Journal of Information & Computational Science 9: 2 (2012) 257–265
Available at http://www.joics.com

Fast but Not Bad Initial Configuration for Metric Multidimensional Scaling

Shuo Xu [a], Xiaodong Qiao [a], Lijun Zhu [a], Yunliang Zhang [a], Lin Li [b,*]

[a] *Information Technology Supporting Center, Institute of Scientific and Technical Information of China, Beijing 100038, China*

[b] *College of Information and Electrical Engineering, China Agricultural University Beijing 100083, China*

Abstract

The Multidimensional Scaling (MDS) has become a standard technique in multivariate data analysis and is widely used in a variety of disciplines. The objective of MDS is to find a configuration matrix so that given pairwise dissimilarities can be preserved as faithfully as possible. But since there exist a lot of non-global minima for (s)stress, many optimization iterative algorithms are liable to converge to local minima. Thus, the choice of a good initial configuration is crucial. Through closer examination on currently used initial configurations, we find that all these configurations require extensive preprocessing and usually are computationally expensive, thus not appropriate for large scale applications. To overcome this problem, we conjecture that several approximating scalable solutions for classical MDS can be used to initialize the configuration for metric MDS at lower complexity, but should have comparative performance with classical MDS. Finally, extensive simulation experimental results verify our assumptions.

Keywords: Multidimensional Scaling (MDS); Classical MDS; (s)stress; FastMap Algorithm; Initial Configuration

1 Introduction

The term *Multidimensional Scaling* (MDS) [1, 2] refers to a family of techniques that construct a configuration of points in a target metric space from information about inter-point dissimilarities, measured in some other metric space. Originally developed by psychometricians, MDS has become a standard technique in multivariate data analysis and is widely used in a variety of disciplines, such as Wireless Sensor Network (WSN) location [3], computational chemistry [4], Knowledge Organization System (KOS) [5], and author co-citation analysis [6], etc. Depending on the meaning of the inter-point dissimilarity, MDS techniques can be further divided into two categories: metric and non-metric. This paper mainly puts focus on metric MDS. For more elaborate and detailed surveys we refer the readers to [1, 2].

*Corresponding author.
Email addresses: xush@istic.ac.cn (Shuo Xu), qiaox@istic.ac.cn (Xiaodong Qiao), zhulj@istic.ac.cn (Lijun Zhu), zhangyl@istic.ac.cn (Yunliang Zhang), lilincau@gmail.com (Lin Li).

February 2012

MDS takes as input an $n \times n$ matrix $\mathbf{\Delta} = [\delta_{i,j}]$ containing pairwise dissimilarities between all n objects. A valid dissimilarity matrix $\mathbf{\Delta}$ must satisfy: (i) non-negative, $\delta_{i,j} \geq 0$ for $\forall (i,j) \in \mathbb{N}_n \times \mathbb{N}_n$, where $\mathbb{N}_n = \{1, 2, \ldots, n\}$; (ii) self-similarity, $\delta_{i,i} = 0$ for $\forall i \in \mathbb{N}_n$ and (iii) symmetry, $\delta_{i,j} = \delta_{j,i}$ for $\forall (i,j) \in \mathbb{N}_n \times \mathbb{N}_n$. A dissimilarity matrix $\mathbf{\Delta}$ is metric if (i)-(iii) hold and $\mathbf{\Delta}$ obeys the triangle inequality: $\delta_{i,j} \leq \delta_{i,k} + \delta_{k,j}$ for $\forall (i,j,k) \in \mathbb{N}_n \times \mathbb{N}_n \times \mathbb{N}_n$. It is worth mentioning that not all metric dissimilarity matrices $\mathbf{\Delta}$ are necessarily Euclidean. By the Euclidean matrix, we means that one can utilize $\mathbf{\Delta}$ to embed the n objects in a Euclidean space in some way such that the Euclidean distance matrix between all n points is equal to $\mathbf{\Delta}$ exactly. Gower and Legendre [7, 8] gave a nice example to illustrate this point.

The objective of MDS is to find corresponding configuration $\mathbf{x}_i \in \mathbb{R}^p$ for each object $i \in \mathbb{N}_n$ so that given pairwise dissimilarities can be preserved as faithfully as possible. That is to say, to find a $n \times p$ *configuration matrix* $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n]^{\mathrm{T}}$ so that the distance matrix $\mathbf{D}(\mathbf{X}) = [d_{i,j}(\mathbf{X})]$ ($\mathbf{D} = [d_{i,j}]$ for short) approximates the dissimilarity matrix $\mathbf{\Delta}$ as possible as one can. Two commonly used criteria for measuring the discrepancy between $\mathbf{D}$ and $\mathbf{\Delta}$ are the raw stress and sstress functions, given by

$$\rho_r(\mathbf{X}, \mathbf{\Delta}, \mathbf{W}) = \sum_{i<j} w_{i,j}[(d_{i,j}(\mathbf{X}))^r - \delta_{i,j}^r]^2 \equiv \sum_{i<j} w_{i,j}[d_{i,j}^r - \delta_{i,j}^r]^2 \tag{1}$$

Here, the nonnegative weights matrix $\mathbf{W} = [w_{i,j}]$ can be used to give more importance to some dissimilarities, or to accommodate missing entities or to exclude some entities from the stress ($\rho_1(\cdot,\cdot,\cdot)$) or sstress ($\rho_2(\cdot,\cdot,\cdot)$) calculation by setting the appropriate $w_{i,j} = 0$. In MDS literature, the sstress function is slight less popular than the stress function, which is the criterion that we consider in this paper.

Since the stress are non-linear non-convex functions w.r.t. $\mathbf{X}$, minima of $\rho_1(\cdot,\cdot,\cdot)$ are usually calculated by some iterative algorithm. A survey of related algorithms for minimizing $\rho_1(\cdot,\cdot,\cdot)$ was made by [9, 10]. Many empirical and theoretical evidences (such as [11, 12], just to name a few) indicate that there exist a lot of non-global minima for (1) and many optimization algorithms are liable to converge to local minima. To improve the chance that the algorithm will converge to a global minimum, using a good initial configuration is therefore important. A popular choice of initial configuration is the configuration obtained by *classical MDS* [13]. Malone *et al.* [14] exploited results from the theory of distance matrices to derive two alternatives, which guaranteed to be at least as good as the classical MDS.

However, all these methods require extensive preprocessing and usually are computationally expensive, thus not appropriate for large scale applications. Since 1995, several approximating scalable solutions for classical MDS have been proposed, such as FastMap [15], Landmark MDS [16, 17], MetricMap [18] and so on. FastMap determines one coordinate at a time, and is exactly an iterated form of Landmark MDS in the simplest case of 2 landmarks [16]. MetricMap attempts to do the entire projection at once. Furthermore, Platt [19] showed that all three algorithms are based on the Nyström approximation of the eigenvectors and eigenvalues of a matrix.

We conjecture that any of the solutions obtained by these three algorithms can be used to initialize the configuration for metric MDS at lower complexity, but should have comparative performance with classical MDS in terms of average distance error (see Section 4). To validate this assumption, FastMap is adopted as an illustrative example. But the discussions here should be equally applicable for MetricMap and Landmark MDS. The organization of the rest of this paper is as follows. The classical MDS and FastMap algorithm are briefly described in Section 2

& 3, respectively. In Section 4 extensive simulation experimental evaluations are conducted and discussed, and Section 5 concludes this paper.

2 Classical MDS

The procedure for classical MDS is summarized in the following steps.

1. Convert the input dissimilarity matrix $\mathbf{\Delta}$ into a matrix of dot products, or a Gram matrix $\mathbf{B}$. This is done by multiplying the matrix of squared dissimilarity $\mathbf{\Delta}^2$ on both sides with a double centering matrix $\mathbf{H}$, which in fact subtracts out the row and column average of each entry and adds back the overall matrix average. Formally,

$$\mathbf{B} = -\frac{1}{2}\mathbf{H}\mathbf{\Delta}^2\mathbf{H} \tag{2}$$

   where $\mathbf{H} = \mathbf{I} - (\frac{1}{n})\mathbf{e}\mathbf{e}^{\mathrm{T}}$ is the mean-centering matrix, $\mathbf{I}$ is the identity matrix and $\mathbf{e}$ is an all-one column vector.

2. Since $\mathbf{B}$ is symmetric, it can be eigendecomposed into $\mathbf{B} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^{\mathrm{T}}$, where $\mathbf{\Lambda}$ is a diagonal matrix containing the eigenvalues and $\mathbf{V} = (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n)$ whose columns are the corresponding eigenvectors.

3. Let the matrix of the first p eigenvalues greater than zero be $\mathbf{\Lambda}_+$ and $\mathbf{V}_+$ the first p columns of $\mathbf{V}$. Then, the configuration of classical MDS is given by $\mathbf{X} = \mathbf{V}_+\mathbf{\Lambda}_+^{\frac{1}{2}}$.

It is easy to see that the Step 2 is the bottleneck of classical MDS. A complete eigendecomposition of $\mathbf{B}$ using QR decomposition takes $\mathcal{O}(n^3)$ time, resulting in an $\mathcal{O}(n^3)$ time complexity for classical MDS. However, a nice property of classical MDS is that the dimensions are nested. This means that, for example, the first $k-1$ dimensions of a k-dimensional classical MDS solution are the same as the $k-1$ dimensions of a $(k-1)$-dimensional classical MDS solution.

Additionally, due to the preprocessing procedure in Step 1, classical MDS is not equivalent to nonlinear least-squares parameter fitting using the original measurements. In fact, classical MDS can be defined by the following optimization problem, which is implicit in Torgerson's [13] and Gower's [20] pioneering studies of metric MDS:

$$\begin{aligned}\min_{\mathbf{X}\in\mathbb{R}^{n\times p}} \mathcal{L}(\mathbf{X}) &= \| -\frac{1}{2}\mathbf{H}[\mathbf{D}^2(\mathbf{X}) - \mathbf{\Delta}^2]\mathbf{H} \|^2 \\ &= \| \mathbf{X}\mathbf{X}^{\mathrm{T}} + \frac{1}{2}\mathbf{H}\mathbf{\Delta}^2\mathbf{H} \|^2 \\ &= \| \mathbf{X}\mathbf{X}^{\mathrm{T}} - \mathbf{B} \|^2 \end{aligned} \tag{3}$$

Here, $\mathcal{L}(\mathbf{X})$ is sometime called *strain*. Gower [20] proved that choosing the solution of classical MDS solves (3).

3 FastMap Algorithm

The FastMap algorithm, which was proposed by Falutsos and Lin [15], is considered as a distance mapping algorithm or dimension reduction method. The procedure for FastMap algorithm is summarized in the following steps.

1. Initialization: Let $\mathbf{D} \leftarrow \mathbf{\Delta}$, $k \leftarrow 1$.

2. Choose two pivot objects (O_a and O_b) to form the project line in the next step. The two pivot objects should be selected such that the distance ($d_{a,b}$) between O_a and O_b is maximized. In order to keep the time complexity of the FastMap algorithm linear, Falusos and Lin [15] proposed a linear heuristic algorithm. The middle two steps in the heuristic algorithm can be repeated a constant number of times, denoted by r. In our experiments, we let $r = 5$.

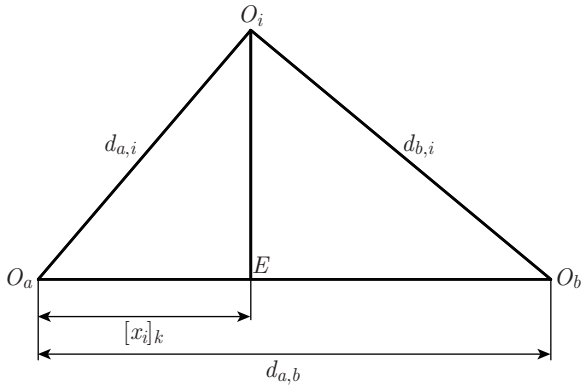

Fig. 1: Illustration of the cosine law – projection on the line O_a – O_b

3. Project all other objects, O_i for $\forall i \in \mathbb{N}_n - \{a, b\}$, on the project line O_a – O_b to find the k-th coordinate $[\mathbf{x}_i]_k$ by utilizing the cosine law. Fig. 1 illustrates the projection of O_i onto the line O_a – O_b. The k-th coordinate $[\mathbf{x}_i]_k$ can be obtained by using Pythagorean Theorem for the rectangles $\triangle O_i E O_a$ and $\triangle O_i E O_b$ and is given formally by

$$[\mathbf{x}_i]_k = \frac{d_{a,i}^2 + d_{a,b}^2 - d_{b,i}^2}{2d_{a,b}}, \text{for } \forall i \in \mathbb{N}_n \tag{4}$$

4. Update the distance matrix $\mathbf{D}$. To make this easier to follow, consider an imaginary hyperplane H perpendicular to the line O_a – O_b, and then project all the objects onto H. For any two distinct objects O_i and O_j, let the resulting projections on H be O_i' and O_j', respectively. Then it can be shown that the distance between O_i' and O_j' on H is given by [15]:

$$d_{i,j}'^2 = d_{i,j}^2 - ([\mathbf{x}_i]_k - [\mathbf{x}_j]_k)^2, \text{for } \forall (i, j) \in \mathbb{N}_n \times \mathbb{N}_n \tag{5}$$

   And let $\mathbf{D} \leftarrow \mathbf{D}' = [d_{i,j}']$.

5. Let $k \leftarrow k + 1$, and then repeat Step 2-4 until $k = p$.

The main advantage of the FastMap algorithm is its low computational complexity. Specifically, the time complexity is $\mathcal{O}(p \times n)$. However, the FastMap algorithm is sensitive to outliers and coordinate alignment [21, 22].

4 Experiments and Discussions

To verify our assumption in Section 1 and illustrate the noise-sensitivity of metric MDS with different initial configurations, $n \in \{500, 1000, 1500, 2000, 2500, 3000, 3500, 4000, 4500, 5000\}$ points are randomly independently generated from the uniform distribution within a square of side 1 (unit distance) and center at (0.5, 0.5) subjected to varying degree of noise. To add noise to the data, the Euclidean distance matrix was multiplied entry-wise by independent random variables distributed as $\exp(N(0, \sigma))$ and symmetrized. The noise levels $\nu\%$ referred to in the Fig. 2 & 3 are defined by $\nu = 100 \times (\exp(\sigma) - 1)$. In this study, we set $\nu \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$, where $\nu = 0$ corresponds to no noise level. This modified distance matrix is then passed to metric MDS.

The quality of the algorithms is measured in two ways: CPU time, and how much average distance error ε is introduced by the embedding:

$$\varepsilon = \frac{1}{n}\sqrt{\sum_{i=1}^{n}\sum_{i=1}^{p}([\mathbf{x}_i]_k - [\hat{\mathbf{x}}_i]_k)^2} \tag{6}$$

where $\mathbf{x}_i$ is the true (unembedded) coordinate and $\hat{\mathbf{x}}_i$ is the estimated one for each object $i \in \mathbb{N}_n$. For each noise level, the experiments are repeated 5 times. And the values in Fig. 2 & 3 correspond to average of 5 different simulations. For nonnegative weights matrix $\mathbf{W} = [w_{i,j}]$, we set each $w_{i,j} = 1$. We have implemented all related algorithms in MATLAB R2006a on an IBM 3850 M2. The corresponding toolbox can be available from the authors upon request for academic use. Since an MDS solution is unique down to a rigid-body transformation, with a possible reflection, the procrustes procedure [1] is utilized when necessary.

From Fig. 2, it is not difficult to see that the FastMap algorithm has comparative performance with classical MDS in terms of average distance error (ε). Moreover, the performance of both FastMap algorithm and classical MDS gradually decreases as the noise level ($\nu\%$) increases, which indicates that both of them are sensitive to noise. However, in terms of CPU time, the performance of two algorithms has a big gap from Fig. 3. When the number of considered objects, n, is small, such as $n = 500, 1000, 1500$, classical MDS has a certain advantage over FastMap algorithm. But when $n \geq 2000$, FastMap algorithm outperforms obviously classical MDS. This indicates that FastMap algorithm has better scalability than classical MDS, thus appropriate for large scale embedding applications. All these verify our assumptions in Section 1.

5 Conclusions

The Multidimensional Scaling (MDS) has become a standard technique in multivariate data analysis and is widely used in a variety of disciplines. The objective of MDS is to find a configuration matrix so that given pairwise dissimilarities can be preserved as faithfully as possible. The raw (s)stress functions are commonly used criteria for measuring the discrepancy. However, the (s)stress are non-linear non-convex functions w.r.t. the configuration matrix, so minima of (s)stress are usually calculated by some iterative algorithms. But since there exist a lot of non-global minima for (s)stress, many optimization algorithms are liable to converge to local minima. Therefore, the choice of an initial configuration from which to begin searching for an optimal configuration is crucial.

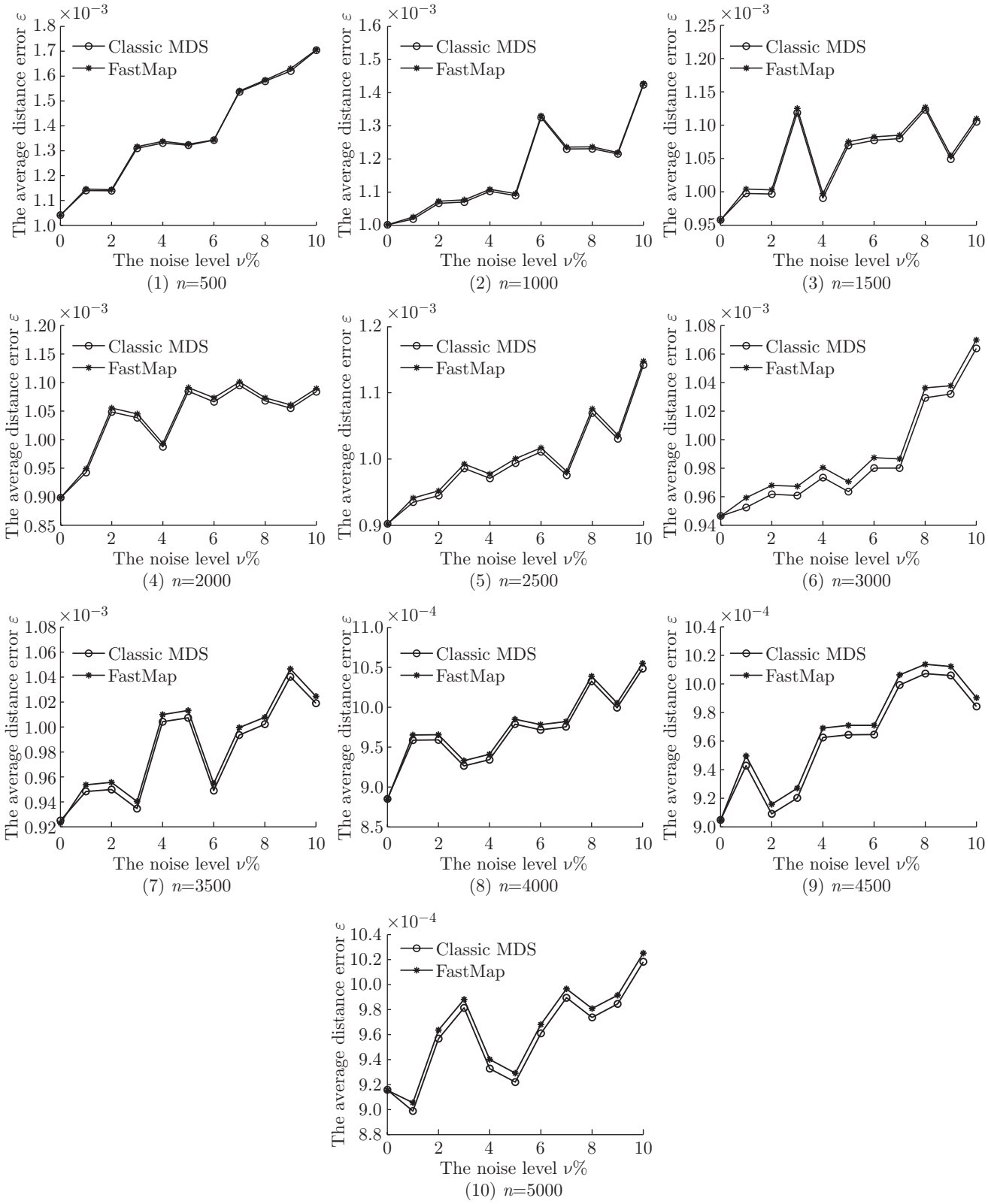

Fig. 2: The average distance error ε (lower is better)

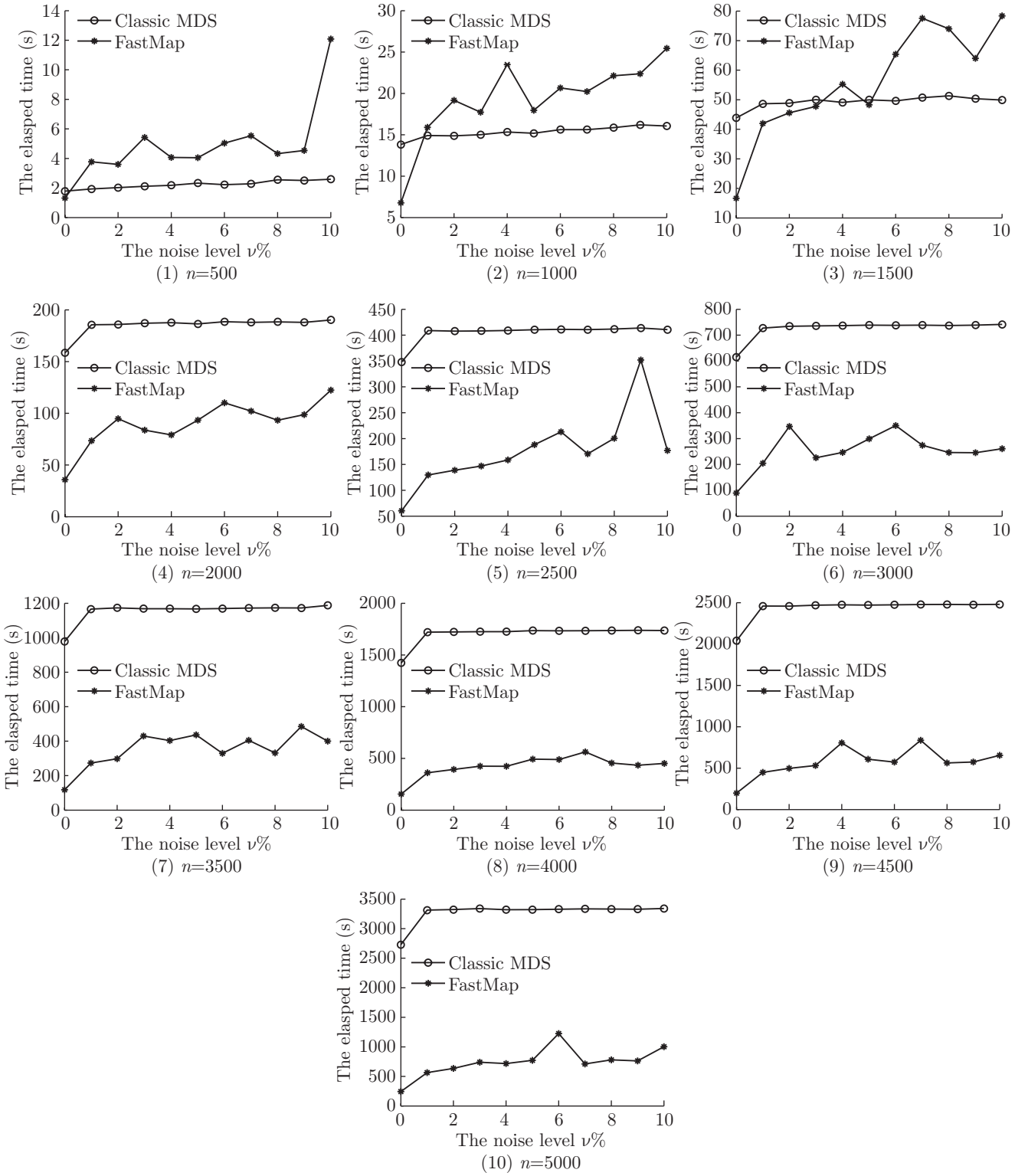

Fig. 3: The CPU time (in seconds) (lower is better)

Many approaches for good initial configurations are proposed, such as classical MDS, two ones raised by Malone *et al.* [14] and so on. On closer examination, we find that all these methods require extensive preprocessing and usually are computationally expensive, thus not appropriate for large scale applications. To overcome this problem, we conjecture that several approximating

scalable solutions for classical MDS, such as FastMap [15], Landmark MDS [16, 17], MetricMap [18], etc., can be used to initialize the configuration for metric MDS at lower complexity, but should have comparative performance with classical MDS in terms of average distance error. Finally, extensive simulation experimental results verify our assumptions.

Acknowledgment

This work was funded partially by the "Key Technologies Research on Large Scale Semantic Computation for Foreign Scientific & Technical Knowledge Organization System", which is sponsored by Key Technologies R&D Program of Chinese 12th Five-Year Plan (2011-2013) under grant number 2011BAH10B04, "Special Funds for Agro-Scientific Research in Public Interest in P. R. China" under grant 200903021, The Predictive Project of Institute of Scientific and Technical Information of China (ISTIC): Monitoring Topics and Recovering Evolution of Topics in Scientific & Technological Literature under grant YY-201129, National Natural Science Foundation: Multilingual Documents Clustering Based on Comparable Corpus under grant 70903032, Social Science Fundation of Jiangsu Province: Study on Automatic Indexing of Digital Newpapers under grant 09TQC011 and MOE Project of Humanitites and Social Sciences: Research on Further Processing of e-Newspaper under grant 09YJC870014.

References

[1] I. Borg, P. J. F. Groenen, Modern Multidimenional Scaling, 2nd Ed., Springer-Verlag, New York, 2005

[2] T. Cox, M. Cox, Multidimensional Scaling, 2nd Ed., Chapman & Hall, 2001

[3] Z. M. Pei, Z. D. Deng, S. Xu, X. Xu. Anchor-free localization method for mobile targets in coal mine wireless sensor networks, Sensor, 9 (2009), 2836-2850

[4] D. K. Argafiotis, D. N. Rassokhin, V. S. Lobanov, Multidimensional scaling and visualization of large molecular similarity tables, Journal of Computational Chemistry, 22 (2001), 488-500

[5] X. Xu, X. D. Qiao, L. J. Zhu, H. E. Guo, An approach to construct vector space models only from proximity data, Journal of the China Society of Scientific and Technical Information, 2011 (in Chinese, To appear)

[6] S. Pushpa, K. S. Easwarakumar, S. Elias, Z. Maamar, Towards a time-based approach for author co-citation analysis, Journal of Information & Computational Science, 8 (2011), 262-279

[7] J. C. Gower, P. Legendre, Metric and euclidean properties of dimensionality coefficients, Journal of Classification, 3 (1986), 5-48

[8] B. S. Everitt, S. Rabe-Hesketh, The Analysis of Proximity Data, Arnold, London, 1997

[9] A. J. Kearsley, R. A. Tapia, M. W. Trosset, The solution of the metric STRESS and SSTRESS problems in multidimensional scaling using Newton's method, Computational Statistics, 13 (1998), 369-396

[10] M. M. Bronstein, A. M. Bronstein, R. Kimmel, I. Yavneh, Multigrid multidimensional scaling, Numerical Linear Algebra with Applications, 13 (2006), 149-171

[11] P. J. F. Groenen, W. J. Heiser, The tunneling method for global optimization in multidimensional scaling, Psychometrika, 61 (1996), 529-550

[12] M. W. Trosset, R. Mathar, On the existence of nonglobal minimizers of the stress criterion for metric multidimensional scaling, Proceeding of the Statistical Computing Section, American Statistical Association, Alexandria, VA, 1997, 158-162

[13] W. S. Torgerson, Multidimensional scaling: I. theory and method, Psychometrika, 17 (1952), 401-419

[14] S. W. Malone, P. Tarazaga, M. W. Trosset, Better initial configurations for metric multidimensional scaling, Computational Statistics & Data Analysis, 41 (2002), 143-156

[15] C. Faloutsos, K. I. Lin, FastMap: A fast algorithm for indexing, data-mining and visualization of traditional and multimedia datasets, Proceedings of the ACM SIGMOD International Conference on Management of Data, M. J. Carey, D. A. Schneider, editors, San Jose, California, 163-174

[16] V. de Silva, J. B. Tenenbaum, Sparse Multidimentional Scaling Using Landmark Points, Technical Report, Stanford University, 2004

[17] V. de Silva, J. B. Tenenbaum, Global versus local methods in nonlinear dimensionality reduction, Advances in Neural Information Processing System 15, S. Becker, S. Thrun, K. Obermayer, editors, MIT Press, Cambridge, MA, 2003, 705-712

[18] J. T. L. Wang, X. Wang, D. Shasha, K. Zhang, MetricMap: An embedding technique for processing distance-based queries in metric spaces, IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics, 35 (2005), 973-987

[19] J. C. Platt, FastMap, MetricMap, and Landmark MDS are all Nyström algorithms, Proceedings of the 10th International Workshop on Artificial Intelligence and Statistics, 2005, 325-338

[20] J. C. Gower, Some distance properties of latent root and vector methods used in multivariate analysis, Biometrika, 53 (1966), 325-338

[21] G. Latsoudas, N. D. Sidiropoulos, A fast and effective multidimensional scaling approach for node localization in wireless sensor networks, IEEE Transactions on Signal Processing, 55 (2007), 5121-5127

[22] G. Ostrouchov, N. F. Samatova, On FastMap and the convex hull of multivariate data: Toward fast and robust dimension reduction, IEEE Transactions on Pattern Analysis and Machine Intelligence, 27 (2005), 1340-1343