

TRAJECTORY COMPRESSION WITH CONSTRAINTS OF ROAD NETWORKS

L. Zhang^{1,2,*}, M. Liu^{1,3}, Y. Long^{1,2}

¹ School of Geography Science, Nanjing Normal University, 210023 Nanjing, China – lingzhang.sky@gmail.com,
longyi@njnu.edu.cn

² Key Lab. of Virtual Geographic Environment, Ministry of Education, Nanjing Normal University, 210023 Nanjing, China

³ School of Geographic Information and Tourism, Chuzhou University, 239000 Chuzhou, China - liuminshi1983@126.com

Commission IV, WG IV/7

KEY WORDS: Trajectory Compression, Trajectory Point Ranking, Spatial-Temporal Characteristic, Road Network, Map Matching

ABSTRACT:

In recent, the trajectory data of moving objects is getting bigger and bigger, and it has become a very important part of the social big data. Its compression is an indispensable operation of data processing, and also it is the basis of the data storage, analysis and mining of moving objects. In the related research, there are two kinds of methods for the trajectory compression. One is to compress trajectory data based on its own spatial-temporal characteristics, another kind of methods is the map-matched trajectory compression. However, for offline trajectory compression, methods based on spatial-temporal characteristics do not take the road network constraints into account. If road networks are considered, the map matching is needed first, and it will greatly affect the efficiency of trajectory compression. Therefore, this paper proposes a new trajectory compression algorithm that combines the spatial-temporal characteristics from trajectories themselves and structural characteristics from road networks to improve the compression precision and efficiency.

1. INTRODUCTION

With the rapid development of mobile positioning technologies, the trajectory data of moving objects is getting bigger and bigger, and it has become a very important part of the social big data. The spatial-temporal trajectory of a moving object is a sequence of nodes with position, attribute, and time (Sun et al., 2016). Although it contains a great deal of knowledge, it cannot be used and applied directly as its amount is very huge and accompanied by much noise. In a real environment, many redundant data are recorded for a moving object due to the loss of satellite signals by obstructions from buildings. In addition, a same position is recorded for too many times as the object stops at a place for a long time. So, its compression, which is so called the trajectory compression, is an indispensable operation of data processing, and also it is the basis of the data storage, analysis and mining of moving objects. The trajectory compression can be adopted to provide different scales of the trajectory data for various application fields and environments.

In the related research, there are two kinds of methods for the trajectory compression. One is to compress trajectory data based on its own spatial-temporal characteristics, e.g. synchronous Euclidean distance (Meratnia and de By, 2004), or spatial-temporal 3-dimensional space (Cao et al., 2006; Trajcevski et al., 2006). They are used to improve the normal curve compression methods, such as Douglas-Peucker (DP) algorithm (Douglas and Peucker, 1973), in remaining the spatial-temporal feature and increasing the compression accuracy. In order to control errors (Muckell et al., 2014), such methods usually remove nodes in a trajectory by setting thresholds of distances (position), angles (direction) or rates (time), and can be divided into the offline and online ways (Lee and Krumm, 2011; Potamias et al., 2006). However, the relationship between a

moving object and its geospatial environment is not considered. For examples, a moving object, especially a car, should be constrained by road networks. So, the semantic information cannot be maintained after compression with such methods.

Another kind of methods is the map-matched trajectory compression. Due to constraints of road networks, a trajectory is no longer represented on a 2-dimensional space but the road network space (Kellaris et al., 2013), and it is always matched to the networks first. Then, the trajectory is compressed based on spatial-temporal characteristics (Song et al., 2014), structure optimizations (Sandu Popa et al., 2015) and semantic segmentations (Feng et al., 2013; Liu et al., 2014; Richter et al., 2012). Such methods can make the compressed trajectories more reasonable. However, the matching algorithms have some certain errors and they are complex, especially the global matching methods (Lou et al., 2009). In addition, although a trajectory can be compressed greatly by the method based on semantic segmentations, it has lost its original data form, and lots of information are lost.

Therefore, this paper proposes an improved spatial-temporal trajectory compression method with consideration of road networks. The idea of this method is that a queue is established for all nodes in a trajectory under certain constraints based on the feature points inside the road network. By removing the nodes in the queue, it can achieve the trajectory compression to any scale quickly.

2. METHOD

A spatial-temporal trajectory of a moving object is a sequence of nodes with position, attribute and time, and it can be represented as follows,

* Corresponding author

$$Tra = \{p_i | i \in [1, n]\}, p_i = (x_i, y_i, t_i), \quad (1)$$

where x_i and y_i are position, and t_i is the recorded time.

It can be shown from the equation that a moving trajectory contains spatial position and time information, and it has the spatial and temporal characteristics. In addition, a moving trajectory is an artificial trajectory that shows a matching relationship with a road network.

The main idea of the method is to give a ranking for all trajectory points according to their importance which are restricted by the spatial-temporal characteristics and the road network. It can be proceeded as follows steps.

- Step 1, to rank all trajectory points by a Binary Line Generalization (BLG) tree;
- Step 2, to extract feature points of a road network;
- Step 3, to adjust the ranking according to the feature points as constraints;
- Step 4, to compress trajectories by removing low ranking points.

2.1 Step 1: Trajectory points ranking

The BLG tree (Oosterom, 1991) is a binary tree structure generated when a curve compression is performed using the divide-and-conquer D-P algorithm (threshold is set to 0). In the tree, the root node is the most important feature point, because its distance to the baseline, which is connected by the curve endpoints, is the longest. The distance can be regarded as the eigenvalue of the node. The left and right child nodes of the root node are the important feature points of their subsets respectively, and so on until all points are added to the tree. The details of the BLG tree construction are provided by (Meratnia and de By, 2004), and an example of the tree is illustrated in Figure 1.

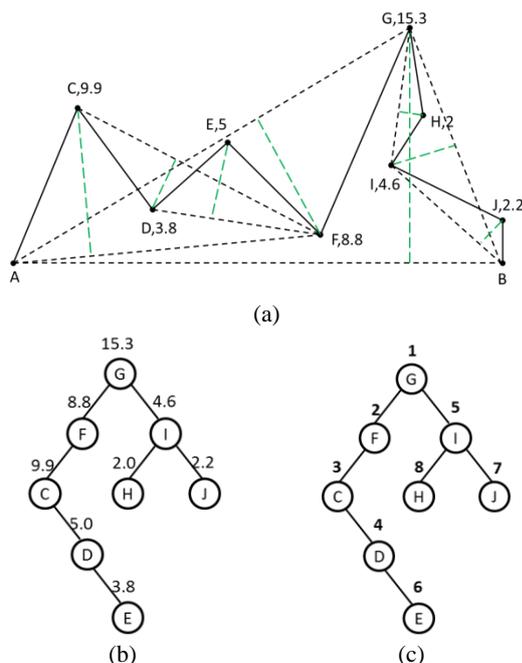


Figure 1. An example of the BLG tree and its ranking

In Figure 1, (a) is a curve example that consist of point sets $\{A, B, C, D, E, F, G, H, I, J, B\}$. A and B are the endpoints, and line AB is the initial baseline. Point G is selected as the root as its distance to the baseline AB is the longest (15.3). Then, points are split into two subsets $\{A, C, D, E, F, G\}$ and $\{G, H, I, J, B\}$, points F and I are the two children of G . By analogy, the tree can be established. Figure 1(b) shows the result of the BLG tree, and the number above the node is the distance between the point and its corresponding baseline.

The BLG tree can represent the eigenvalues of trajectory points and their relations, but the importance rank of trajectory points cannot be determined yet. If it is directly ranked according to the eigenvalues, the "inheritance relationship" among the trajectory points will be destroyed. Taking C and F in Figure 1(b) for example, F is the parent node of C , but the eigenvalue of C (9.9) is larger than F (8.8). If ranking the trajectory points according to the level of the tree first and then the eigenvalue of node in the same level, the "inheritance relationship" is preserved, however, the node with a high eigenvalue but low level will be removed first. As shown in Figure 1(b), the level of I is higher than C , but the eigenvalue of C is far greater than I . Therefore, an improved ranking method needs to be proposed according the eigenvalue with constraints of the "inheritance relationship", and its ranking result is shown in Figure 1(c).

2.2 Step 2: Feature points extraction

In road networks, junctions and corners are always considered as the characteristic points due to its spatial structure. Therefore, in this research, the feature points in a trajectory are the ones that close to the junctions of road networks where the moving object passes by. Figure 2 shows an example.

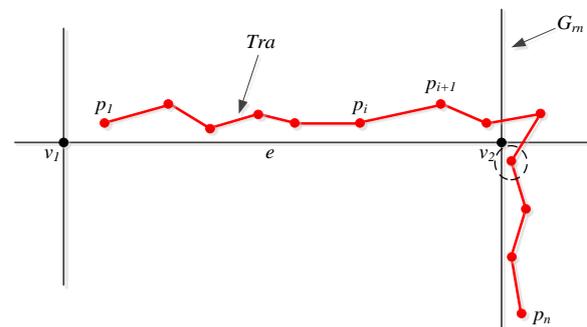


Figure 2. An example of feature points in a trajectory

In Figure 2, Tra in red colour is a trajectory, p_1 and p_n are the endpoints. v_1 and v_2 are junctions. The point in dashed circle is the closest to v_2 , and it is a feature point of Tra .

In the work, all feature points need to be ranked according to their feature value. A feature point in a trajectory is used to show its road network structural characteristics, so its value can be given by its corresponding road junction or corner. It can be calculated by the road level value associated with the junction or corner as,

$$s_v = \sum_{i=1}^n ce_i, \quad (2)$$

where ce is the road level, and there are n roads connecting to the junction or corner. Roads can be classified as express roads, trunk roads, secondary trunk roads and branch roads, and their level values are set to 4, 3, 2, and 1, respectively.

Here, we take all closest trajectory points to road junctions as constraint conditions.

2.3 Step 3: Ranking adjustment

1. To take the feature points of the road network structure extracted from the trajectory as constraint points, and queues the constraint points according to the size of the feature values;
2. To segment the entire trajectory according to the position order of the constraint points. If the number of constraint points is n , the original trajectory is divided into $n-1$ sub-trajectories, and the BLG tree is constructed for each sub-trajectory.
3. To rank all BLG trees of $n-1$ sub-trajectories are collaboratively.

2.4 Step 4: Trajectory points compression

After ranking adjustment, we can remove the corresponding proportion of points from the tail of the queue according to the needs of the compression ratio, so as to achieve fast arbitrary scale compression. The compression ratio is set according to the actual situation. For example, the compression ratio is directly provided, otherwise, if the data scale requirement is provided, the calculation method of the principles of selection (Topfer and Pillewizer, 1966) can be used according to the requirement of the scale transformation, and the compression ratio is derived from the scale conversion ratio.

3. EXPERIMENT AND RESULT

3.1 Experiment design

This research takes 2 real trajectory datasets generated by 2 taxis over a period of 1 week in Beijing city (Figure 3). For the experiment, the proposed compression algorithm is applied to test the datasets, and we also provide an accuracy assessment method.

In the experiment, the results are analysed by comparing the proposed road network constrained compression (RNCC) method with the classical TD-TR (top-down time ratio) algorithm, which is improved from the DP (Douglas-Peucker) algorithm, in both aspects of efficiency and accuracy.

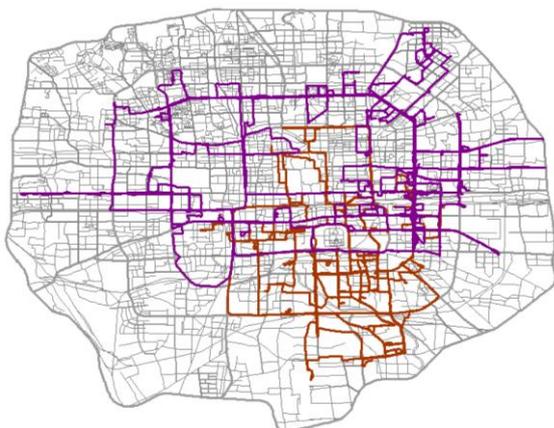


Figure 3. Experimental trajectory data and road networks

3.2 Assessment method

Since the trajectory is usually distributed on the road network, this research proposes the concept of network homomorphic distance error. The network homomorphic distance error is to first match the trajectory to the road network, and use the matched result as the original data, and then calculate the homomorphic distance error between the compressed trajectory data and the matched trajectory data. The calculation formula is as shown in Equation (3), and an illustration is shown in Figure 4.

$$NSD(Trac, Trao) = \frac{1}{n} \sum_{i=1}^n NSD_i \quad (3)$$

where $Trao$ is the original trajectory, $Trac$ is the compressed trajectory, and n is the number of points in $Trac$. NSD_i is the distance between trajectory point i and its homomorphic point in the road network.

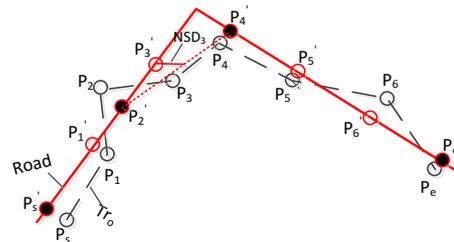
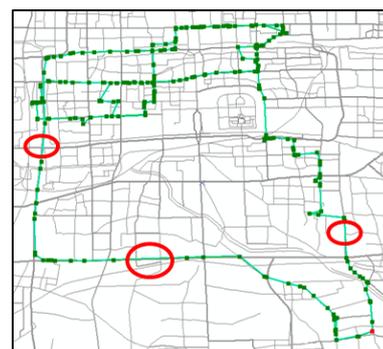


Figure 4. Network homomorphic distance error

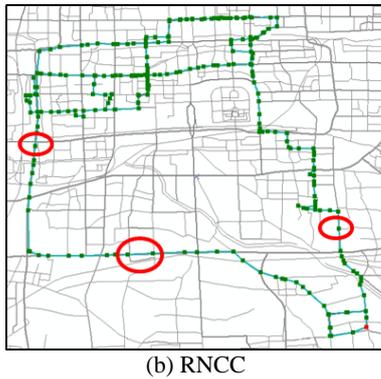
3.3 Experiment result

3.3.1 Compression result

From the compression result, the RNCC method can not only retain the feature points of spatial-time morphological features, but also maintain good maintenance for road network nodes. As shown in Figure 5, they are the 90% compressed results of one trajectory data. (a) shows that the TD-DR method can better maintain the point where the spatial features are prominent (time characteristics are lost). However, the RNCC method preserved not only the points with prominent spatial features, but also the feature points of the trajectories close to the road network. The red circles in Figure 5 are obvious difference.



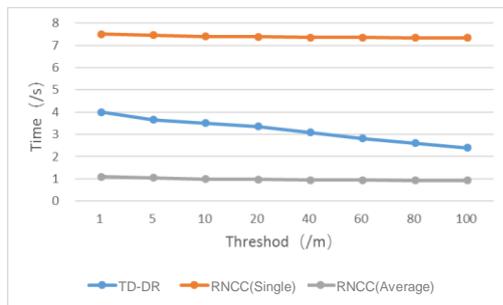
(a) TD-DR



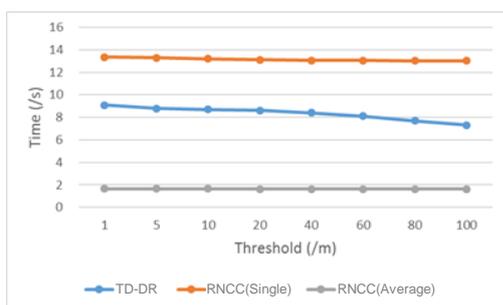
(b) RNCC
 Figure 5. Part compressed results

3.3.2 Efficiency analysis

Since the TD-DR algorithm controls the compression ratio through the threshold, it is difficult to obtain an arbitrary compression ratio result, and the RNCC algorithm can easily obtain the result of any compression ratio. Therefore, the threshold data is used to perform TD-DR compression on the trajectory data, and the corresponding compression ratio is used to perform the RNCC algorithm compression process. After the experiment, eight thresholds such as 1, 5, 10, 20, 40, 60, 80, and 100 m are used for compression, and the compression ratios of the two data are 0.52, 0.69, 0.78, 0.85, 0.91, 0.93, 0.94, 0.95 (trajectory 1) and 0.22, 0.30, 0.37, 0.48, 0.62, 0.71, 0.76, 0.79 (trajectory 2) respectively.



(a) trajectory 1



(b) trajectory 2

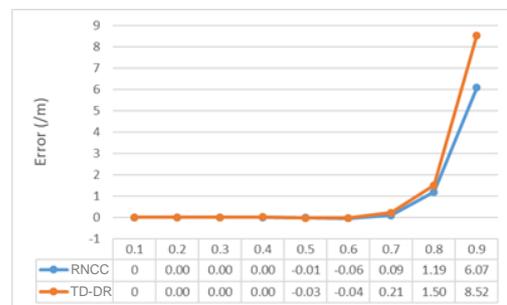
Figure 6. Efficiency comparison

The results are shown in Figure 6. (a) is trajectory 1, and (b) is trajectory 2. The blue line is the running time of the TD-DR algorithm under different thresholds. The orange and grey line are the running time of the RNCC algorithm. Since the running time of the RNCC algorithm is mostly spent on the ranking process, therefore, from a single scale (compression scale)

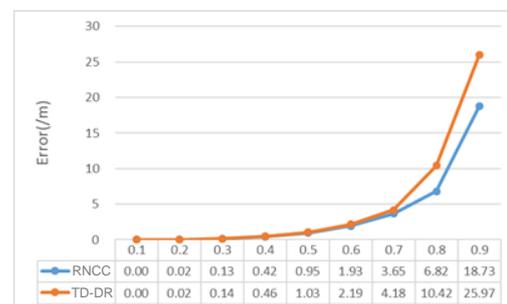
compression, the running time (orange line) of the RNCC algorithm is more than the TD-DR algorithm. However, from multiple scales compression, only one ranking process is required, and its time should be calculated on average. Therefore, its running time is much lower than that of the TD-DR algorithm.

3.3.3 Accuracy analysis

Figure 7 shows the accuracy comparison between the TD-DR algorithm and the RNCC algorithm after the compression of the two data. The accuracy calculation uses the network homomorphic distance calculation method introduced in Section 3.2. At the same time, in order to eliminate the error of the original data, the accuracy is that the network homomorphic error after the trajectory compression is subtracted from the network homomorphic error of the original data. It can be seen from the figure that: (a) the error increases with the increase of the scale, but when the compression ratio is 0.5, 0.6, the error is reduced or even negative, which indicates that the accuracy of the data is higher than the original data; (2) When the compression ratio is small, the accuracy of the two methods are almost the same. When the compression ratio is greater than 0.6, it is obvious that the error of the RNCC algorithm is smaller, especially in trajectory 2.



(a) trajectory 1



(b) trajectory 2

Figure 7. Accuracy comparison

4. CONCLUSION AND RESPECTIVE

From the experiment, there are some conclusions can be addressed:

- Compression results from both algorithms are the same, but the proposed algorithm is more efficient. In addition, it is more suitable for multi-ratio compression once the point ranking is established.
- The accuracy of this algorithm is higher than the TD-DR algorithm. The larger the compression ratio, the higher the accuracy. When the compression ratio is larger than 50%, the result is more significant.

c) The proposed algorithm can preserve both the spatial characteristic and the road structure characteristic of original trajectory at the same time.

This is not a completed work. For the future work, we will compress and simplify the trajectory from the perspective of trajectory semantics. The compression of the current moving trajectory is more focused on spatiotemporal features or road network features, and less on semantic features, such as various staying semantic features of trajectories and landmark semantic features. The purpose of the trajectory compression is not only to reduce the amount of data, but also to extract the features of the trajectory at different scales through the compression and simplification of the trajectory, thereby serving different application scenarios.

ACKNOWLEDGEMENTS

This research was supported by the National Natural Science Foundation of China (41501496, 41271449), and the Key University Natural Science Research Project of Anhui Province (KJ2017A415).

REFERENCES

- Cao, H., Wolfson, O., and Trajcevski, G. (2006). Spatio-temporal data reduction with deterministic error bounds. *The VLDB Journal*, 15(3), 211–228.
- Douglas, D. H., and Peucker, T. K. (1973). Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 10(2), 112–122.
- Feng, S., Xu, J., Xu, M., Zheng, N., and Zhang, X. (2013). EHSTC: an enhanced method for semantic trajectory compression. In *Proceedings of the 4th ACM SIGSPATIAL International Workshop on GeoStreaming - IWGS '13* (pp. 43–49). New York, New York, USA: ACM Press. <https://doi.org/10.1145/2534303.2534306>
- Kellaris, G., Pelekis, N., and Theodoridis, Y. (2013). Map-matched trajectory compression. *Journal of Systems and Software*, 86(6), 1566–1579. <https://doi.org/10.1016/j.jss.2013.01.071>
- Lee, W.-C., and Krumm, J. (2011). *Trajectory Preprocessing*. In *Computing with Spatial Trajectories* (pp. 3–33). New York, NY: Springer New York.
- Liu, K., Li, Y., Dai, J., Shang, S., and Zheng, K. (2014). Compressing large scale urban trajectory data. In *Proceedings of the Fourth International Workshop on Cloud Data and Platforms - CloudDP '14* (pp. 1–6). New York, New York, USA: ACM Press. <https://doi.org/10.1145/2592784.2592787>
- Lou, Y., Zhang, C., Zheng, Y., Xie, X., Wang, W., and Huang, Y. (2009). Map-matching for low-sampling-rate GPS trajectories. In *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems - GIS '09* (p. 352). New York, New York, USA: ACM Press. <https://doi.org/10.1145/1653771.1653820>
- Meratnia, N., and de By, R. A. (2004). Spatiotemporal Compression Techniques for Moving Point Objects. In *Advances in Database Technology - EDBT 2004* (pp. 765–782). Springer, Berlin, Heidelberg.
- Muckell, J., Olsen, P. W., Hwang, J.-H., Lawson, C. T., and Ravi, S. S. (2014). Compression of trajectory data: a comprehensive evaluation and new approach. *GeoInformatica*, 18(3), 435–460.
- Oosterom, P. J. M. van. (1991). The reactive-tree: a storage structure for a seamless, scaleless geographic database. In *Proceedings Tenth International Symposium on Computer-Assisted Cartography, AUTO-CARTO 10* (pp. 393–407). Baltimore, Md, USA.: ACSM/ASPRS.
- Potamias, M., Patroumpas, K., and Sellis, T. (2006). Sampling Trajectory Streams with Spatiotemporal Criteria. In *18th International Conference on Scientific and Statistical Database Management (SSDBM'06)* (pp. 275–284). Vienna, Austria: IEEE.
- Richter, K.-F., Schmid, F., and Laube, P. (2012). Semantic trajectory compression: Representing urban movement in a nutshell. *Journal of Spatial Information Science*, 2012(4), 3–30. <https://doi.org/10.5311/JOSIS.2012.4.62>
- Sandu Popa, I., Zeitouni, K., Oria, V., and Kharrat, A. (2015). Spatio-temporal compression of trajectories in road networks. *GeoInformatica*, 19(1), 117–145. <https://doi.org/10.1007/s10707-014-0208-4>
- Song, R., Sun, W., Zheng, B., and Zheng, Y. (2014). PRESS: a novel framework of trajectory compression in road networks. *Proceedings of the VLDB Endowment*, 7(9), 661–672. <https://doi.org/10.14778/2732939.2732940>
- Sun, P., Xia, S., Yuan, G., and Li, D. (2016). An Overview of Moving Object Trajectory Compression Algorithms. *Mathematical Problems in Engineering*, 2016, 1–13.
- Topfer, F., and Pillewizer, W. (1966). The Principles of Selection. *Cartographic Journal*, 3(1), 10–16.
- Trajcevski, G., Cao, H., Scheuermann, P., Wolfson, O., and Vaccaro, D. (2006). On-line data reduction and the quality of history in moving objects databases. In *Proceedings of the 5th ACM international workshop on Data engineering for wireless and mobile access - MobiDE '06* (p. 19). New York, New York, USA: ACM Press.