time together, we recommend Algorithm 1 for the large-scale data set. Algorithm 2 can be used to validate whether there is any zigzag shape data in the data set, although this is not likely to occur with a railway. Therefore, for this special problem, a good heuristic algorithm is very close to the optimal one in performance, while it also greatly enhances the efficiency.

In general, for the railway GPS data-reduction problem, the bigger the error bound, the straighter the railway, and the higher the data-reduction rate will become. According to the results found in this paper, if δ is 0.5 m, only about 6300 key data points among all of the 500 000 data points are needed to describe the QTR in a simpler way, and the cumulative length error is about 65 m. If δ is 1 m, only about 4500 key data points are needed, and the cumulative length error is about 250 m. Therefore, the proposed data-reduction algorithms will effectively save memory space and thus increase the speed of the train positioning. Based on the data reduction, the real-time train positioning algorithms still need further research.

APPENDIX A
PROOF FOR THE 2δ TECHNIQUE

Under Assumption 1.1, we can prove the following Lemma 1.2. This lemma implies that no feasible line segments are missed when generating adjacency matrix with the 2δ technique.

Assumption 1.1: If $\overline{p_i p_h}$ satisfies the 2δ condition for all $h = i+1, \ldots, j$, then

$$\|p_{j+1} - p_i\|_2 \geq \|p_h - p_i\|_2 + \delta. \tag{11}$$

If the data points are given in the same order as the GPS data points in this paper, this assumption is very reasonable. Assume that δ is small. What our assumption really means is the following.

If all the GPS data points between points i to j are approximately on a straight line (within 2δ error), then the next GPS point will be at least one δ distance away.

In a railway, it means that if a given railway segment is approximately a straight line from its starting point p_i to the end point p_j, then the train will generally go on roughly along the same direction to the next GPS point p_{j+1} (from p_j to p_{j+1}), which means that point p_{j+1} will be further away by at least one δ.

Lemma 1.2: Under Assumption 1, if $\overline{p_i p_{j+1}}$ does not satisfy the 2δ condition, then for any $l \geq j+1$, $\overline{p_i p_l}$ does not satisfy the δ condition, which means that $A(i,l) = 0$ for all $l \geq j+1$.

Proof: If $\overline{p_i p_{j+1}}$ does not satisfy the 2δ condition, there must exist p_k so that

$$dist\,(L(i,j+1),k) > 2\delta, \qquad i < k < j+1. \tag{12}$$

Without loss of generality, we assume that $D(i,l) \leq 2\delta$, for all $l = i+1, \ldots, j$. We define the set $S_1 = \{p :$ the projection distance from p_{j+1} to the line segment $\overline{p_i p}$ is no more than δ and $\langle p_{j+1} - p_i, p - p_i \rangle \geq 0\}$

$$S_0 = \{Q : Q = p - p_i, p \in S_1\} = S_1 - \{p_i\} \tag{13}$$

where S_0 is a convex cone. Let $S_2 = \{p_i\} - S_0$, $S = S_1 \cup S_2$. Projecting p_k to set S and the line segment $\overline{p_i p_{j+1}}$ separately, we get the projective point Q_1 and Q_2; projecting Q_1 to the line $\overline{p_i p_{j+1}}$, we get Q_3; projecting p_{j+1} to the line $\overline{p_i Q_1}$, we get Q_4; and projecting Q_4 to line $\overline{p_i p_{j+1}}$, we get Q_5.

Since Q_2 is the projection point from p_k to the line segment $\overline{p_i p_{j+1}}$, from Assumption 1.1, we have

$$\|p_i - Q_2\| \leq \|p_i - p_k\| \leq \|p_i - p_{j+1}\| - \delta. \tag{14}$$

Similarly, we have

$$\|p_i - Q_1\| \leq \|p_i - p_k\| \leq \|p_i - p_{j+1}\| - \delta \tag{15}$$
$$\|p_i - Q_3\| \leq \|p_i - p_k\| \leq \|p_i - p_{j+1}\| - \delta. \tag{16}$$

Since Q_5 is the projection point from Q_4 to $\overline{p_i p_{j+1}}$, then Q_5 must lie within the ball $B(p_{j+1}, \delta)$, and $\|Q_5 - p_{j+1}\| < \delta$; hence, $\|p_i - Q_5\| > \|p_i - p_{j+1}\| - \delta$. It follows that

$$\|p_i - Q_3\| \leq \|p_i - Q_5\| \tag{17}$$
$$\|Q_1 - Q_3\| < \|Q_4 - Q_5\| < \|p_{j+1} - Q_4\| = \delta. \tag{18}$$

Since $\|p_k - Q_2\| < \|p_k - Q_1\| + \|Q_1 - Q_3\|$ and $\|p_k - Q_2\| > 2\delta$, we obtain $\|p_k - Q_1\| > \delta$. Assuming that there is a point p_l $(l > j+1)$ so that $\overline{p_i p_l}$ satisfies the δ condition, then p_l must be in the set S. However, as the projection distance from p_k to S is larger than δ, $dist(L(i,l), j+1)$ and $dist(L(i,l), k)$ cannot be less than δ at the same time. Therefore, it demonstrates that there cannot be a point p_l $(l > j+1)$ so that $\overline{p_i p_l}$ satisfies the δ condition. This completes our proof. ■

ACKNOWLEDGMENT

The authors would like to thank Prof. M. Grótschel of Zuse Institute Berlin, Berlin, Germany, for giving valuable suggestions on combinatorial optimization and the associate editor and the anonymous reviewers for their helpful comments and constructive suggestions, which greatly helped to improve this paper. The authors would also like to thank the Qinghai–Tibet Railway Company for providing GPS data and field test support and the Railway GPS Research Group of Beijing Jiaotong University for validating the research results in the harsh QTR field.

REFERENCES

[1] H. Blomenhofer, “GNSS in the 21st century—The user perspective,” *Acta Astronautica*, vol. 54, no. 11, pp. 965–968, Jun. 2004.

[2] I. Skog and P. Handel, “In-car positioning and navigation technologies—A survey,” *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 1, pp. 4–21, Mar. 2009.

[3] J. Zhang, D. Chen, and U. Kruger, “Adaptive constraint K-segment principal curves for intelligent transportation systems,” *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 4, pp. 666–677, Dec. 2008.

[4] A. Santos, A. Soares, F. Redondo, and N. Carvalho, “Tracking trains via radio frequency systems,” *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 2, pp. 244–258, Jun. 2005.

[5] R. George, M. Juliette, and B. Marion, “Innovation brings satellite-based train control within reach,” *Railway Gazette Int.*, vol. 160, no. 12, pp. 835–837, Dec. 2004.

[6] A. Urech, J.-P. Diestro, and O. Gonzalez, “GADEROS—A Galileo Demonstrator for Railway Operation System,” in *Proc. DASIA*, Dublin, Ireland, May 2002, pp. 442–447.

[7] A. Albanese, L. Marradi, G. Labbiento, and G. Venturi, “The RUNE project: The integrity performances of GNSS-based railway user navigation equipment,” in *Proc. ASME/IEEE Joint Rail Conf.*, Pueblo, CO, Mar. 2005, pp. 211–218.

[8] B. Jeff and C. John, “ITCS: A new approach to increasing capacity,” *Int. Railway J.*, vol. 45, no. 10, pp. 46–47, 2005.

[9] K. Li, “Plan of ITCS signal control system on Qinghai–Tibet railway,” *Chin. Railways*, no. 7, pp. 31–36, 2005.

[10] A. Simsky, F. Wilms, and J.-P. Franckart, “GNSS-based failsafe train positioning system for low-density traffic lines based on one dimensional positioning algorithm,” in *Proc. NAVITEC*, Noordwijk, The Netherlands, Dec. 2004, pp. 8–10.

[11] Y. Gu and T. Tjahjadi, “Coarse-to-fine planar object identification using invariant curve features and B-spline modeling,” *Pattern Recognit.*, vol. 33, no. 9, pp. 1411–1422, Sep. 2000.

[12] G. Gao and B. Cai, “Research on the automatic electronic map generation algorithm for the train supervision system,” *J. China Railway Soc.*, vol. 28, no. 1, pp. 63–67, 2006.

[13] R. A. Brualdi, *Introductory Combinatorics*, 4th ed. Englewood Cliffs, NJ: Prentice-Hall, 2004.

[14] S. Pemmaraju and S. Skiena, *Computational Discrete Mathematics: Combinatorics and Graph Theory With Mathematica*. Cambridge, U.K.: Cambridge Univ. Press, 2003.