

Research Article

Practical Constraint K-Segment Principal Curve Algorithms for Generating Railway GPS Digital Map

Dewang Chen¹ and Long Chen²

¹ State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, No. 3 Shang Yuan Cun, Hai Dian District, Beijing 10044, China

² Faculty of Science and Technology, University of Macau, Av. Padre Tomás Pereira Taipa, Macau

Correspondence should be addressed to Dewang Chen; dwchen@bjtu.edu.cn

Received 23 June 2013; Accepted 30 October 2013

Academic Editor: Yingwei Zhang

Copyright © 2013 D. Chen and L. Chen. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In order to obtain a decent trade-off between the low-cost, low-accuracy Global Positioning System (GPS) receivers and the requirements of high-precision digital maps for modern railways, using the concept of constraint K-segment principal curves (CKPCS) and the expert knowledge on railways, we propose three practical CKPCS generation algorithms with reduced computational complexity, and thereafter more suitable for engineering applications. The three algorithms are named ALLOpt, MPMopt, and DCopt, in which ALLOpt exploits global optimization and MPMopt and DCopt apply local optimization with different initial solutions. We compare the three practical algorithms according to their performance on average projection error, stability, and the fitness for simple and complex simulated trajectories with noise data. It is found that ALLOpt only works well for simple curves and small data sets. The other two algorithms can work better for complex curves and large data sets. Moreover, MPMopt runs faster than DCopt, but DCopt can work better for some curves with cross points. The three algorithms are also applied in generating GPS digital maps for two railway GPS data sets measured in Qinghai-Tibet Railway (QTR). Similar results like the ones in synthetic data are obtained. Because the trajectory of a railway is relatively simple and straight, we conclude that MPMopt works best according to the comprehensive considerations on the speed of computation and the quality of generated CKPCS. MPMopt can be used to obtain some key points to represent a large amount of GPS data. Hence, it can greatly reduce the data storage requirements and increase the positioning speed for real-time digital map applications.

1. Introduction

1.1. Principal Curves. Spearman proposed principal component analysis (PCA) in 1904, and now PCA is one of the most important tools for statistical data analysis [1]. But PCA is one linear analysis method which cannot deal with some intrinsic nonlinear data sets. Then Hastie proposed the concept of principal curves and its generation algorithm in 1983 [2]. As the nonlinear extension of PCA, principal curve uses a smooth curve instead of a straight line to summarize the given data set of the symmetric variables. Recently, the principal curves have been widely used in different real engineering problems such as control of the electron beam trajectory in Linear Collider [3], sound data model reduction, and data visualization in speech processing [4–6] and chemical process monitoring [7, 8].

In 1992, Banfield and Raftery proposed BR principal curves, which resolved the problem of original principal curves that the estimation error has significant effects on the actual principal curves due to the much large curvature in closed principal curves [9]. But BR principal curves algorithm also brings the numerical instability, which may obtain smooth but false principal curves in practice. In 1994 and 1995, Duchamp and Stuezel studied the entire differential geometric properties of principal curves [10, 11] and they found that the principal curves are not unique.

In 1997, Kégl proposed the concept of length constraint principal curves [12–14] and proved the existence and uniqueness of the K-segment principal curves (KPCS) defined in some theoretical distributions. He also studied the learning nature of KPCS about data distribution and proved the convergence of KPCS. Then, he proposed the related polygonal

line algorithm to obtain KPCS. This is the first article giving proofs of existence of principal curves, but the length of the curve must be prefixed. In 2000, Verbeek et al. proposed soft-K-segment principal curve algorithm [15] which applied local PCA to develop K-segments and then connected them smoothly to form principal curves. However, this method cannot be extended to multidimensional application and it did not consider the curvature penalty in the objective function. In 2011, Ozertem and Erdogmus proposed a new way of generating principal curve by probability density estimation [16]. For the data set with non-Gaussian process [17], generating principal curves is more difficult than that with Gaussian process. As for the parameter selection, Biau addressed the parameter selection issue using the empirical risk minimization perspective and model selection via penalization [18]. Gerber and Whitaker made a meaningful exploration in regularization-free principal curve estimation by a gradient-descent-based method for controlling the model complexity [19]. Zhang et al. applied granular computation on principal curves and better results were obtained [20]. These studies greatly enrich the theoretical research and applications of principal curves.

1.2. Constraint K-Segment Principal Curves (CKPCS) and GPS Railway Digital Maps. Motivated by some applications about transportation data with fixed endpoints, we have proposed the concept of constraint K-segment principal curves (CKPCS) and its generation algorithm in 2008 [21]. Unfortunately, the proposed algorithm is complicated and computing intensive; thus, its practicality is not very good. To decrease the computational complexity, we later proposed a heuristic optimization algorithm [22] for fusing multiple GPS track data with fixed endpoints, trying to make the CKPCS algorithm more practical and easier to be used. However, it only works for some simple curves.

GPS technology has been intensively used to guide aircraft, ships, vehicles, and trains to accurately reach destinations on time along the selected routes. Nowadays, accurate GPS digital maps play a very important role in railway control systems especially in precise positioning of trains [23]. The application of GPS digital maps in train positioning can effectively reduce positioning errors and improve positioning reliability [24]. To generate high quality GPS digital maps, accurate GPS data are to be collected by GPS receivers. However, the accuracy of GPS receivers may be deteriorated by many factors including track error, satellite clock difference, ionospheric error, troposphere error, carrier phase cycle slips, and multipath effects. Besides, the extremely complex electromagnetic environment in railway system also negatively affected the accuracy of GPS receivers. Consequently, it is difficult for low-cost GPS receivers to collect high-accuracy GPS data for railway digital mapping. Differential GPS system can be employed to obtain high-accuracy GPS data, but the cost for building such kind of digital mapping system is much higher compared with using normal GPS receivers.

Considering that there are some important static points like switches and signal machines in railways, long-time static measurement can be implemented to get their accurate

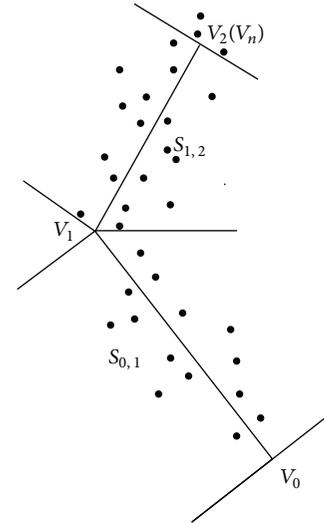


FIGURE 1: Principle of nearest neighbor.

positions. Or in other words, some endpoints with high-accuracy can be obtained. As a result, we conclude that the CKPCS with fixed endpoints is very suitable for digital mapping of railways. Using CKPCS algorithms to generate high-accuracy data with multiple GPS data collected by low-cost GPS receivers will help us to obtain a decent balance between low-cost devices and high-accuracy digital maps.

In this paper, based on the discussions above, we further our study on CKPCS by using some expert knowledge about railways. We will develop several heuristic CKPCS generation algorithms which are more computational efficient and easier to be used. By using simulation data and the real GPS data collected in Qinghai-Tibet railway, the new algorithms will be validated according to data accuracy, computing time, stability, and application scope. The remainder of this paper is organized as follows. The model of CKPCS based on the principle of nearest neighbor is presented in Section 2. The three practical CKPCS algorithms are developed in Section 3. After analyzing errors and fitness of proposed algorithms by simulation data and validating their practical applicability by railway GPS data sets, the advantages and disadvantages of different proposed algorithms are compared in Section 4. We conclude this paper with some discussion on future work in Section 5.

2. Mathematical Model for CKPCS

K-segment principal curves (KPCS) are generally the curves containing a series of connected segments that pass through the “middle” of collected data. KPCS are usually obtained by adding points gradually from the initial PCA solution and minimizing the distance from points to segments and vertices according to the principle of nearest neighbor. However, as constrained K-segment principal curves (CKPCS) have two fixed endpoints, the segment connecting the two endpoints composites the initial solution, and the two endpoints are kept unchanged during the optimization process of minimizing the distance from points to line segments and vertices. As illustrated in Figure 1, CKPCS consist of

vertices V_j ($j = 0, 1, 2, 3, \dots, n$) and segments $S_{j,j+1}$ ($j = 0, 1, 2, 3, \dots, n-1$), and the positions of endpoints V_0, V_n are fixed. According to the principle of nearest neighbor, all data points d_i ($i = 1, 2, 3, \dots, m$) are partitioned to their corresponding nearest segments and vertices. Here, n is the number of vertices of constraints principal curves and m is the number of GPS data points.

The distance from a data point d_i to one segment $S_{j,j+1}$ is defined as

$$DL_{i,j} = \frac{\left| (X_{v_j} - X_{v_{j+1}}) \cdot Y_{d_i} - (Y_{v_j} - Y_{v_{j+1}}) \cdot X_{d_i} + X_{v_{j+1}} \cdot Y_{v_j} - Y_{v_{j+1}} \cdot X_{v_j} \right|}{\sqrt{(X_{v_j} - X_{v_{j+1}})^2 + (Y_{v_j} - Y_{v_{j+1}})^2}} \quad (1)$$

Here, X_{d_i} and Y_{d_i} are X and Y coordinates of data point d_i . X_{v_j} and Y_{v_j} are X and Y coordinates of vertex V_j . And the segment $S_{j,j+1}$ connects V_j and V_{j+1} .

The distance from a data point d_i to the vertex V_j is defined as

$$DP_{i,j} = \sqrt{(X_{d_i} - X_{v_j})^2 + (Y_{d_i} - Y_{v_j})^2} \quad (2)$$

The smallest one of the distances $DL_{i,j}$ ($j = 0, 1, 2, 3, \dots, n$) and $DP_{i,j}$ ($j = 0, 1, 2, 3, \dots, n-1$) is defined as projection distance of a data point to the K -segment principal curve. The average projection distance of all data points AD is defined as follows and will be minimized during the CKPCS generation algorithm:

$$AD = \frac{\sum_{i=1}^m \min\{DL_{i,j} (j = 0, 1, 2, 3, \dots, n), DP_{i,j} (j = 0, 1, 2, 3, \dots, n-1)\}}{m} \quad (3)$$

To the problem of minimizing AD , both the number and positions of vertices are to be optimized. Because we do not know the number of vertices in advance, getting the optimal solution in one step is not easy and some iterative algorithms become a great need. Moreover, the practical engineering applications prohibit complex algorithms with long running time.

3. Three Iterative CKPCS Algorithms

According to the model above, this section proposes three iterative algorithms to generate CKPCS from noisy observed data. For all the three algorithms, two endpoints are supposed as the constrained points and kept unchanged in the iteration. But different algorithms imply different ways in finding the initial solution and adding other vertices according to our domain knowledge. Many successful applications have showed that good initial solutions and high computational efficiency could be achieved with the help of expert knowledge [25–27].

For all the three algorithms, when the vertex number is fixed, adjusting several or all the positions of vertices to

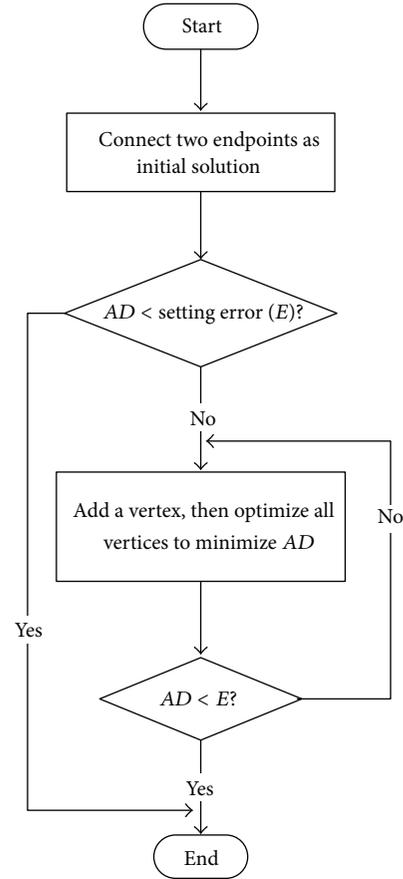


FIGURE 2: Flow chart of ALLOpt algorithm.

minimize the objective function AD is a classical nonlinear optimization problem, which is solved in this paper by the commercial nonlinear optimization routine “fmincon” provided by Matlab. The “fmincon” uses traditional SQP optimization [28] for small-scale problems and trust-region reflective method for large-scale problems [29].

3.1. Splitting Optimizing Algorithm: ALLOpt. In this algorithm, CKPCS is obtained by two major parts: dividing some segment and optimizing all vertices as shown in Figure 2. The main steps of this algorithm are listed as follows.

Step 1. Connect two endpoints and take the segment as an initial solution for the CKPCS.

Step 2. If AD is larger than the presetting error, go to the next step; otherwise, current curve is the obtained CKPCS and the algorithm is terminated.

Step 3. Add one new vertex which is the middle point of the line segment where the maximal AD is found among all line segments to generate a new solution. Then, the positions of all vertices are adjusted using a nonlinear optimization method to minimize the AD .

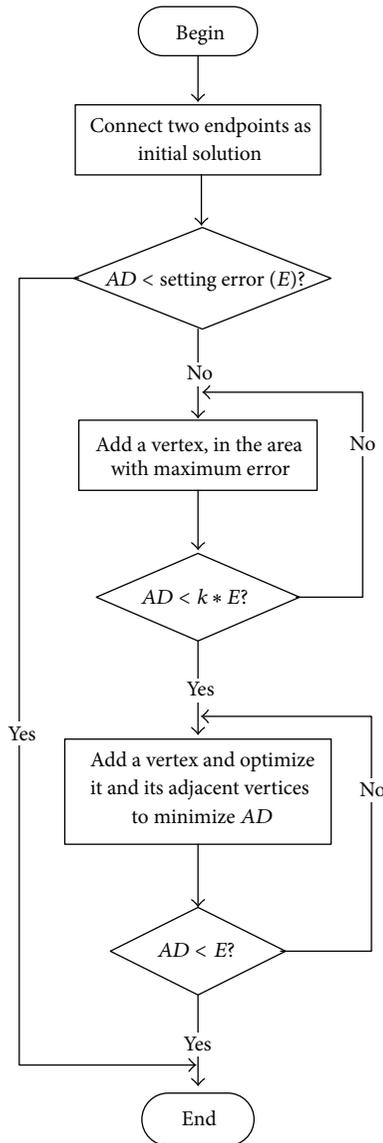


FIGURE 3: Flow chart of MPMopt algorithm.

Step 4. If AD is bigger than the setting error, go to Step 3; otherwise, the CKPCS is obtained. As this splitting optimization algorithm needs to optimize all vertices after adding a new one, we call it ALLOpt for short.

3.2. Improved Splitting Optimizing Algorithm: MPMopt. Because ALLOpt optimizes the positions of all vertices in each iteration, it needs long running time if the volume of data is large. To solve this problem, an improved algorithm is proposed by applying a better initial solution and adjusting at most three vertices once a time to accelerate the optimization process. As illustrated in Figure 3, this algorithm includes two loops: one loop for generating a candidate principal curve, and the other loop for finding the final CKPCS.

Loop A. Generate initial principal curve.

Step 1. Connect two endpoints, take the line as initial solution.

Step 2. Calculate the AD. If AD is bigger than the setting error, go to step three; otherwise, the initial solution is the CKPCS obtained.

Step 3. Add one vertex V_j in a heuristic way and generate a new solution. Then, calculate the AD; if the AD is bigger than k (usually between 2 and 4) times of the setting error, repeat this step until the AD is less than k times of the setting error.

The heuristic way of adding a new vertex is as follows: the area where its AD is maximum among all line segments is selected. Then, in this area, the point with the maximal projection distance among all points is selected as the new vertex. We call this method Max Point Method (MPM).

Loop B. Find the final CKPCS.

Step 4. Based on the initial CKPCS obtained in Loop A, the method MPM introduced above is applied to find the vertex $V_{j'}$.

Step 5. Locally optimize the positions of vertex $V_{j'}$ and the two adjacent vertices V_j and V_{j+1} , making the AD minimum, but do not change the positions of two endpoints.

Step 6. If AD is larger than the setting error, repeat Steps 4 and 5; otherwise, the final CKPCS is obtained.

This algorithm is called an MPMopt for short as its optimization is based on MPM method.

3.3. Dividing and Connecting Optimizing Algorithm: DCopt. Although MPMopt can run faster than the first algorithm ALLOpt, we find that it also has some disadvantages in handling the complicated curves with cross points. Therefore, we propose the third algorithm which contains three parts: dividing, connecting, and optimizing as shown in Figure 4. For short, the third algorithm is called DCopt.

Part 1. Dividing: divide all data points to the adjacent region of all vertices including two endpoints.

Step 1. According to the nearest neighbor principle, divide all data points into different vertices with adjacent regions.

Step 2. Add a new vertex at the each junction of two adjacent regions; then divide all data points again.

Step 3. Repeat Step 2 to keep dividing the adjacent ranges, until the maximal radius of all adjacent regions is less than the setting radius, which is m times of the setting error.

Part 2. Connecting: according to the shape of trajectory, connect the generated vertices orderly and obtain the initial solution.

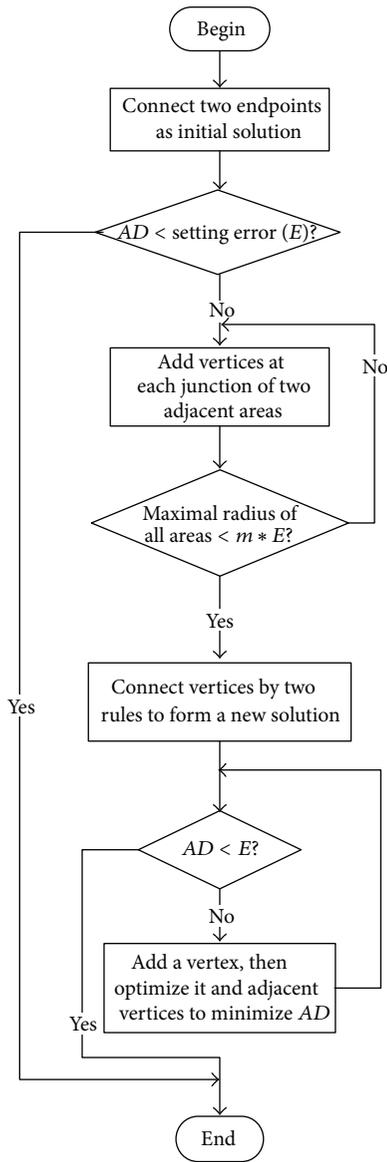


FIGURE 4: Flow chart of DCopt algorithm.

Two rules are followed in connecting to make the CKPCS smoothly and continuously as follows.

Rule One. the slope change of adjacent line segments should be as small as possible.

Rule Two. connect the nearest points.

In connecting part, the starting point follows rule two; while other points follow rule one.

Part Three. Optimizing.

If AD is bigger than the setting error, add one new vertex using MPM methods that is mentioned above and optimize it and its adjacent vertices except the two endpoints, until AD is less than the setting error.

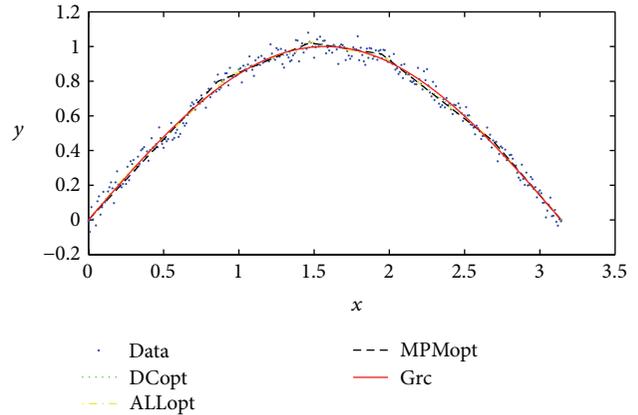


FIGURE 5: Error analysis for three algorithms by simulated sin data.

4. Verification, Analysis, and Comparison of Proposed Algorithms

In this section, some simulation data are used to analyze the error, stability, and fitness of the three proposed algorithms first. Then, their practicability is verified and analyzed using the measured GPS data on Qinghai-Tibet Railway (QTR). QTR is the highest railway in the world and also the first railway in China to adopt GPS technique for train positioning [30, 31].

We have applied some different thresholds for MPMopt and DCopt in our experiments. By trial-and-error, we find out that using k (ranging from 2 to 4) in MPMopt and setting the radius of each circle as the m (ranging from 3 to 7) times of the setting error can work very well in all experiments. We set k as 3 and m as 5 in this paper to keep the balance of accuracy and running time. Small k or small m means long time in finding initial solution but less time in iterative optimization. For the setting error E , which is often set manually, the less of its value means higher requirement in accuracy and longer computing time in optimization.

4.1. Analysis of Algorithms' Performance by Synthetic Data

4.1.1. Analysis of Algorithms' Error by Simple Synthetic Data. There are three steps to finish the analytic process for synthetic data illustrated in Figure 5. First, a reference curve is generated and two fixed endpoints are recorded; secondly, some data points near the curve are randomly generated to simulate the real measured data points with noise; finally, the new CKPCS are obtained by the proposed algorithms using the generated data points and the two endpoints. The obtained CKPCS are compared with the standard curve to compute performance indexes.

In Figure 5, the red curve is the generated reference curve (Grc) $y = \sin(x)$ with two fixed endpoints $(0, 0)$ and $(\pi, 0)$. The blue data points represent the generated random data points simulating the measured data points from the reference curve. The yellow curve, the black curve and the green curve are the three CKPCS generated by ALLOpt, MPMopt, and DCopt, respectively.

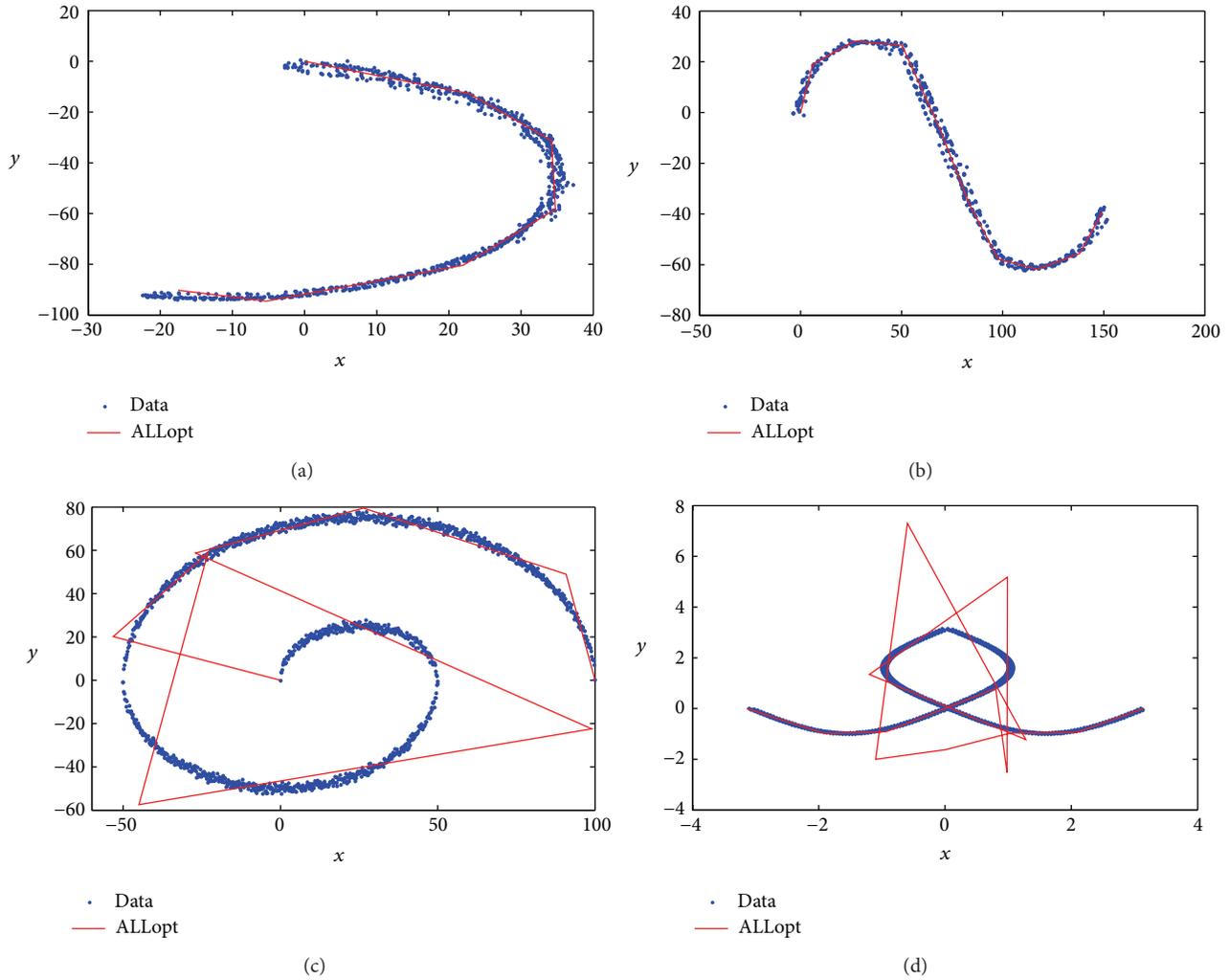


FIGURE 6: Applicability analysis for ALLOpt.

To test the performance of proposed algorithms, the simulation repeats 30 times for each algorithm. For the reference curve, the data points are regenerated each time and with the random error between 0 and 0.06. And we find that the three algorithm can generate CKPCS well and make the AD less than 0.03. Therefore, for this simple curve, the fitness and stability are good for all the three algorithms.

4.1.2. Analysis of the Applicability of Three Algorithms by Complicate Synthetic Data. As $y = \sin(x)$ is a simple and smooth curve, we do not know whether the three algorithms can work well for more complicated curves. To validate the applicability of three algorithms for the complicated curves, different shapes of reference curves are needed to be tested.

Here, some more complex curves are generated using the above introduced method and the applicability of the three algorithms are analyzed. The results are illustrated in Figures 6, 7, and 8. We can see that ALLOpt has a good applicability on simple curves, while it fails to track the complex curves with spiral trajectory or with cross points. MPMopt has a good applicability on simple curves and some complex ones, but

it has some problems on the curve with cross points. DCopt has a good applicability on all the study cases including the curves with spiral trajectory and cross points.

4.2. Application of the Proposed Algorithms in Railway Digital Mapping. As previous studies focus on the synthetic curves and data, we need some real GPS data to test which algorithm can solve the real-world problems best. In this subsection, we will use some real GPS data collected on Qinghai-Tibet Railway to test the performance of the three algorithms in generating CKPCS for railway digital mapping.

The railway GPS data has the characteristics of simple trajectory, long length, and large volume. So, we test the practicability of the proposed algorithms on the railway GPS data by analyzing their approximation error, running time, storage space (vertex number) and other performance indexes.

4.2.1. Data Collection and Preprocessing. The GPS data were collected in one short segment T-04 and one long segment T-10 along Qinghai-Tibet Railway (QTR) as shown in

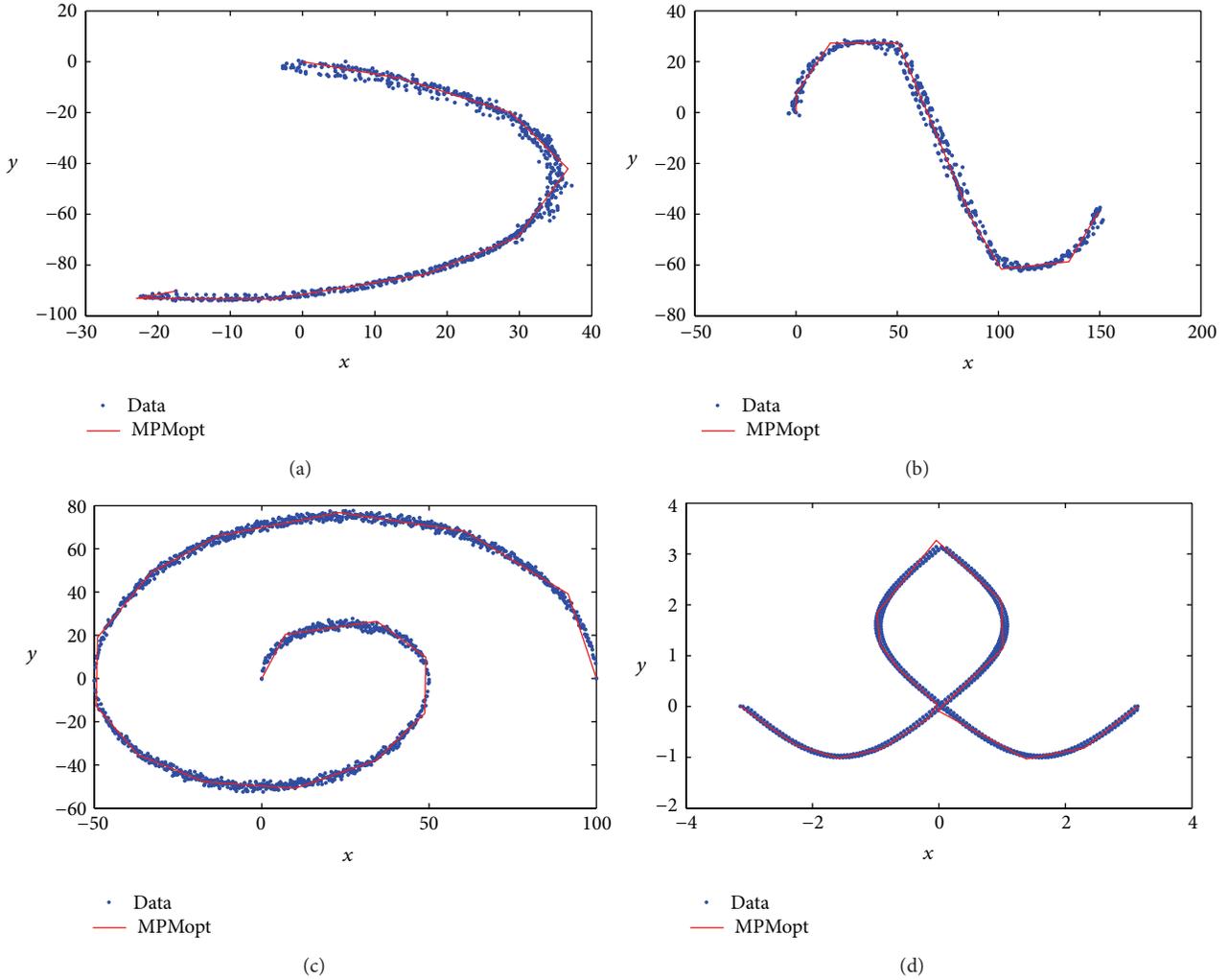


FIGURE 7: Applicability analysis for MPMopt.

Figures 9 and 10. The measurement equipment is Thales z -max with the accuracy of about 6 meters, and the AD for one-time measurement is about 3 m. We collected data three times, hoping to decrease the AD to 1 m.

T-04 is about 2 kilometers long and its two endpoints are two signal machines. Because of the importance of the two signal machines positions, we took about 30 minutes to get the accurate coordinates for each signal machine by averaging the measuring coordinates. Then, the dynamic measurement on trajectory was done three times and 1093 data points were obtained. With the combination of long-time static measurement for two endpoints and short-time dynamic measurements along the railway track, we achieved a decent trade-off between the efficiency and accuracy of the measurements. The T-10 is about 40 kilometers and its two endpoints are two switch turning machines. Two endpoints and 15213 data points for T-10 were obtained using the similar measurement methods for T-04.

4.2.2. Results and Analysis. The setting error for the proposed algorithms is 1 m for T-04 and T-10, which is accurate enough

TABLE 1: Comparisons among the three algorithms on T-04 railway.

	Error (m)	Time (s)	Number
ALLOpt	0.7572	72.8017	9
MPMopt	0.5480	2.6177	8
DCopt	0.6515	18.9546	8

for train positioning. The algorithms will stop if AD is less than 1 m or the decrease of AD is relatively slow; that is, one algorithm will stop if adding a vertex cannot decrease AD by 1 percent.

The CKPCS are obtained from railway GPS data sets collected in QTR using the three proposed algorithms, respectively. For T-04, the obtained CKPCS by the three algorithms are shown in Figures 11, 12, and 13, respectively. Table 1 lists three numerical performance indexes for the three algorithms, where the error means the AD, time represents the running time of the algorithm, and number is the number of vertices in the final CKPCS. We can find that all the three algorithms can generate CKPCS very well but

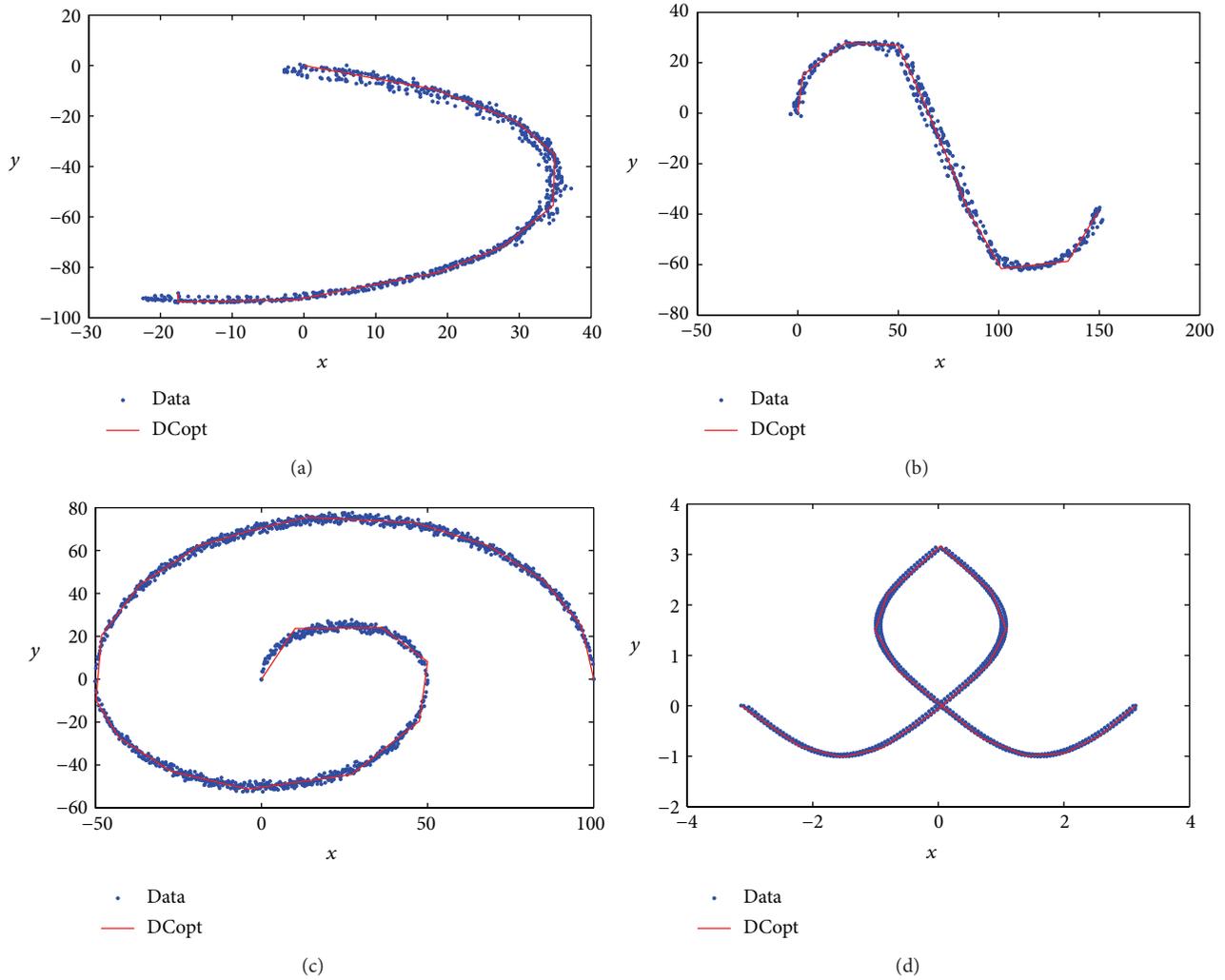


FIGURE 8: Applicability analysis for DCopt.

TABLE 2: Comparison among the three algorithms on T-10 railway.

	Error (m)	Time (s)	Number
ALLOpt	26.912	172832.2	56
MPMopt	0.9971	103.4528	68
DCopt	10.091	4940.191	51

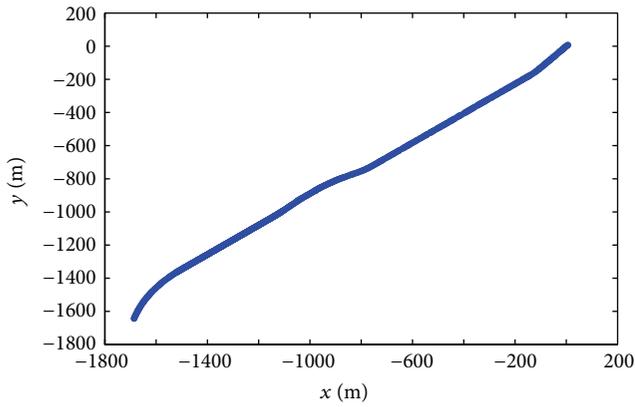
the ALLOpt takes a longer time than DCopt and MPMopt. Moreover, MPMopt algorithm uses the shortest time in generating CKPCS and obtains the minimal AD. As the GPS data set collected in T-04 is a relative small one, the difference among the three algorithms is not distinctive. Next, we will explore the difference of the three algorithms on the large data set collected on T-10.

For T-10 data, the simulation results and the performance indexes are provided in Figures 14, 15, and 16 and Table 2. We can see that the MPMopt algorithm only takes 103 seconds to generate the CKPCS and its accuracy is less than 1 m. But the DCopt takes 82 min with a bigger error of 10 m and ALLOpt takes 24 hours with the biggest error of 27 m. Or

in other words, these two algorithms become invalid on the large data set. MPMopt not only saves time but also greatly reduces storage space for digital maps as it just uses 68 points to represent the 15,213 GPS data points. This example also shows the importance and efficiency of good initial solution and local optimization for generating CKPCS from a huge amount of data points.

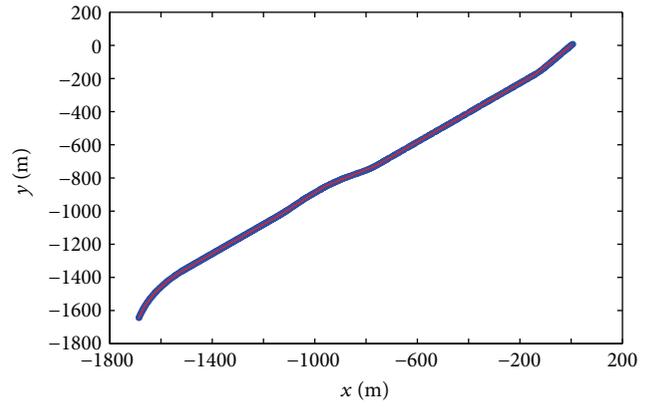
5. Conclusions and Future Work

In this paper, the origination, history, and defects of principal curves are revisited first. For some special problems with fixed two endpoints, CKPCS with complicated computational process was proposed before. Aiming to make the complicated CKPCS algorithm more applicable to real engineering applications especially the railway digital mapping, three CKPCS generation algorithms, named ALLOpt, MPMopt, and DCopt, are proposed and their implementations are explained in detail. All the three algorithms need to keep the two endpoints fixed during the iterative optimization process. There are also some significant differences among them.



• Data

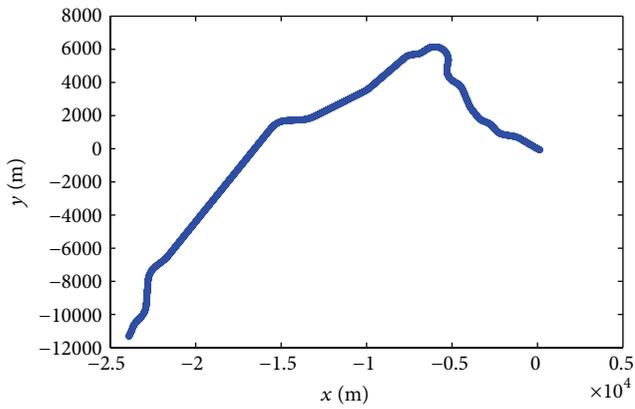
FIGURE 9: T-04 railway GPS data.



• Data

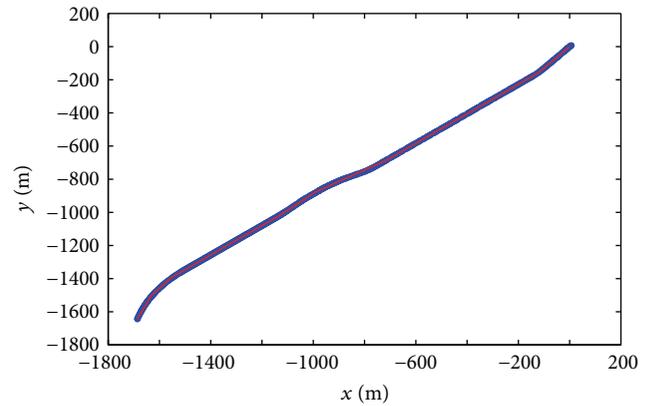
— MPMopt

FIGURE 12: CKPCS obtained by MPMopt for T-04.



• Data

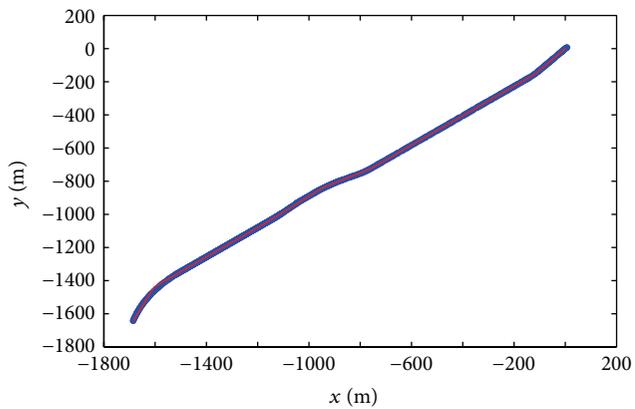
FIGURE 10: T-10 railway GPS data.



• Data

— DCopt

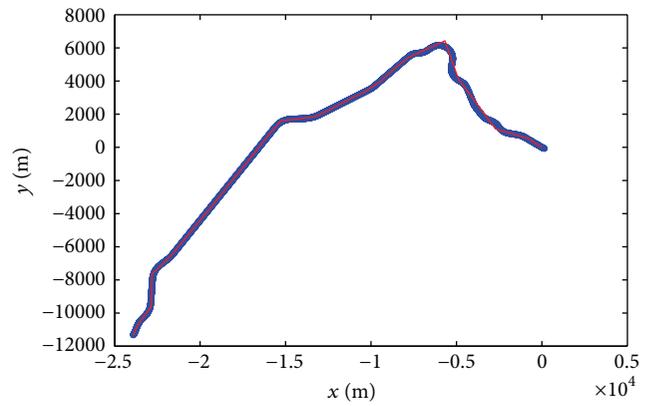
FIGURE 13: CKPCS obtained by DCopt for T-04.



• Data

— ALLOpt

FIGURE 11: CKPCS obtained by ALLOpt for T-04.



• Data

— ALLOpt

FIGURE 14: CKPCS obtained by ALLOpt for T-10.

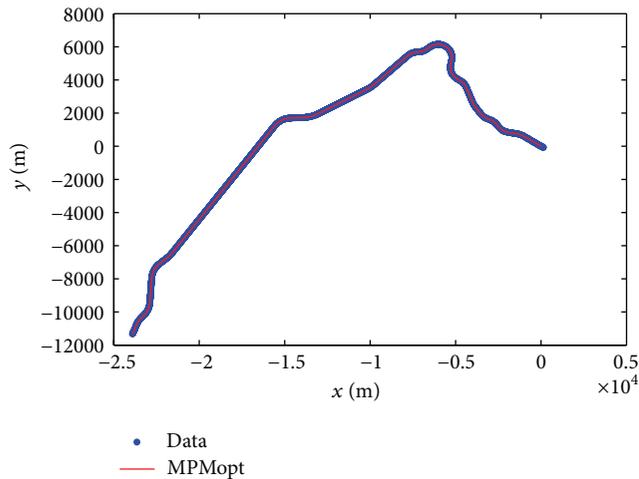


FIGURE 15: CKPCS obtained by MPMopt for T-10.

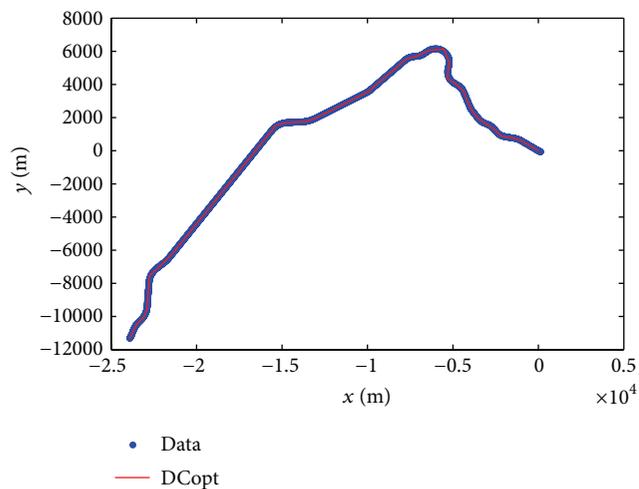


FIGURE 16: CKPCS obtained by DCopt for T-10.

ALlopt use the idea of global optimization, which optimizes the location of all vertices to minimize the average projection distance. However, MPMopt and DCopt employ the idea of the partial optimization, which only heuristically optimize partial vertices to speed up the optimization process. The difference between MPMopt and DCopt lies in the generation of initial solutions, where MPMopt always find the point with the maximal distance while DCopt use the idea of dividing and connecting. In comparison analysis, the average error (AD), the running time, and the number of vertices in the final CKPCS are used to validate the proposed algorithms' performance.

First, verification and analysis of the proposed algorithms are performed on some synthetic data. We found that ALlopt works well only on simple curves and small data sets and takes the longest running time among all the three proposed algorithms. MPMopt can work well both on simple curves and complicated curves, but its performance will get worse on the curves with cross points. DCopt can work well on all curves, but its running time is longer than MPMopt.

Then, two measured GPS data sets collected in Qinghai-Tibet railway are used to test the three algorithms to see which can work for real engineering problems. The results show that the MPMopt algorithm has the best comprehensive performance according to the accuracy, the computational efficiency, and the data reduction capability. As the real track of railway is not very complicated, it is recommended to use MPMopt in CKPCS generation for railway digital map. Of course, ALlopt can be used for small data set of simple curve and DCopt is preferable for very complicate curves with cross points. By using these proposed algorithms, we can produce the CKPCS easily and quickly so as to give a simpler description for complex curve or huge amount of GPS data points; thus, storage space is greatly reduced which can make GPS digital maps work better for real-time GPS positioning.

Although the proposed algorithms obtained good results in synthetic data and railway GPS data, there is further work that can be done, for example, considering more performance indexes like the maximal projection error, the smoothness of the CKPCS, and testing the proposed algorithms with more real engineering data with different shapes. Our methods are not used for generating all kinds of PCs, but used for generating PC with fixed endpoints, which is defined as CKPCS. For CKPCS, the existing PC algorithms with solid theoretical foundation cannot be applied directly. We must admit that we propose our methods with focus more on simple implementation and fast applications. This makes our methods look rather ad-hoc, but suitable for real engineering applications. The experiments on real data collected in Qinghai-Tibet railway also demonstrate that the proposed approaches work very well in real applications. Anyway, the research on practical algorithms with more solid mathematical foundations for CKPCS should not be discouraged and we think it is a good direction of future work.

Acknowledgments

This work is partially supported by the New Scientific Star Program of Beijing under Grant 2010B015, by the Fundamental Research Funds for the Central Universities under Grant 2012JBM016, by the State Key Laboratory of Rail Traffic Control and Safety and Beijing Key Laboratory of Urban Rail Transit Automation and Control under Grant RCS2011ZT001, and by 863 project from Ministry of Science and Technology of China under Grant 2012AA112801.

References

- [1] J.-P. Zhang and J. Wang, "Overview of principal curves," *Chinese Journal of Computer*, vol. 26, no. 2, pp. 129–146, 2003.
- [2] T. J. Hastie, *Principal curves and surfaces [Ph.D. thesis]*, Statistics Department, Stanford University, Stanford, Calif, USA, 1984, Tech. Rep. no. 11, laboratory for Computer Statistics, 1984.
- [3] T. Hastie and W. Stuetzle, "Principal curves," *Journal of the American Statistical Association*, vol. 84, no. 406, pp. 502–516, 1989.
- [4] K. Reinhard and M. Niranjana, "Subspace models for speech transitions using principal curves," *Proceedings of Institute of Acoustics*, vol. 20, no. 6, pp. 53–60, 1998.

- [5] K. Reinhard and M. Niranjani, "Parametric subspace modeling of speech transitions," *Speech Communication*, vol. 27, no. 1, pp. 19–42, 1999.
- [6] T. Hermann, P. Meinicke, and H. Ritter, "Principal curve sonification," in *Proceedings of the International Conference on Auditory Display (ICAD '00)*, pp. 81–86, Atlanta, Ga, USA, 2000.
- [7] D. Dong and T. J. McAvoy, "Nonlinear principal component analysis—based on principal curves and neural networks," *Computers & Chemical Engineering*, vol. 20, no. 1, pp. 65–78, 1996.
- [8] D. J. H. Wilson, G. W. Irwin, and G. Lightbody, "RBF principal manifolds for process monitoring," *IEEE Transactions on Neural Networks*, vol. 10, no. 6, pp. 1424–1434, 1999.
- [9] J. D. Banfield and A. E. Raftery, "Ice floe identification in satellite images using mathematical morphology and clustering about principal curves," *Journal of the American Statistical Association*, vol. 87, no. 417, pp. 7–16, 1992.
- [10] T. Duchamp and W. Stuetzle, "Extremal properties of principal curves in the plane," *The Annals of Statistics*, vol. 24, no. 4, pp. 1511–1520, 1996.
- [11] T. Duchamp and W. Stuetzle, "Geometric properties of principal curves in the plane," in *Robust Statistics, Data Analysis, and Computer Intensive Methods*, vol. 109 of *Lecture Notes in Statistics*, pp. 135–152, Springer, New York, NY, USA, 1996.
- [12] B. Kégl, *Principal curves: learning, design, and applications [Ph.D. thesis]*, Concordia University, Quebec, Canada, 1999.
- [13] B. Kégl and K. Aetal, "A polygonal line algorithm for constructing principal curves," in *Proceedings of the Neural Information Processing Systems (NIPS '99)*, pp. 501–507, Denver, Colo, USA, December 1999.
- [14] B. Kégl and K. Aetal, "Learning and design of principal curves," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 3, pp. 281–297, 2000.
- [15] J. J. Verbeek, N. Vlassis, and B. Kröse, "A k-segments algorithm for finding principal curves," Tech. Rep. IAS-UVA-00-11, Computer Science of Institute, University of Amsterdam, Amsterdam, The Netherlands, 2000.
- [16] U. Ozertem and D. Erdogmus, "Locally defined principal curves and surfaces," *Journal of Machine Learning Research*, vol. 12, no. 4, pp. 1249–1286, 2011.
- [17] Y. Zhang, J. An, and C. Ma, "Fault detection of non-Gaussian processes based on model migration," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 5, pp. 1517–1526, 2013.
- [18] G. Biau and A. Fischer, "Parameter selection for principal curves," *IEEE Transactions on Information Theory*, vol. 58, no. 3, pp. 1924–1939, 2012.
- [19] S. Gerber and R. Whitaker, "Regularization-free principal curve estimation," *Journal of Machine Learning Research*, vol. 14, no. 1, pp. 1285–1302, 2013.
- [20] H. Zhang, W. Pedrycz, D. Miao, and Z. Wei, "From principal curves to granular principal curves," *IEEE Transactions on Cybernetics*, 2013.
- [21] J. Zhang, D. Chen, and U. Kruger, "Adaptive constraint K-segment principal curves for intelligent transportation systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 9, no. 4, pp. 666–677, 2008.
- [22] D. Chen, B. Cai, and T. Tang, "An information fusion algorithm for multiple GPS track data," in *Proceedings of the 4th International Conference on Natural Computation (ICNC '08)*, pp. 468–472, Jinan, China, October 2008.
- [23] G.-G. Gao and B.-G. Cai, "Research on the automatic electronic map generation algorithm for the train supervision system," *Journal of the China Railway Society*, vol. 28, no. 1, pp. 63–67, 2006.
- [24] B.-Q. Guo, T. Tang, and D.-T. Zhou, "Map aided positioning method used in train locating application," *Journal of the China Railway Society*, vol. 29, no. 6, pp. 44–47, 2007.
- [25] T.-S. Lee, C.-C. Chiu, C.-J. Lu, and I.-F. Chen, "Credit scoring using the hybrid neural discriminant technique," *Expert Systems with Applications*, vol. 23, no. 3, pp. 245–254, 2002.
- [26] S.-H. Liao, "Expert system methodologies and applications—a decade review from 1995 to 2004," *Expert Systems with Applications*, vol. 28, no. 1, pp. 93–103, 2005.
- [27] M. Karabatak and M. C. Ince, "An expert system for detection of breast cancer based on association rules and neural network," *Expert Systems with Applications*, vol. 36, no. 2, pp. 3465–3469, 2009.
- [28] J.-B. Jian, J.-L. Li, and X.-D. Mo, "A strongly and superlinearly convergent SQP algorithm for optimization problems with linear complementarity constraints," *Applied Mathematics & Optimization*, vol. 54, no. 1, pp. 17–46, 2006.
- [29] R. H. Byrd, J. C. Gilbert, and J. Nocedal, "A trust region method based on interior point techniques for nonlinear programming," *Mathematical Programming: Series B*, vol. 89, no. 1, pp. 149–185, 2000.
- [30] D. Chen, Y.-S. Fu, B. Cai, and Y.-X. Yuan, "Modeling and algorithms of GPS data reduction for the Qinghai-Tibet Railway," *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, no. 3, pp. 753–758, 2010.
- [31] D. Chen, T. Tang, F. Cao, and B. Cai, "An integrated error-detecting method based on expert knowledge for GPS data points measured in Qinghai-Tibet Railway," *Expert Systems with Applications*, vol. 39, no. 2, pp. 2220–2226, 2012.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

