Available online at www.sciencedirect.com
SCIENCE DIRECT®
Journal of Molecular Graphics and Modelling 22 (2003) 133–140
Journal of Molecular Graphics and Modelling
www.elsevier.com/locate/JMGM

A modified update rule for stochastic proximity embedding

Dmitrii N. Rassokhin*, Dimitris K. Agrafiotis

3-Dimensional Pharmaceuticals Inc., 665 Stockton Drive, Exton, PA 19341, USA

Received 18 December 2002; received in revised form 6 June 2003; accepted 6 June 2003

Abstract

Recently, we described a fast self-organizing algorithm for embedding a set of objects into a low-dimensional Euclidean space in a way that preserves the intrinsic dimensionality and metric structure of the data [Proc. Natl. Acad. Sci. U.S.A. 99 (2002) 15869–15872]. The method, called stochastic proximity embedding (SPE), attempts to preserve the geodesic distances between the embedded objects, and scales linearly with the size of the data set. SPE starts with an initial configuration, and iteratively refines it by repeatedly selecting pairs of objects at random, and adjusting their coordinates so that their distances on the map match more closely their respective proximities. Here, we describe an alternative update rule that drastically reduces the number of calls to the random number generator and thus improves the efficiency of the algorithm.

Index terms: Stochastic proximity embedding; Multidimensional scaling; Non-linear mapping; Sammon mapping; Non-linear manifold; Manifold learning; Dimensionality reduction; Data mining; Conformational analysis; Combinatorial chemistry; Molecular similarity; Molecular diversity; QSAR

Keywords: Stochastic proximity embedding; Multidimensional scaling; Nonlinear mapping; Sammon mapping; Principal component analysis

1. Introduction

Extracting knowledge from large volumes of data is a prevalent theme in modern scientific research. The problem is particularly relevant in the chemical and biological sciences, where technologies such as combinatorial chemistry, high-throughput screening and expression profiling are routinely employed in order to identify potential therapeutic targets, decipher biochemical pathways, engineer proteins, and design new drugs.

The major difficulty posed by large data sets is the fact that the variables that determine the behavior of a system are either not directly observable or are hidden under a pile of redundancies. The classical methods for removing excessive variables and extracting a meaningful low-dimensional structure are principal component analysis (PCA) [1] and multidimensional scaling (MDS) [2]. The former reduces a set of partially cross-correlated data into a smaller set of orthogonal variables with minimal loss in the contribution to variation, whereas the latter produces an embedding that preserves the interpoint distances. Although these methods work well with linear or quasi-linear subspaces, they fail to detect non-linear structures, curved manifolds, and arbitrarily shaped clusters.

The primary failure of MDS lies in the fact that it tries to preserve all pairwise distances in the data sample, both local and remote. Indeed, given a set of N objects, a symmetric matrix, r_{ij}, of relationships (proximities) between these objects, and a set of images on a D-dimensional display map $\{\boldsymbol{x}_i, i = 1, 2, \ldots, N; \boldsymbol{x}_i \in R^D\}$, MDS attempts to place $\boldsymbol{x}_i$ onto the plane in such a way that their Euclidean distances $d_{ij} = ||\boldsymbol{x}_i - \boldsymbol{x}_j||$ approximate as closely as possible the corresponding values r_{ij}. This is typically accomplished by minimizing an error function that measures the discrepancy between the input and output distances, such as Kruskal's stress

$$S = \sqrt{\frac{\sum_{i<j}(d_{ij} - r_{ij})^2}{\sum_{i<j} d_{ij}^2}}$$

However, it has been known for a long time that conventional similarity measures such as the Euclidean distance tend to underestimate the proximity of points on a non-linear manifold, and lead to erroneous embeddings [3,4]. Sammon's non-linear mapping (NLM) algorithm [5] partly alleviates this problem by introducing a normalization factor in the error function to give increasing weight to

* Corresponding author. Tel.: +1-610-458-5264x6546; fax: +1-610-458-8249.
E-mail address: dima.rassokhin@3dp.com (D.N. Rassokhin).

1093-3263/$ – see front matter © 2003 Elsevier Inc. All rights reserved.
doi:10.1016/S1093-3263(03)00155-4

short range distances over long range ones

$$S = \frac{\sum_{i<j}((d_{ij} - r_{ij})^2 / r_{ij})}{\sum_{i<j} r_{ij}}.$$

This scheme, however, is arbitrary and fails with highly folded topologies. To remedy this problem, Tenenbaum et al. [6] introduced the ISOMAP method, which uses an estimated geodesic distance instead of the conventional Euclidean one as input to the MDS procedure. However, this method requires expensive nearest neighbor and shortest path computations, and scales at least quadratically with the number of data points. A similar scaling problem plagues locally linear embedding (LLE) [7], a related approach that produces globally ordered maps by constructing locally linear relationships between the data points.

Recently, we introduced stochastic proximity embedding (SPE) [8], a novel self-organizing scheme that addresses the key limitations of ISOMAP and LLE. SPE builds on the same geodesic principle first proposed and exploited by ISOMAP, but introduces two important algorithmic advances: (1) it circumvents the calculation of estimated geodesic distances, and (2) it uses a pairwise refinement scheme that does not require the complete distance (d_{ij}) or proximity (r_{ij}) matrix and scales linearly with the number of points. The method minimizes the stress function:

$$S = \frac{\sum_{i<j}(f(d_{ij}, r_{ij})/r_{ij})}{\sum_{i<j} r_{ij}}$$

where $f(d_{ij}, r_{ij})$ is the pairwise stress defined as $f(d_{ij}, r_{ij}) = (d_{ij} - r_{ij})^2$ if $r_{ij} \leq r_c$ or $d_{ij} < r_{ij}$, and $f(d_{ij}, r_{ij}) = 0$ if $r_{ij} > r_c$ and $d_{ij} \geq r_{ij}$, and r_c is a pre-defined neighborhood radius. This is accomplished using a stochastic approximation of steepest descent that attempts to bring each individual term $f(d_{ij}, r_{ij})$ rapidly to 0. The method starts with an initial configuration and iteratively refines it by repeatedly selecting two points at random, and adjusting their coordinates so that their Euclidean distance on the map d_{ij} matches more closely their corresponding proximity r_{ij}. The correction is proportional to the disparity $\lambda(|r_{ij} - d_{ij}|/d_{ij})$, where λ is a learning rate parameter that decreases during the course of the refinement in order to avoid oscillatory behavior. If $r_{ij} > r_c$ and $d_{ij} \geq r_{ij}$, i.e. if the points are non-local and their distance on the map is already greater than their proximity r_{ij}, their coordinates remain unchanged. Unlike conventional MDS, SPE preserves exact distances between neighboring points and lower bounds between remote points, thus allowing the manifold to unfold and reveal its true intrinsic dimensionality.

Although the method is extremely fast, profiling experiments showed that a very significant fraction of the time required for the refinement was spent inside the random number generator (RNG). SPE requires two calls to the RNG for every pairwise refinement step, and for simple proximity measures such as the Euclidean distance or Tanimoto coefficient, this corresponds to a significant fraction of the overall computational work. Here, we describe an alternative update rule that reduces the number of RNG calls and thus improves the efficiency of the algorithm. The advantages of the new algorithm are demonstrated using three data sets of different origin, structure and intrinsic dimensionality.

2. Methods

2.1. Original stochastic proximity embedding algorithm

The original SPE algorithm proceeds as follows:

1. Initialize the D-dimensional coordinates of the N points, $\{x_{ik};\ i = 1, 2, \ldots, N;\ k = 1, 2, \ldots, D\}$. Select a cutoff distance r_c, and an initial learning rate $\lambda > 0$.
2. Select two points, i and j, at random, retrieve (or evaluate) their proximity in the input space, r_{ij}, and compute their Euclidean distance on the D-dimensional map, $d_{ij} = ||\boldsymbol{x}_i - \boldsymbol{x}_j||$. If $r_{ij} \leq r_c$, or if $r_{ij} > r_c$ and $d_{ij} < r_{ij}$, update the coordinates $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ by

$$\boldsymbol{x}_i \leftarrow \boldsymbol{x}_i + \frac{\lambda}{2}\frac{r_{ij} - d_{ij}}{d_{ij} + \varepsilon}(\boldsymbol{x}_i - \boldsymbol{x}_j)$$

and

$$\boldsymbol{x}_j \leftarrow \boldsymbol{x}_j + \frac{\lambda}{2}\frac{r_{ij} - d_{ij}}{d_{ij} + \varepsilon}(\boldsymbol{x}_j - \boldsymbol{x}_i)$$

where ε is a small number used to avoid division by 0 (here set to 1.0×10^{-10}). If $r_{ij} > r_c$ and $d_{ij} \geq r_{ij}$, leave the coordinates unchanged.
3. Repeat (2) for a prescribed number of steps, S.
4. Decrease the learning rate λ by a prescribed $\delta\lambda$.
5. Repeat (2)–(4) for a prescribed number of cycles, C.

2.2. Modified stochastic proximity embedding algorithm

The modified SPE algorithm proceeds as follows:

1. Initialize the D-dimensional coordinates of the N points, $\{x_{ik};\ i = 1, 2, \ldots, N;\ k = 1, 2, \ldots, D\}$. Select a cutoff distance r_c, and an initial learning rate $\lambda > 0$.
2. Select a point, i, at random (pivot). For every point $j \neq i$, retrieve (or evaluate) its proximity to i in the input space, r_{ij}, and compute their Euclidean distance on the D-dimensional map, $d_{ij} = ||\boldsymbol{x}_i - \boldsymbol{x}_j||$. If $r_{ij} \leq r_c$, or if $r_{ij} > r_c$ and $d_{ij} < r_{ij}$, update the coordinates $\boldsymbol{x}_j$ by

$$\boldsymbol{x}_j \leftarrow \boldsymbol{x}_j + \lambda\frac{r_{ij} - d_{ij}}{d_{ij} + \varepsilon}(\boldsymbol{x}_j - \boldsymbol{x}_i)$$

where ε is a small number used to avoid division by 0 (here set to 1.0×10^{-10}). If $r_{ij} > r_c$ and $d_{ij} \geq r_{ij}$, leave the coordinates unchanged.

3. Decrease the learning rate λ by a prescribed δλ.
4. Repeat (2)–(3) for a prescribed number of cycles, *C*.

2.3. Data sets

The new algorithm was tested on three different chemical data sets.

2.3.1. Ether

The first data set consists of 1000 conformations of methylpropylether, generated using a variant of SPE for conformational sampling [9]. Just like conventional distance geometry [10], this method uses covalent constraints to establish a set of upper and lower interatomic distance bounds, and then attempts to generate conformations that are consistent with these bounds. The proximity between conformations was measured by the RMSD, which is defined as the minimum Euclidean distance between the atomic coordinate vectors of two conformations superimposed through translations and rotations.

2.3.2. Amination library

The second data set represents a two-component virtual combinatorial library [11] containing 10,000 compounds, derived by combining 100 amines and 100 aldehydes using the reductive amination reaction. Each of the products was described by 117 topological descriptors including molecular connectivity indices, kappa shape indices, subgraph counts, information-theoretic indices, Bonchev-Trinajstic indices, and topological state indices [12]. To eliminate strong linear correlations, which are typical of graph-theoretic descriptors, the data were normalized and decorrelated using PCA. Molecular dissimilarity was defined as the Euclidean distance in the latent variable space formed by the 23 principal components that accounted for 99% of the total variance in the data.

2.3.3. Ugi library

The third data set represents a four-component virtual combinatorial library containing 10,000 compounds derived by combining 10 carboxylic acids, 10 primary amines, 10 aldehydes and 10 isonitriles using the Ugi reaction. Each of the products was described by a 166-dimensional binary fingerprint, where each bit encoded the presence or absence of a particular structural feature in the target molecule, as defined in the ISIS chemical database management system. Molecular dissimilarity was based on the Tanimoto coefficient:

$$r_{ij} = 1 - \frac{|\mathrm{AND(a, b)}|}{|\mathrm{IOR(a, b)}|}$$

where a and b represent two binary encoded molecules, *AND* is the binary "and" operation (a bit in the result is set if both of the corresponding bits in the two operands are set), and *IOR* is the binary "inclusive or" operation (a bit in the result is set if the either of corresponding bits in the two operands are set).

2.4. Implementation

All programs were implemented in the C++ programming language and are part of the DirectedDiversity® software suite [13]. All calculations were carried out on a Dell Inspiron 8000 laptop computer equipped with a 1.0 GHz Pentium III Intel processor running Windows 2000 Professional.

3. Results and discussion

We first show that our modified update rule leads to a reduction in stress over the course of the refinement. For simplicity, let us define the stress function as

$$S = \sum_{i \neq j} (d_{ij} - r_{ij})^2 = \sum_{i \neq j} S_{ij}$$

where $d_{ij} = |\boldsymbol{x}_i - \boldsymbol{x}_j|$ is the distance between points i and j on the non-linear map, r_{ij} is the desired distance between them, and $S_{ij} = (d_{ij} - r_{ij})^2$ is the pairwise stress. Given a randomly chosen pivot i, the change in the position $\boldsymbol{x}_j$ of point $j \neq i$ is

$$\Delta \boldsymbol{x}_j(i) = \lambda \nabla_i S_{ij} = -\lambda \nabla_j S_{ij}$$

Averaged over all possible choices of i, the expectation value of the move becomes

$$\langle \Delta \boldsymbol{x}_j \rangle = -\frac{\lambda}{N} \sum_i \nabla_j S_{ij} = -\frac{\lambda}{N} \nabla_j S$$

Thus, just like our original pairwise refinement scheme, the expected movement of a point follows a trajectory anti-parallel to the gradient of the stress function, and therefore the stress will decrease on average when these movements are of small magnitude. However, this scheme does not preclude temporary increases after a particular step. Indeed, the change in stress for a given i is

$$\Delta S(i) = \sum_{j \neq i} (\nabla_j S)^T \Delta \boldsymbol{x}_j = \sum_{j \neq i} \left(\sum_{k \neq j} \nabla_j S_{kj} \right)^T (-\lambda \nabla_j S_{ij})$$

$$= -\lambda \sum_{j \neq i} \left(||\nabla_j S_{ij}||^2 + \sum_{k \neq i,j} (\nabla_j S_{kj})^T (\nabla_j S_{ij}) \right)$$

Since the second term in the preceding equation may be negative, the stress may temporarily increase, allowing the algorithm to escape from local minima.

We examine the behavior of the new update rule in two different applications of proximity embedding. The first preserves all pairwise distances (i.e. it uses an infinite cutoff r_c) and corresponds to conventional MDS [14], whereas the second uses a neighborhood radius chosen according to the procedure described in [15]. The importance of the neighborhood radius is central in SPE. If the value of r_c is too small, the map disintegrates into a large number of disconnected

fragments and singletons, whereas if it is too large, it creates short-circuits, causing the surface to fold and conceal its intrinsic dimension. The method introduced in [15] selects the threshold that minimizes both the stress and the number of connected components in the neighborhood graph, i.e. one that produces a low stress configuration without causing excessive fragmentation of the data manifold. The value of r_c also affects the time that is required to reach convergence. When r_c is set to infinity, SPE operates at its maximum efficiency, since virtually all pairs result in "productive" work (i.e. a refinement in the coordinates of the selected points). However, as the cutoff decreases, an increasing fraction of pairwise comparisons do not result in any refinement once the general structure of the map has been established, since most of the remote points are already separated beyond their lower bounds. The total number of steps required for convergence is proportional to the fraction of pairs within the specified cutoff radius.

To ensure a fair comparison, each algorithm was run using the same learning schedule (initial and final learning rates) and the same number of cycles and steps. The latter was set to $N-1$, where N is the number of points in the data set. Thus, the only difference between the two algorithms was the replacement of the stochastic inner loop in the original SPE formalism with a deterministic one using a randomly chosen pivot. In the supporting information of our original paper [8], we showed that SPE is relatively insensitive to the learning schedule, as long as prudent learning rates are chosen. A large learning rate at the beginning of the embedding induces fast reorganization of the data points, while a small learning rate towards the end avoids oscillatory behavior. As long as the final learning rate is any number smaller than 1.0, SPE converges to the optimum embedding given a sufficient number of refinement cycles. The learning rates used in this study were $\lambda_0 = 2.0$ and $\lambda_1 = 0.01$.

The rate of convergence of the two algorithms was monitored by embedding the data using different numbers of cycles, and recording the stress of the final configuration. To assess the dependence of the final embedding on the initialization conditions, each mapping experiment was carried out 10 times, each starting from the same random initial configuration but following a different random trajectory (random number seed). The embeddings were limited to two dimensions in order to simplify the visualization and interpretation of the maps.

As illustrated in Figs. 1–3, the original (SPE) and pivot-based (PSPE) algorithms are practically equivalent. While PSPE appears to converge faster than the original algorithm for classical embeddings ($r_c = \infty$), this advantage vanishes with isometric maps. Still, even in the former case, a satisfactory stress is reached at a comparable number of cycles, typically around 1000 for all three data sets. As one would expect from intuition, the first few cycles of PSPE are far more sensitive to the random seed as manifested by the substantially greater variance of the stress value, since the overall organization of the map is dependent on the specific pivots employed. However, as the number of cycles increases, the pivots become more representative of the data sample as a whole, and the map self-organizes reliably to a globally ordered configuration. This is not the case with the pairwise refinement scheme, which provides much more uniform sampling in the early stages of refinement. This difference is not observed when isometry is preserved (i.e. when a neighborhood cutoff is used), and the two algorithms are essentially identical. The exact number of cycles that are necessary to fully unfold the non-linear manifold depends on its curvature and the frequency at which it is sampled, and may vary from case to case.

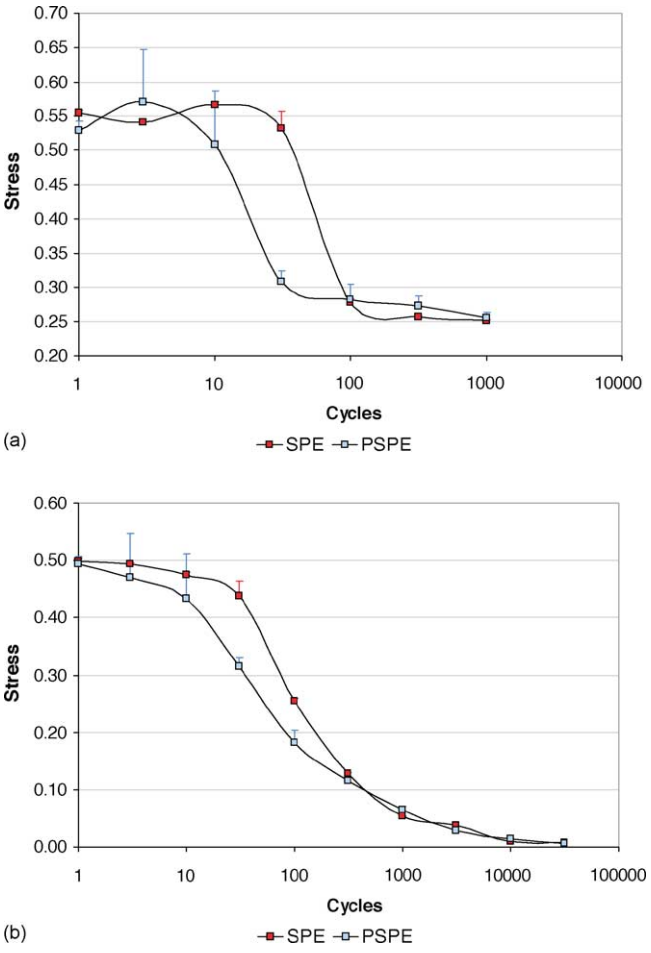

Fig. 1. Effect of the number of cycles on the final stress of the two-dimensional embeddings of the methylpropylether conformations obtained by SPE and PSPE. Each point represents the mean and standard deviation of 10 independent runs starting from the same random initial configuration and using a different random trajectory: (a) $r_c = \infty$; (b) $r_c = 0.2$.

The advantage of the modified rule is that it eliminates the need for a call to the RNG inside the inner SPE loop. As illustrated in Fig. 4, this results in a two-to-three-fold speed-up in execution time, though the exact value depends on the complexity of the distance function and the output dimensionality.

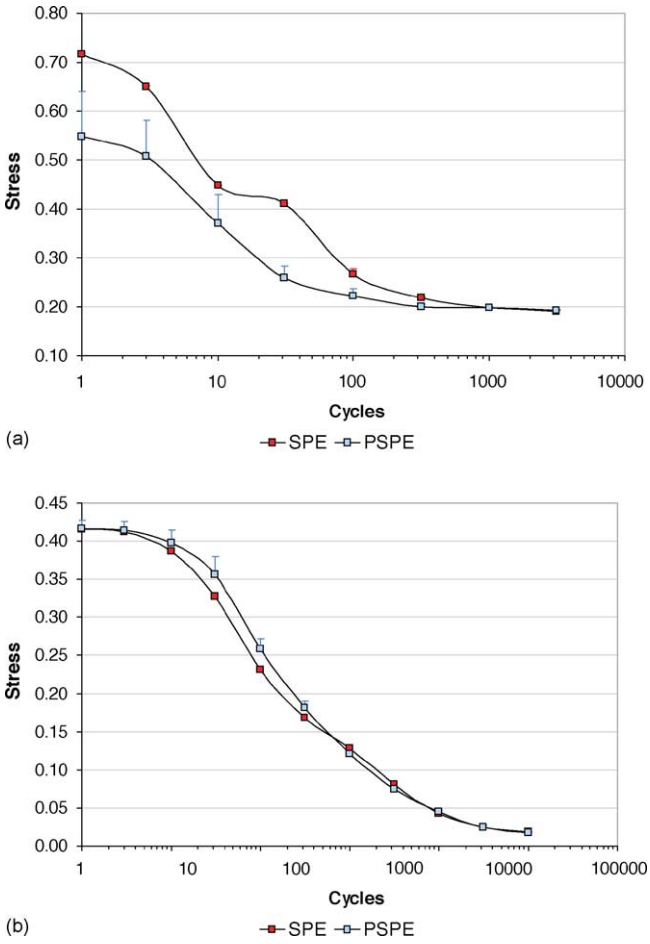

Fig. 2. Effect of the number of cycles on the final stress of the two-dimensional embeddings of the amination library obtained by SPE and PSPE. Each point represents the mean and standard deviation of 10 independent runs starting from the same random initial configuration and using a different random trajectory: (a) $r_c = \infty$; (b) $r_c = 0.4$.

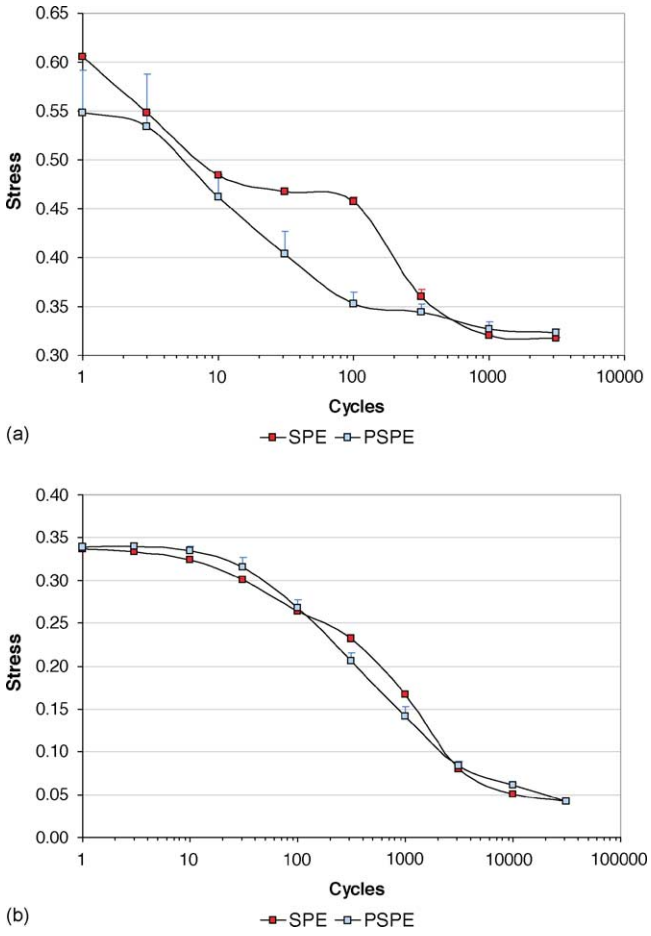

Fig. 3. Effect of the number of cycles on the final stress of the two-dimensional embeddings of the Ugi library obtained by SPE and PSPE. Each point represents the mean and standard deviation of 10 independent runs starting from the same random initial configuration and using a different random trajectory: (a) $r_c = \infty$; (b) $r_c = 0.15$.

As illustrated in Figs. 5–7, the resulting maps are visually compelling and highly informative. When the data is embedded in a space of the intrinsic dimension, the map reveals the true underlying variables in a manner that is consistent with intuition. For example, the two-dimensional isometric embedding of the methylpropylether conformations exhibits 0 stress and its principal axes correlate very strongly with the molecule's intrinsic conformational degrees of freedom (the two central rotatable bonds), even though it is constructed using a similarity measure that is based on an atom-by-atom superposition (RMSD) and has apparent dimensionality of nine. The low-dimensional maps exhibit a meaningful structure even when the intrinsic dimensionality is much higher and the data set lacks a clear manifold geometry. For example, the combinatorial libraries illustrated in Figs. 6 and 7 using two different sets of descriptors and distance metrics exhibit clusters that correspond to distinct chemical classes resulting from the discrete nature of the descriptors and the diversity of the chemical fragments employed. These maps, which are discussed in greater detail in [15], are

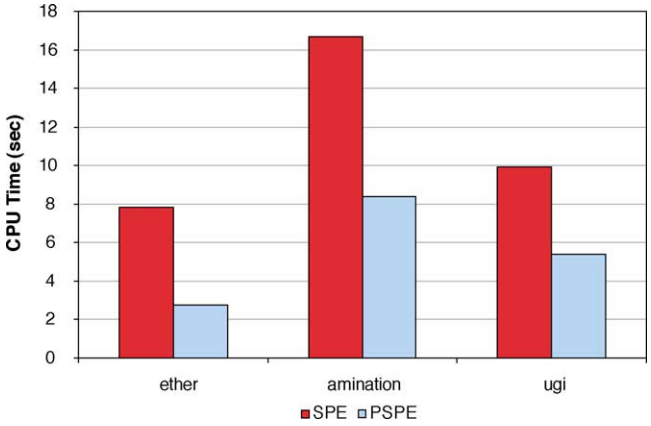

Fig. 4. CPU time required for 10,000,000 refinement steps using SPE and PSPE for each of the three data sets.

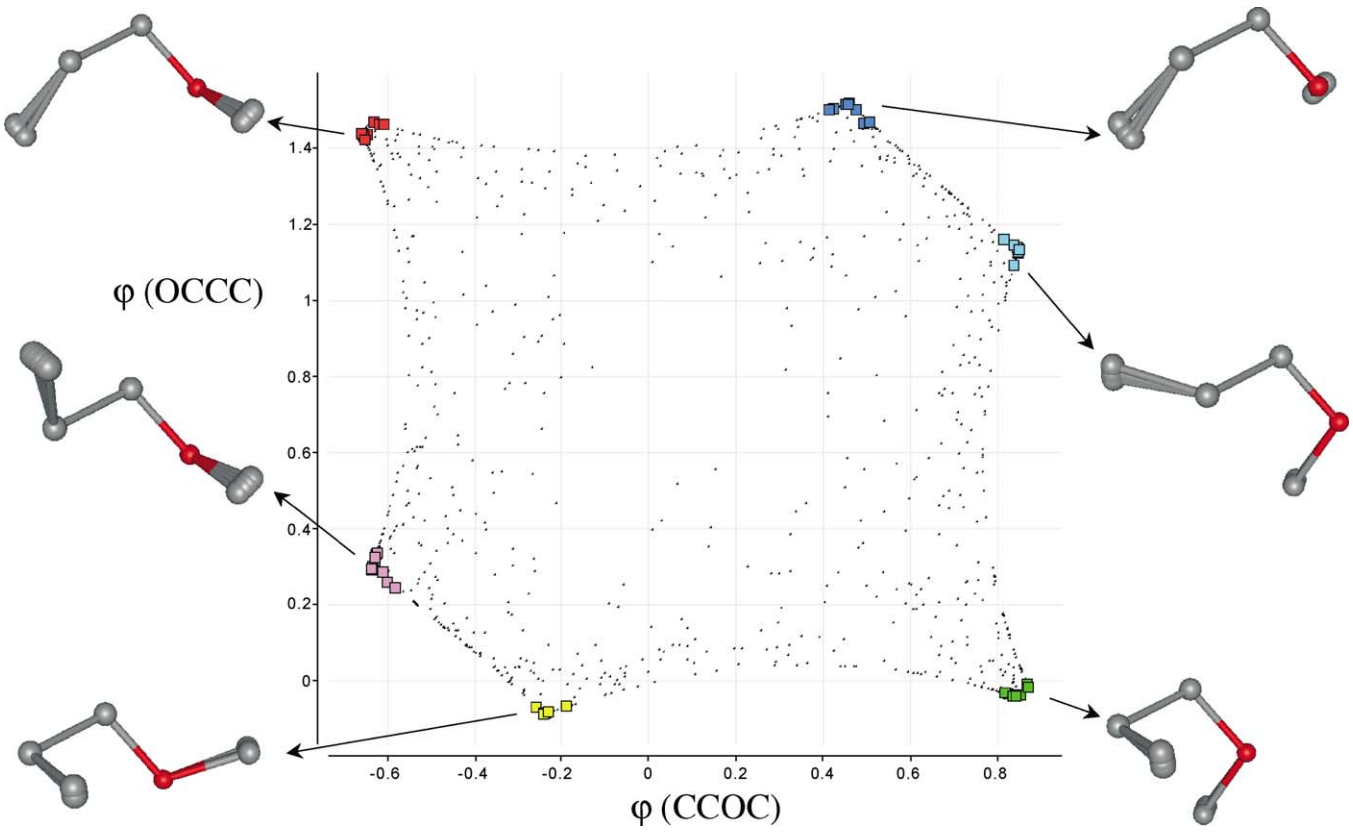

Fig. 5. Two-dimensional stochastic proximity map of the methylpropylether conformations obtained using $r_c = 0.2$. Representative conformations are shown next to the highlighted points, all superimposed along the central CCO atoms and displayed in the same orientation. Reproduced with permission from [15].

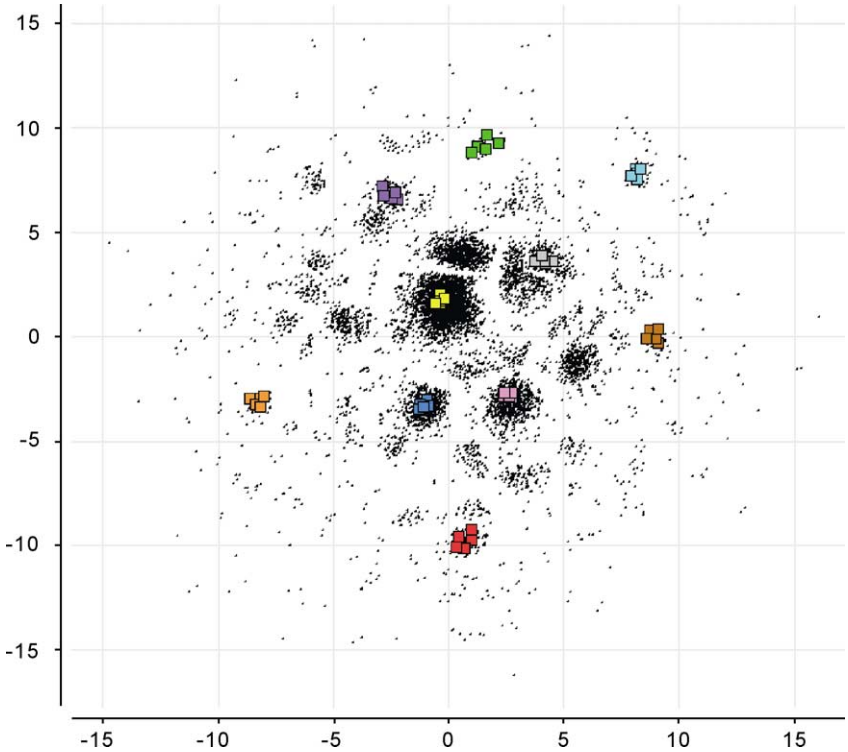

Fig. 6. Two-dimensional stochastic proximity map of the amination library obtained using $r_c = 0.4$. Ten representative clusters of closely related compounds (10 random compounds along with their 9 nearest neighbors) are highlighted in different colors to demonstrate the ability of SPE to preserve close proximities. Reproduced with permission from [15].

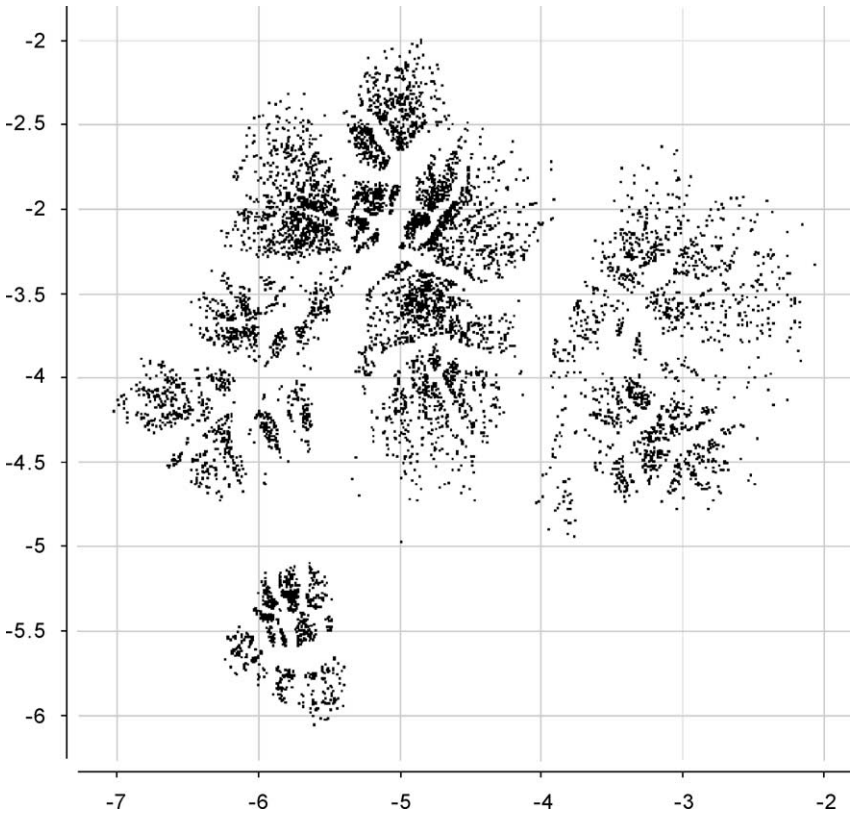

Fig. 7. Two-dimensional stochastic proximity map of the Ugi library obtained using $r_c = 0.15$.

illustrative of the ability of SPE to identify natural clusters in the data set without prior knowledge or expert guidance. Since SPE is based exclusively on distance comparisons, it is applicable to data sets of any input dimension. Although the highest dimensionality examined in this work is 166, SPE can be easily applied to other types of binary fingerprints that are commonly employed in molecular similarity and diversity studies and typically consist of a few thousand bits. Indeed, the only additional overhead associated with such representations is that of computing the distance function, which scales linearly with the number of bits.

A final remark. While this manuscript was being drafted, we became aware of a paper by Demartines and Hérault [16] describing a related method for non-linear dimensionality reduction known as curvilinear component analysis (CCA). Although the objective function is very different, CCA uses an optimization heuristic that is very similar to the one described in this work. The principal difference between the two methods is that CCA utterly disregards remote distances, whereas SPE differentiates them from local distances by their intrinsic relationship to the true geodesic distances, and utilizes both types accordingly in order to improve the embedding. In essence, our method views the input distances between remote points as lower bounds of their true geodesic distances, and uses them as a means to impose global structure.

4. Conclusions

SPE is a fast and scalable method for producing low-dimensional Euclidean maps that preserve the intrinsic dimensionality and non-linear geometry of complex high-dimensional observation spaces. For typical distance functions, SPE consumes most of the time generating random numbers. The optimization heuristic presented in this paper effectively eliminates random number generation as the rate-limiting step, and results in substantial savings in the CPU time required for the embedding, without affecting the quality of the resulting map. Of course, this approach is one of many that can be used to improve the efficiency of the original SPE algorithm. Since the role of the RNG is to make sure that the algorithm is not biased towards certain points or pairs of points thus introducing systematic errors in the embedding, a crude but fast RNG or even a systematic protocol for selecting the pairs would probably suffice.

References

[1] H. Hotelling, J. Educ, Psychology 24 (1933) 417–441, 498–520.
[2] I. Borg, P.J.F. Groenen, Modern Multidimensional Scaling: Theory and Applications, Springer, New York, 1997.
[3] R.N. Shepard, J.D. Carroll, in: P.R. Krishnaiah (Ed.), International Symposium on Multivariate Analysis, Academic Press, New York, 1965, pp. 561–592.

[4] T. Martinetz, K. Schulten, Neural Netw. 7 (1994) 507–522.
[5] J.W. Sammon, IEEE Trans. Comp. 18 (1969) 401–409.
[6] J.B. Tenenbaum, V. de Silva, J.C. Langford, Science 290 (2000) 2319–2323.
[7] S.T. Roweis, L.K. Saul, Science 290 (2000) 2323–2326.
[8] D.K. Agrafiotis, H. Xu, Proc. Natl. Acad. Sci. U.S.A. 99 (2002) 15869–15872.
[9] H. Xu, S. Izrailev, D.K. Agrafiotis, J. Chem. Inf. Comput. Sci. 43 (2003) 475–484.
[10] G.M. Crippen, T.F. Havel, Distance Geometry and Molecular Conformation, Research Studies Press, Somerset, UK, 1988.
[11] D.K. Agrafiotis, V.S. Lobanov, F.R. Salemme, Nat. Rev. Drug Discov. 1 (2002) 337–346.
[12] L.H. Hall, L.B. Kier, in: D.B. Boyd, K.B. Lipkowitz (Eds.), Reviews in Computational Chemistry, VCH Publishers, New York, 1991, pp. 367–422.
[13] D.K. Agrafiotis, R.F. Bone, F.R. Salemme, R.M. Soll, System and method for automatically generating chemical compounds with desired properties, US Patent 5,463,564, Issued October 31.
[14] D.K. Agrafiotis, J. Comput. Chem. 24 (2003) 1215–1221.
[15] D.K. Agrafiotis, H. Xu, J. Chem. Inf. Comput. Sci. 43 (2003) 475–484.
[16] P. Demartines, J. Hérault, IEEE Trans. Neural Netw. 8 (1997) 148–154.