Computing and Informatics, Vol. 31, 2012, 1417–1440

PROCRUSTES ANALYSIS OF TRUNCATED LEAST SQUARES MULTIDIMENSIONAL SCALING

Marcin Kurdziel, Krzysztof Boryczko, Witold Dzwinel

AGH University of Science and Technology
Faculty of Computer Science, Electronics and Telecommunications
Department of Computer Science
al. A. Mickiewicza 30
30-059 Krakow, Poland
e-mail: {kurdziel, boryczko, dzwinel}@agh.edu.pl

Communicated by Jacek Kitowski

Abstract. Multidimensional Scaling (MDS) is an important class of techniques for embedding sets of patterns in Euclidean space. Most often it is used to visualize in $\mathbb{R}^3$ multidimensional data sets or data sets given by dissimilarity measures that are not distance metrics. Unfortunately, embedding n patterns with MDS involves processing $O(n^2)$ pairwise pattern dissimilarities, making MDS computationally demanding for large data sets. Especially in Least Squares MDS (LS-MDS) methods, that proceed by finding a minimum of a multimodal stress function, computational cost is a limiting factor. Several works therefore explored approximate MDS techniques that are less computationally expensive. These approximate methods were evaluated in terms of correlation between Euclidean distances in the embedding and the pattern dissimilarities or value of the stress function. We employ Procrustes Analysis to directly quantify differences between embeddings constructed with an approximate LS-MDS method and embeddings constructed with exact LS-MDS. We then compare our findings to the results of classical analysis, i.e. that based on stress value and correlation between Euclidean distances and pattern dissimilarities. Our results demonstrate that small changes in stress value or correlation coefficient can translate to large differences between embeddings. The differences can be attributed not only to the inevitable variability resulting from the multimodality of the stress function but also to the approximation errors. These results show that approximation may have larger impact on MDS than what was thus far revealed by analyzes of stress value and correlation between Euclidean distances and pattern dissimilarities.

Keywords: Approximate multidimensional scaling, least squares multidimensional scaling, procrustes analysis, approximation errors, three-dimensional embedding

Mathematics Subject Classification 2010: 68T10, 91C15, 68W25, 90C26, 65C35

1 INTRODUCTION

Multidimensional scaling is a class of techniques for embedding sets of patterns in Euclidean space. These methods seek to replicate pairwise pattern distances or dissimilarities by the distances between the points in the embedding. Typically MDS is used for visualization purposes and therefore patterns are usually embedded in $\mathbb{R}^3$ or $\mathbb{R}^2$.

Let d_{ij} be the input pairwise pattern dissimilarities and $\|\mathbf{r}_i - \mathbf{r}_j\|$, $\mathbf{r}_i, \mathbf{r}_j \in Y$ be the distances between points in the Euclidean embedding Y. In Classical MDS (see e.g. [7, Chapter 2.2]) patterns are assumed to be points in high-dimensional Euclidean space and a low dimensional representation is constructed by performing an eigendecomposition of a Gram matrix $\mathbf{B}$ constructed from the distance matrix $[d_{ij}]_{n\times n}$. This technique has also been applied to the cases where $[d_{ij}]_{n\times n}$ is not an Euclidean distance matrix, and therefore $\mathbf{B}$ is not a proper Gram matrix. Then, either the eigenvectors of $\mathbf{B}$ that correspond to negative eigenvalues are discarded or some constant is added to the dissimilarities d_{ij} so that $\mathbf{B}$ becomes positive semidefinite [7, Chapter 2.2.3]. Another approach to embedding non-Euclidean patterns, Least Squares Multidimensional Scaling (see e.g. [7, Chapter 2.4]), proceeds by finding a minimum to a stress function that reflects discrepancy between dissimilarities d_{ij} and distances in the embedding, $\|\mathbf{r}_i - \mathbf{r}_j\|$. Various stress functions can be used with LS-MDS [7, Chapter 2.4]. One simple, but often used function of this kind is the *raw stress*:

$$\mathrm{RawStress}\,(R) = \sum_{\substack{\mathbf{r}_i,\mathbf{r}_j\in Y \\ i<j}} \left(\|\mathbf{r}_i - \mathbf{r}_j\| - d_{ij}\right)^2. \tag{1}$$

The strength of the LS-MDS is that it can be used as is for embedding sets of patterns that are not only non-Euclidean but are in fact given by dissimilarity measure that is not even a proper distance metric.

Embedding n-pattern data set with MDS requires processing $\frac{n\cdot(n-1)}{2}$ pairwise pattern dissimilarities, rendering MDS a computationally demanding process. This is particularly significant limitation in LS-MDS. Stress functions employed in LS-MDS are typically multimodal – see e.g. [20] for an investigation of local minima of a class of stress functions that include the raw stress. One usually needs to employ stochastic optimization methods to find a deep local minimum of the stress

function. Stochastic optimization methods over stress functions involving $O(n^2)$ terms are, however, expensive for any but small data sets. These difficulties led to research on approximate MDS methods that use only a subset of pattern dissimilarities when embedding a data set in an Euclidean space [27, 6, 13, 19, 15, 3, 22, 2, 23]. The main observation here is that in MDS constraints on point positions may, in many cases, be an overdetermined system. For example, in a perfect embedding in $\mathbb{R}^3$ position of a point is determined by its distances from any four non-planar points. Still, MDS will normally account for its distances from all other points in the embedding. Approximate methods therefore seek to reduce the number of pairwise pattern dissimilarities accounted for in MDS while keeping the distortion to the resultant embedding at an acceptable level. Two approaches were typically used for measuring the distortion to the embedding resulting from not accounting for certain dissimilarities. In the first approach all $\frac{n\cdot(n-1)}{2}$ Euclidean distances, $\|\mathbf{r}_i - \mathbf{r}_j\|$, are calculated after performing approximate MDS and their correlation to the input pattern dissimilarities d_{ij} is used to assess the quality of the approximate embedding. The correlation coefficient for the embedding constructed with exact MDS is used as a reference. Second approach, used mainly with LS-MDS, employs stress value (as in e.g. Equation (1)) instead of correlation coefficients.

Even though stress value and correlation between distances and dissimilarities are the two main quality measures in studies on approximate MDS, it is not clear to what extent they reflect the level of geometrical distortion exhibited by an approximate embedding relative to the exact embedding. In particular, disregarding certain dissimilarities may allow for locally optimal solutions that, while reflecting pairwise pattern dissimilarities, differ geometrically from solutions explored by exact MDS. This may have significant implications when employing MDS in its typical application, namely data exploration through embedding in $\mathbb{R}^3$ or $\mathbb{R}^2$. In this work we therefore employ Procrustes Analysis (see e.g. [7, Chapter 5]) to directly compare three-dimensional embeddings constructed with an approximate LS-MDS method to the corresponding reference embeddings constructed with exact LS-MDS. We chose Procrustes Analysis as it fits MDS configurations in a way that minimize sum of squared distances between their points [7, Chapter 5.2]. Therefore, it is well suited for direct geometrical comparison. Although, when one seeks to assess correlational rather than geometrical similarity between configurations, measures such as Congruence Coefficient [4, Chapter 19.7] can also be used.

We relate results of comparison with Procrustes Analysis to the classical quality assessment that relies on stress value and correlation between distances and dissimilarities. For the approximate method we employ LS-MDS with modified raw stress functions that account for only a small subset of pattern dissimilarities – in the following part of this work we refer to this technique as *truncated LS-MDS*. Three large data sets are used in the experiments, including two that employ dissimilarity measures that are not proper distance metrics.

2 RELATED WORK

Initial research on approximate Multidimensional Scaling, presented in [27, 13, 15], focused on Classical MDS. These works propose and explore a technique that reduces the number of pairwise pattern comparisons in MDS by carefully selecting a subset of patterns, called *frame*, and then factoring similarities between frame and non-frame patterns. The approach was evaluated mainly in terms of correlation coefficients between pattern similarities and recovered distances, and compared to Classical MDS over both complete similarity matrices and matrices containing only a subset of similarities. Obtained results showed that the frame-based approach is advantageous over Classical MDS with incomplete similarity matrices.

A simplified, frame-based approach was also applied to LS-MDS [6]. Therein the authors propose a heuristics for constructing approximate Sammon mapping [25], which first maps a subset of patterns in $\mathbb{R}^2$ and then maps the remaining patterns, taking into account only the distances to the patterns in the already mapped subset. Two example maps are reported to demonstrate the performance of this approach. Further work along this line explored sequential mapping of patterns [19]. Algorithms proposed therein attempt to preserve a small subset of pairwise pattern distances, namely distances in the Minimal Spanning Tree (MST) of patterns or a combination of MST distances and distances to a chosen, *reference*, pattern. Performance of this approach was demonstrated with three example embeddings. An improvement over purely sequential mapping was proposed in [3]. This method combines frame-based heuristic with sequential mapping of MST distances. Initially, a subset of patterns is embedded in $\mathbb{R}^2$ with Sammon mapping and then serves as the frame for the remaining patterns. Non-frame patterns are embedded employing a variant of the method proposed in [19], which considers only the distances between frame and non-frame patterns. The approach was compared to six related methods, over four different data sets. Comparison included several performance indices, namely embedding runtime, residual errors, e.g. stress value in the case of Sammon mapping, leave-one-out classification accuracy when using embedded data set for pattern classification and percentage of MST edges preserved by the embedding. In addition, several embedding results were reported. Another MST-based heuristic for approximate Sammon mapping was proposed in [22]. This method divides MST of the data set into a number of subtrees, which are then used in a hierarchical embedding procedure. The procedure follows the edges removed during subdivision of MST and for every such edge adds up to four patterns to the constructed embedding. Patterns are added with a modified Sammon mapping, in which positions of previously embedded patterns are fixed and a genetic algorithm is used to minimize the Sammon mapping criterion. Experimental evaluation of this technique was carried out on one multidimensional data set and employed two indices of embedding quality. One of the indices was stress values for different dimensionality of embedding and number of generations in the genetic algorithm. The other was a scatter-plot and corresponding Spearman's rank correlation coefficient between input pattern distances and

distances in a six-dimensional embedding. Result of embedding in $\mathbb{R}^2$ was also reported.

A different approach to finding minima of LS-MDS stress functions was proposed in [9]. It employs particle dynamics with conservative forces derived from the stress function, treated therein as a potential function for pairwise particle interactions. In addition, friction forces are included in the particle dynamics and steadily decrease kinetic energy of the system. This approach is well suited for approximate MDS – particle dynamics can be performed with a small subset of $\frac{n\cdot(n-1)}{2}$ conservative interactions, in effect realizing LS-MDS over a small subset of input pattern dissimilarities. Approximate LS-MDS techniques of this kind were recently studied in [2, 23]. Performance of these methods was evaluated in terms of stress value. Obtained results demonstrated that for large data sets, approximation as strong as several-fold decrease in the number of considered input pattern dissimilarities often corresponds to only a small increase in the stress value.

3 LEAST SQUARES MULTIDIMENSIONAL SCALING WITH DISSIPATIVE PARTICLE DYNAMICS

Experiments reported in this work were carried out using explicit-temperature, particle dynamics-based LS-MDS algorithm we described in [2]. The algorithm extends particle dynamics-based LS-MDS method introduced in [9] by employing Dissipative Particle Dynamics (DPD) [18, 10] instead of classical particle dynamics with friction forces. This extension introduces an explicit temperature control to the particle dynamics-based stress minimization technique, in effect providing an explicit control over the breadth of exploration of the stress function. A similar simulated annealing-like, molecular dynamics-based LS-MDS technique was recently proposed in [1]. In the context of Sammon Mapping, simulated annealing was also studied in [8].

In DPD, particle dynamics is governed by three kinds of forces, namely conservative forces $\mathbf{F}^C$, dissipative forces $\mathbf{F}^D$ and random forces $\mathbf{F}^R$ [10]. Therefore, total force acting on the i^{th} particle is:

$$\mathbf{F}_i = \sum_{j\neq i} \mathbf{F}^C\left(\mathbf{r}_{ij}\right) + \sum_{j\neq i} \mathbf{F}^D\left(\mathbf{r}_{ij}, \mathbf{v}_{ij}\right) + \sum_{j\neq i} \mathbf{F}^R\left(\mathbf{r}_{ij}\right). \tag{2}$$

In the above, $\mathbf{r}_{ij}$ is a vector pointing from the i^{th} to the j^{th} particle and $\mathbf{v}_{ij}$ is a velocity of the i^{th} particle relative to the j^{th} particle. Dissipative and random forces act between pairs of particles and, respectively, dissipate and add kinetic energy to the system:

$$\begin{aligned} \mathbf{F}^D\left(\mathbf{r}_{ij}, \mathbf{v}_{ij}\right) &= -\gamma\omega\left(\|\mathbf{r}_{ij}\|\right)^2 \left[\mathbf{v}_{ij} \circ \mathbf{e}_{ij}\right] \mathbf{e}_{ij}, \\ \mathbf{F}^R\left(\mathbf{r}_{ij}\right) &= \sigma\omega\left(\|\mathbf{r}_{ij}\|\right) \xi_{ij} \mathbf{e}_{ij}. \end{aligned} \tag{3}$$

Here, $\mathbf{e}_{ij}$ is a unit vector pointing from the i^{th} to the j^{th} particle and $\xi_{ij} = \xi_{ji}$ are random variables drawn from normal distribution with zero mean and unit variance[1]. The temperature of the system is maintained by coupling the coefficients σ and γ:

$$\sigma = \sqrt{2\gamma k_B T}, \tag{4}$$

where k_B is the Boltzmann constant[2] and T is the desired temperature of the system. The weighting function $\omega(\cdot)$ is most often modelled to give soft repulsive potential:

$$\omega(r) = \begin{cases} 1 - \frac{r}{r_c} & r <= r_c \\ 0 & r > r_c \end{cases}, \tag{5}$$

where r_c is a cut-off distance at which dissipative and random interactions vanish.

Particle dynamic can be harnessed to perform LS-MDS by assigning one particle to each pattern, putting the MDS stress function as the potential for the conservative interactions and performing the dynamics with kinetic energy dissipation [9]. In this work we employ raw stress function (Equation (1)) as the potential for DPD's conservative interactions, which leads to a following conservative force acting on particles:

$$\mathbf{F}_i^C = \sum_{j \neq i} \mathbf{F}^C(\mathbf{r}_{ij}) = -\sum_{j \neq i} \nabla V(\mathbf{r}_{ij}) = -2 \sum_{j \neq i} (\|\mathbf{r}_{ij}\| - d_{ij}) \mathbf{e}_{ij}. \tag{6}$$

Conclusions of this work do not depend on any particular choice of stress function, and the raw stress was chosen mainly to simplify the experiments. Other stress functions can also be used with particle dynamics-based LS-MDS, as long as their gradient gives rise to pairwise particle interactions. Unlike in [9], energy dissipation in this work is not a matter of simple friction forces but is controlled by the system's temperature T. We begin with a temperature T corresponding to particles travelling, on average, a mean pairwise pattern dissimilarity in a unit time and then minimize the stress by lowering the temperature until it approaches zero and particles stop, giving the final MDS configuration.

Pseudo-code for this embedding technique is reported in Algorithm 3. The pseudo-code applies to truncated LS-MDS, mentioned in Section 1, as well as to exact LS-MDS when all $\frac{n \cdot (n-1)}{2}$ terms from the stress function are included in the calculated set (`S`).

4 PROCRUSTES ANALYSIS

Multidimensional Scaling techniques do not enforce one fixed orientation in the Euclidean space for the constructed embeddings. In particular, any geometrical trans-

[1] We employ a uniform distribution with zero mean and unit variance, which suffice for explicit temperature control.

[2] For simplicity, we set $k_B = 1$ in this work.

formation of the embedding that do not change Euclidean distances is admissible as far as MDS is concerned. For the same reason, MDS may not enforce location of the embedding's center in the Euclidean space. When comparing an approximate MDS embedding to its corresponding exact variant, one should also acknowledge that certain differences may arise from small uniform dilation exhibited by the approximate result. Dilation of this kind does not change structure of the embedding and in typical MDS applications does not change interpretation of the result. To avoid overestimation of differences between approximate and exact embeddings, uniform dilation should be neglected in the comparison. We employ Procrustes Analysis [7, Chapter 5] for comparing embeddings that, in addition to differences due to approximation errors, are possibly rotated, translated and rescaled with respect to each other.

In essence, Procrustes Analysis (PA) consists of several matrix operations that move the compared embeddings to the origin of the coordinate system as well as rotate and rescale one of them to match the orientation and diameter of the other. In more precise terms, PA finds the optimal rotation, translation and scaling of ordered, equinumerous sets of points (configurations) with respect to their root mean square distance (RMSD). Note that the ordering of points is fixed in this fitting, i.e. first point in one of the sets corresponds to the first point in the other set, second point in one set corresponds to the second point in the other set, and so on with third and subsequent points.

Implementation of PA employed in this work is based on [7, Chapter 5.2] and the following summary of steps in fitting point configurations is based on the description given therein. Let $\mathbf{x}_i, \mathbf{y}_i \in \mathbf{R}^3,\ i = 1 \dots n$, be the two compared MDS embeddings and $\mathbf{X}_{n\times 3}$ and $\mathbf{Y}_{n\times 3}$ be their corresponding coordinate matrices. In PA translation is normalized by centering the configurations at the origin of the coordinate system, i.e. setting $\mathbf{x}'_i = \mathbf{x}_i - \overline{\mathbf{x}}$ and $\mathbf{y}'_i = \mathbf{y}_i - \overline{\mathbf{y}}$, where $\overline{\mathbf{x}}$ and $\overline{\mathbf{y}}$ are the mean rows of $\mathbf{X}$ and $\mathbf{Y}$, respectively. Let $\mathbf{X}'$ and $\mathbf{Y}'$ be the coordinate matrices of the centered configurations. To fit the configurations with respect to rotation, PA rotates $\mathbf{X}'$ with a rotation matrix $\mathbf{R}$ constructed through singular value decomposition of the matrix $\mathbf{Y}'^{\mathrm{T}}\mathbf{X}'$. That is, given the singular value decomposition $\mathbf{Y}'^{\mathrm{T}}\mathbf{X}' = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^{\mathrm{T}}$, the optimal rotation matrix is:

$$\mathbf{R} = \mathbf{V}\mathbf{U}^{\mathrm{T}}. \tag{7}$$

Finally, to fit the configurations with respect to scaling, PA dilates the rotated matrix $\mathbf{X}'\mathbf{R}$ by a factor σ of:

$$\sigma = \frac{\operatorname{tr}\left(\mathbf{R}\mathbf{Y}'^{\mathrm{T}}\mathbf{X}'\right)}{\operatorname{tr}\left(\mathbf{X}'\mathbf{X}'^{\mathrm{T}}\right)}. \tag{8}$$

The pseudo-code for this procedure is given in Algorithm 2.

In this work we measure the difference between the fitted configurations by their RMSD divided by the root mean square distance of points from the configu-

rations' center, i.e. by:

$$\text{N-RMSD} = \sqrt{\frac{\sum_{i=1}^{n} \|\sigma \mathbf{R}^{\mathrm{T}} \mathbf{x}'_i - \mathbf{y}'_i\|^2}{\frac{1}{2}\sum_{i=1}^{n} \left(\|\sigma \mathbf{R}^{\mathrm{T}} \mathbf{x}'_i\|^2 + \|\mathbf{y}'_i\|^2\right)}}. \tag{9}$$

In the next part of this work we refer to the measure given above as *normalized RMSD* (N-RMSD). Note that it captures a relative rather than absolute difference between the configurations and, therefore, can be used to interpret the result irrespective of the mean pairwise pattern dissimilarity.

5 TEST DATA SETS AND EXPERIMENTS

Experiments reported in this work were carried out on three large, dense test data sets. One of these sets represents a classical case of feature vector data, i.e. it consists of multidimensional vectors enclosed by the Euclidean distance. The other two data sets are enclosed by dissimilarity measures that are not proper distance metrics, and describe gene expression patterns and sequences of non-coding RNA molecules.

The multidimensional feature vector data set was published in [12] and provided in the UCI Machine Learning Repository [11]. The data set consists of 20 000 sixteen-dimensional vectors, describing capital letters in the English alphabet written in 20 different fonts. All vector attributes are integer numbers and, as stated above, their dissimilarity is measured by the Euclidean distance. In the following sections, we refer to this data set as *Letter Data*.

Second test set, further called *Gene Data*, is an example of data produced in a modern functional genomics study. It was prepared from data published in [14, Supplementary Table 17], namely gene expression levels measured by The National Human Genome Research Institute Model Organism Encyclopedia of DNA Elements (modENCODE) Project [5] using high throughput RNA sequencing. More precisely, we use abundances of modENCODE genes in the Drosophila melanogaster transcriptome, measured in 30 samples and reported as FPKM values [26]. From this data, abundances of 14 861 genes that have FPKM >= 1 in at least one sample were included in the test data set. Similarities between gene expression profiles were assessed by correlation coefficients between log-transformed FPKM values. That is, dissimilarity between i^{th} and j^{th} gene was measured as:

$$d_{ij} = 1.0 - \sigma_{ij}^2, \tag{10}$$

where σ_{ij} is the Pearson correlation coefficient between log-transformed FPKM values[3] for the i^{th} and the j^{th} gene. With this measure, genes with well correlated or anti-correlated transcript abundances are considered to be similar patterns, while genes with uncorrelated transcript abundances are distant patterns.

Third data set, further called ncRNA Data, exemplifies material gathered in numerous studies focusing on sequencing bio-molecules. It consists of sequences

[3] i.e. $\log(\text{FPKM} + 1.0)$

of 32 897 non-coding RNA (ncRNA) molecules exhibiting less than 90 % pairwise sequence identity, published in the RNA Families database [16, 17] release 8.0. To calculate dissimilarities in this test data set we performed pairwise sequence alignment [21], [24, pp. 76–81] for every pair of sequences. We then set the dissimilarities to:

$$d_{ij} = \left(1.0 - \frac{s_{ij}}{s_{ii}}\right) \cdot \left(1.0 - \frac{s_{ij}}{s_{jj}}\right), \tag{11}$$

where s_{kl} is a pairwise sequence alignment score for the k^{th} and the l^{th} ncRNA sequence. The above measure was designed to alleviate the impact of the sequence length on the dissimilarity value. Even short but well aligned sequences will have dissimilarities approaching zero, whereas dissimilarities of poorly aligned sequences, including short ones, will approach 1.0. Note that this dissimilarity measure may severely break triangle inequality – different parts of one sequence may align well to two other sequences, leading to small dissimilarity values, even though the two other sequences do not align to each other, and thus have high pairwise dissimilarity.

Using the algorithm described in Section 3, for each data set we constructed six different, three-dimensional LS-MDS embeddings. Each embedding projected all patterns from the embedded data set into $\mathbb{R}^3$. However, one embedding in each data set accounted for all $\frac{n\cdot(n-1)}{2}$ pairwise pattern dissimilarities, and served as a *reference embedding*, while other five *approximate embeddings* accounted for only a subset of dissimilarities. More precisely, these five embeddings were constructed with the procedure described in Section 3 run over a *truncated raw stress function* (see Algorithm 3), i.e. stress function constructed from Equation (1) by removing most of the summation terms and preserving only a small subset. The five approximate embeddings were constructed with truncated stress functions accounting for approximately 5 %, 7.5 %, 10.0 %, 12.5 % and 15.0 % of the $\frac{n\cdot(n-1)}{2}$ input dissimilarities, respectively.

The non-metric nature of the dissimilarity measures used in the test data sets brings our results closer to contemporary applications of LS-MDS techniques. The cost of this is, however, a limited choice for criteria in selecting terms for the truncated stress function. Sophisticated geometrical considerations cannot be easily applied to arbitrary non-metric dissimilarity measures. Consequently, terms for the truncated stress function with non−metric dissimilarities are often selected employing graph constructs, e.g. k-nearest and k-farthest graphs or Minimal Spanning Trees, or are selected at random [2, 23]. In our initial study on approximate MDS [2] we investigated selection of terms for the truncated stress function based on k-nearest and k-farthest neighbor graphs as well as random selection that sample distribution of dissimilarities. Results obtained in this study demonstrated that in the context of LS-MDS random selection performs much better. In particular, this selection scheme led to significantly smaller increase in the stress value in approximate embeddings, relative to the reference embeddings. Given these results, we adopted random selection of terms in this work and did not pursue experiments for the infe-

rior alternative. Algorithm 1 contains pseudo-code describing the random selection of terms.

6 RESULTS AND DISCUSSION

To assess the impact of MDS approximation on the resultant embeddings we employed Procrustes Analysis (Section 4) and fitted each approximate embedding to its corresponding reference embedding. We then calculated N-RMSD values (Equation (9)) between the fitted embeddings (Figure 1). For each reference and approximate embedding we also calculated the raw stress value according to Equation (1), i.e. including all $\frac{n\cdot(n-1)}{2}$ summation terms and thereby accounting for all input pattern dissimilarities. These numbers were used to assess the increase in stress value in approximate embeddings relative to their corresponding reference embeddings (Figure 2). Finally, for each embedding we calculated Pearson correlation coefficient between input pattern dissimilarities and the corresponding distances between points in the embedding (Table 1).

Data set	**Subset of pattern dissimilarities in the truncated stress function**					
	5.0 %	7.4 %	9.8 %	12.2 %	14.5 %	N^2
ncRNA Data	0.9479	0.9478	0.9480	0.9481	0.9481	0.9486
Gene Data	0.7469	0.7510	0.7527	0.7537	0.7546	0.7583
Letter Data	0.9073	0.9075	0.9078	0.9081	0.9084	0.9095

Table 1. Pearson correlation coefficients between distances of points in the LS-MDS embeddings and the corresponding pairwise pattern dissimilarities

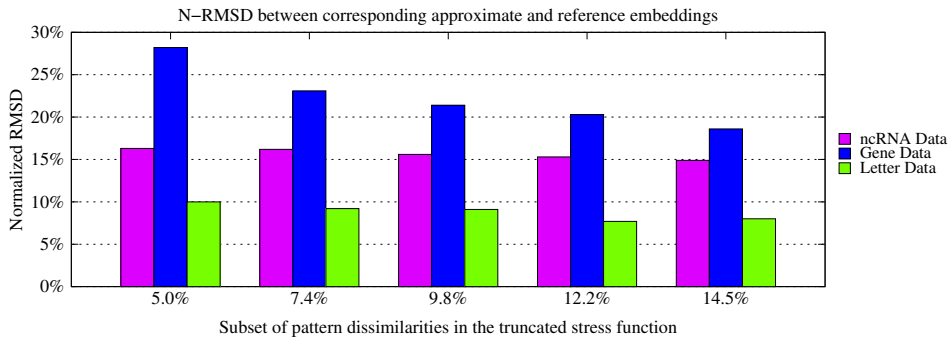


Fig. 1. Normalized Root Mean Square Distance between approximate embeddings and their corresponding reference embeddings

In our experiments approximate embeddings employing less than 10 % of input pattern dissimilarities have stress values that exceed the stress values of correspond-

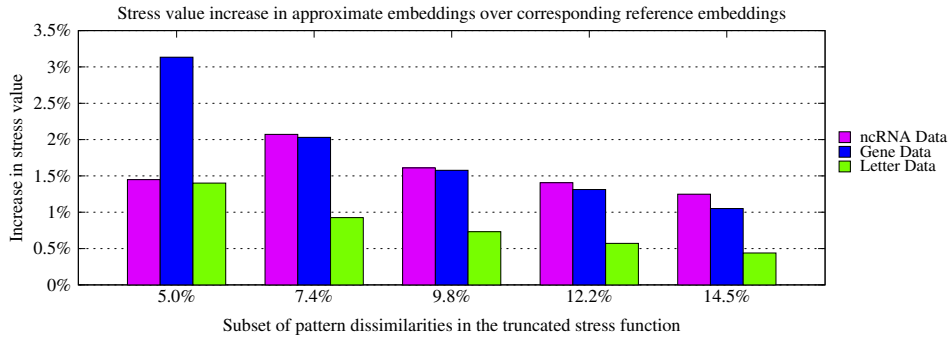


Fig. 2. Increase in stress value in approximate embeddings relative to their corresponding reference embeddings

ing reference embeddings by only a few percent. The largest increase, about 3 %, is for the Gene Data over approximately 5 % of input pattern dissimilarities. Note that even with 5 % of input dissimilarities, each pattern, and thus also each point in the embedding, is bounded by between approximately 700 (Gene Data) and 1 600 (ncRNA Data) randomly sampled dissimilarities. This may explain why stress values remain well minimized. Correlation coefficients obtained in the experiments also do not reveal large differences between approximate and reference embeddings. We observe that correlation depends mainly on the data set, with Gene Data appearing to be difficult to embed in $\mathbb{R}^3$, and the two other test data sets exhibiting better embedding. However, correlation coefficients in approximate embeddings are only slightly worse than in the corresponding reference embeddings.

Results of Procrustes Analysis reveal more substantial differences between approximate and reference embeddings. These differences are largest for the Gene Data, where four out of the five approximate embeddings exhibit N-RMSD to the reference embedding in excess of 20 % and the fifth one exhibits N-RMSD close to 20 %. In other words, RMSD in these cases is close to, or exceeds, one fifth of the root mean square distance of points from the embeddings' center. The other data set with large N-RMSD is the ncRNA Data. In this case N-RMSD remains slightly above 15 %. In Letter Data agreement between approximate embeddings and the reference embedding is better, with N-RMSD less than 10 %. For Gene Data and Letter Data we also observe that N-RMSD clearly decreases with the increasing number of pattern dissimilarities accounted for in the truncated stress function. Similar trend, however substantially weaker, can be seen for ncRNA Data. Differences between approximate embeddings and the corresponding reference embeddings can not, therefore, be explained away by fluctuations of MDS results around close minima of the multimodal stress function, that would be natural in a stochastic optimization process. The fact that N-RMSD decreases with increasing number of pattern dissimilarities in the truncated stress function suggests that approximate MDS explores configurations of points that are not explored when exact stress func-

tion is minimized. It is also worth noting that impact of MDS approximation is largest in the data set that, as revealed by the correlation coefficients, is significantly more difficult to embed in $\mathbb{R}^3$ than the other two data sets, namely in Gene Data. In more general terms, approximation in MDS has substantially larger impact on the two data sets described by dissimilarity measures that are not distance metrics, namely Gene Data and ncRNA Data, than on the Euclidean Letter Data.

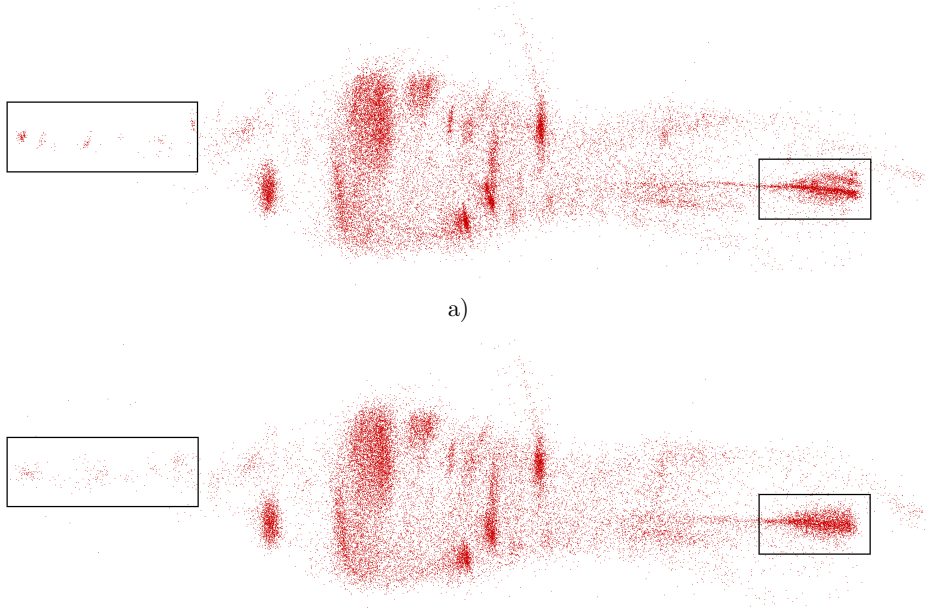
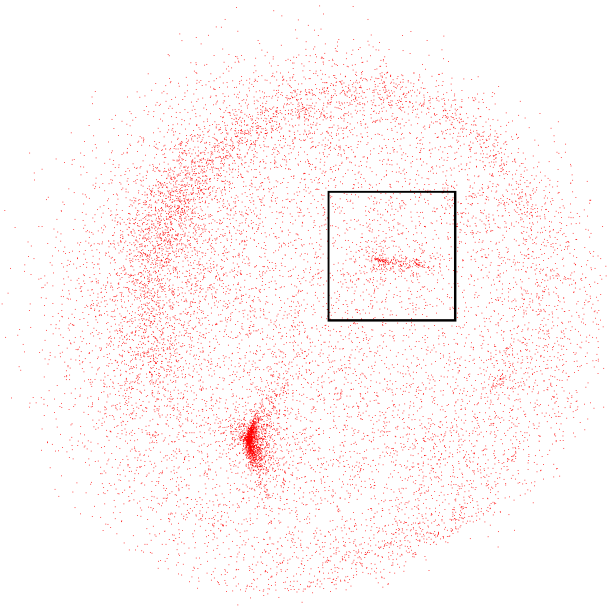
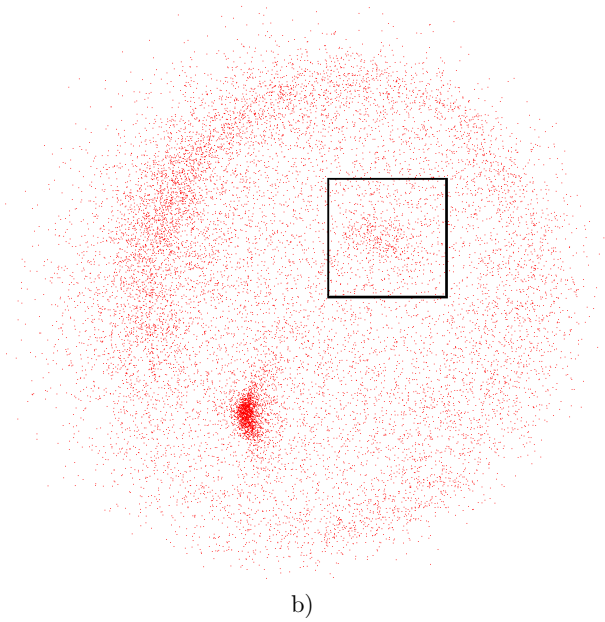


Fig. 3. a) Reference ncRNA Data embedding. b) Approximate ncRNA Data embedding constructed with 5 % of input pattern dissimilarities, fitted with Procrustes Analysis to its corresponding reference embedding. In regions marked by rectangles, approximate embedding displays a different clustering structure than the reference embedding

How the differences revealed by Procrustes Analysis translate into the structure visible on embeddings can be seen in Figures 3, 4 and 5. Therein we report approximate embeddings constructed with 5 % of input pattern dissimilarities, fitted to their corresponding reference embeddings. The broad structures of the data sets are similar in the approximate and the corresponding reference embeddings. However, at least in Gene Data and ncRNA Data approximate embeddings reveal less fine details than their corresponding reference embeddings. For example, on the reference Gene Data embedding (Figure 4 a)) we observe two distinct clusters on a spherically distributed cloud of patterns. However, on the corresponding approximate embedding (Figure 4 b)) only one of these clusters is apparent whereas the other one is disrupted. Similarly, at the right end of the reference ncRNA Data embedding

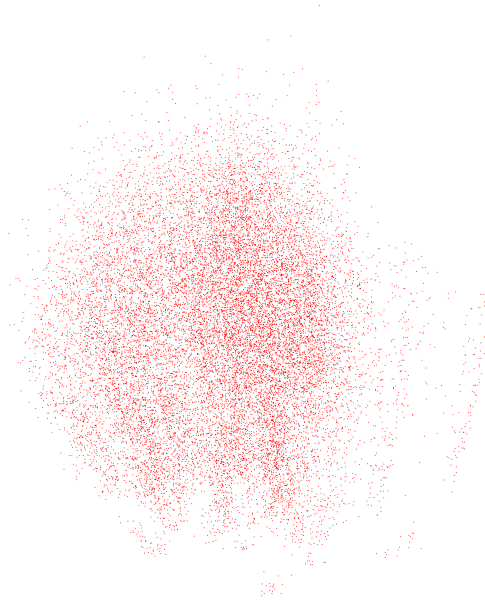


a)

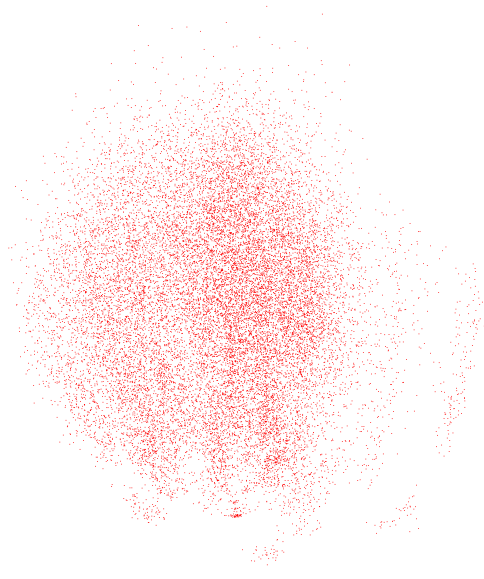


b)

Fig. 4. a) Reference Gene Data embedding. b) Approximate Gene Data embedding constructed with 5 % of input pattern dissimilarities, fitted with Procrustes Analysis to its corresponding reference embedding. Reference embedding displays two clusters, one of which, marked by a rectangle, is disrupted in the approximate embedding



a)



b)

Fig. 5. a) Reference Letter Data embedding. b) Approximate Letter Data embedding constructed with 5 % of input pattern dissimilarities, fitted with Procrustes Analysis to its corresponding reference embedding. Both embeddings display similar data set structure

(Figure 3 a)) we observe a dense tail made of two clusters. On the corresponding approximate embedding (Figure 3 b)) this tail appears to form only one, more noisy cluster. Also, at the left end of the reference embedding we observe a string of fine, point-like clusters. On the approximate embedding these clusters are disrupted and resemble sparsely distributed groups of patterns. In general, approximate embeddings on Figures 4 b) and 3 b) appear more noisy than their corresponding reference embeddings. In Letter Data, approximate embedding (Figure 5 b)) appears more faithful to the reference embedding (Figure 5 a)). Both embeddings display similar structure, with patterns forming layered shells.

7 CONCLUSIONS

Thus far most studies of approximate MDS focused on two indices of approximate embedding quality. First of these indices was correlation coefficient between pattern dissimilarities and the corresponding distances in the embedding. Second of the quality indices was increase in stress value in approximate embedding relative to the reference embedding. However, results of direct comparison by Procrustes Analysis between approximate and reference embeddings, that we presented in this work, revealed a more complex picture than that arising from analysis of these two indices. In particular, Procrustes Analysis revealed significant differences between approximate and reference embeddings in cases where increases in stress value and differences in correlation coefficients were small. Differences between approximate and reference embeddings were prominent in data sets described by dissimilarity measures that are not proper distance metric, and especially so in the data set that was difficult to embed in $\mathbb{R}^3$. Our results indicate that studies of approximate MDS algorithms should not be limited to assessment of correlation between distances and dissimilarities or increase in stress value, but should also report other quality measures. This additional measures may include, e.g., indices such as the Congruence Coefficient [4, Chapter 19.7] mentioned in Section 1, or may be derived following the approach used in this work, i.e. by quantifying geometric differences after fitting reference and approximate embeddings with Procrustes Analysis.

Acknowledgements

The National Human Genome Research Institute modENCODE Consortium is kindly acknowledged for making available transcriptome data used in this work. The UCI Machine Learning Repository is kindly acknowledged for distributing one of the test data sets used in this work. This work was partly funded by the Polish Ministry of Science and Higher Education grant No. N N519 443039.

A ALGORITHMS

Algorithm 1: Select terms (pattern pairs) for a truncated stress function

input : X – set of patterns
n – requested number of terms per pattern
output : S – set of terms for the truncated stress function
notation: `permute(A)` – random permutation of elements in A

for x ∈ X **do**
  S[x] ← ∅
  Y = `permute`(X \ {x})
  for $i \leftarrow 1$ **to** n **do**
    y ← Y[i]
    if x ∉ S[y] **then** add y to S[x]
return S

Algorithm 2: Fit set of points P to the set of points Q with Procrustes Analysis

input : P – first set of points
Q – second set of points, equinumerous with P
output : O – set of points from P fitted with PA to the set of points Q
notation: `transpose(X)` – transposition of matrix X
`mmul(X, Y)` – matrix multiplication of matrix X and matrix Y
`SVD(X)` – singular value decomposition of matrix X

```
c ← [0,0,0]
n ← 0
for p ∈ P do
    c ← c + p
    n ← n + 1
for p ∈ P do p ← p − c/n
c ← [0,0,0]
for p ∈ Q do c ← c + p
for p ∈ Q do p ← p − c/n
```

Construct matrices X and Y from points in P and Q, respectively.

```
/* Each raw stores coordinates of one point                    */
M ← mmul(transpose(Y), X)
U, L, V ← SVD(M)
R ← mmul(transpose(V), transpose(U))
O ← mmul(X, R)
sigma ← 0
for i ← 0 to n do
    sigma ← sigma +X[i, 1]∗X[i, 1]+X[i, 2]∗X[i, 2]+X[i, 3]∗X[i, 3]
M ← mmul(mmul(R, transpose(Y)), X)
sigma ← (M[1,1] + M[2,2] + M[3,3]) /sigma
for p ∈ O do p ← p ∗ sigma
return O
```

Algorithm 3: DPD-based implementation of LS-MDS

input : X – set of patterns
D – set of dissimilarities between patterns from X
S – terms in the truncated stress function, selected with Algorithm 1
g, rc – dissipative forces coef. and cut-off distance for DPD forces
dt – time step for DPD
output : P – set of points in $\mathbb{R}^3$ representing patterns in X
notation: `sqrt(z)` – square root of z
p[x] – point from P representing pattern x in $\mathbb{R}^3$
V[a], dV[a] – velocity and change in velocity of point a

```
set initial positions of points in P /* e.g. randomly                    */
T ← initial DPD temperature /* e.g. temperature that gives avg.
   velocity equal to mean dissimilarity per time unit                    */
for x ∈ X do
   a ← p[x]
   V[a] ← 0, dV[a] ← 0
repeat
   for x ∈ X do
      a ← p[x]
      V[a] ← V[a]+0.5*dV[a]
      p[x] ← p[x]+V[a]*dt
   fc ← CForces(X, D, S, P)
   fd ← DForces(X, S, P, V, g, rc)
   fr ← RForces(X, S, P, T, g, rc)
   for x ∈ X do
      a ← p[x]
      dV[a] ← fc[a]*dt +fd[a]*dt +fr[a]*sqrt(dt)
      V[a] ← V[a]+0.5*dV[a]
   fd ← DForces(X, S, P, V, g, rc)
   for x ∈ X do  dV[a] ← fc[a]*dt +fd[a]*dt +fr[a]*sqrt(dt)
   decrease temperature T by a small fraction
until not (T ≈ 0 and particle velocities are negligible)
return P = {p[x] : x ∈ X}
```

Function CForces(X, D, S, P) calculate conservative forces for Algorithm 3

input : X – set of patterns
D – set of dissimilarities between patterns from X
S – terms in the truncated stress function, selected with Algorithm 1
P – set of points in $\mathbb{R}^3$ representing patterns in X
output : F– conservative forces acting on points from P
notation: `len(q)` – length of vector q
D[x, y] – dissimilarity between pattern x and pattern y
p[x] – point from P representing pattern x in $\mathbb{R}^3$

```
for x ∈ X do F[p[x]] ← 0
for x ∈ X do
    a ← p[x]
    for y ∈ S[x] do
        b ← p[y]
        r ← a − b
        F[a] ← F[a]−2.0∗(len(r)-D[x, y])∗r/len(r)
        F[b] ← F[b]+2.0∗(len(r)-D[x, y])∗r/len(r)
return F
```

Function DForces(X, S, P, V, g, rc) calculate dissipative forces for Algorithm 3

input : X – set of patterns
S – terms in the truncated stress function, selected with Algorithm 1
P – set of points in $\mathbb{R}^3$ representing patterns in X
V – velocities of points from P
g, rc – dissipative forces coef. and cut-off distance for DPD forces
output : F– dissipative forces acting on points from P
notation: `len(q)` – length of vector q
`dot(q, u)` – dot product of vector q and vector u
p[x] – point from P representing pattern x in $\mathbb{R}^3$

```
for x ∈ X do F[p[x]] ← 0
for x ∈ X do
    a ← p[x]
    for y ∈ S[x] do
        b ← p[y]
        r ← a − b
        if len(r) < rc then
            force = g ∗ (1 − len(r)/rc) ∗ (1 − len(r)/rc)
            force = force ∗dot(V[a]−V[b], r/len(r))∗r/len(r)
            F[a] ← F[a]−force
            F[b] ← F[b]+force
return F
```

Function RForces(X, S, P, T, g, rc) calculate random forces for Algorithm 3

input : X – set of patterns
S – terms in the truncated stress function, selected with Algorithm 1
P – set of points in $\mathbb{R}^3$ representing patterns in X
T – current DPD temperature
g, rc – dissipative forces coef. and cut-off distance for DPD forces

output : F– random forces acting on points from P

notation: `sqrt(z)` – square root of z
`len(q)` – length of vector q
`nrand()` – generate random number from N(0, 1) distribution
p[x] – point from P representing pattern x in $\mathbb{R}^3$

```
for x ∈ X do F[p[x]] ← 0
for x ∈ X do
    a ← p[x]
    for y ∈ S[x] do
        b ← p[y]
        r ← a − b
        if len(r) < rc then
            force = sqrt(2 ∗ g ∗ T)∗(1 − len(r)/rc)
            force = force ∗nrand()∗r/len(r)
            F[a] ← F[a]+force
            F[b] ← F[b]−force
return F
```

REFERENCES

[1] Andrecut, M.: Molecular Dynamics Multidimensional Scaling. Physics Letters A, Vol. 373, 2009, No. 23-24, pp. 2001–2006.

[2] Arodz, T.—Boryczko, K.—Dzwinel, W.—Kurdziel, M.—Yuen, D. A.: Visual Exploration of Multidimensional Feature Space of Biological Data. In: IEEE Visualization 2005 – VIS '05, IEEE Computer Society, Minneapolis, MN, USA, October 2005, p. 90, Poster session & abstract.

[3] Biswas, G.—Jain, A. K.—Dubes, R. C.: Evaluation of Projection Algorithms. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 3, 1981, No. 6, pp. 701–708.

[4] Borg, I.—Groenen, P. J. F.: Modern Multidimensional Scaling: Theory and Applications. Springer Verlag, NY, USA 1997.

[5] Celniker, S. E.—Dillon, L. A. K.—Gerstein, M. B.—Gunsalus, K. C.—Henikoff, S.—Karpen, G. H. et al.: Unlocking the Secrets of the Genome. Nature, Vol. 459, 2009, pp. 927–930.

[6] Chang, C. L.—Lee, R. C. T.: A Heuristic Relaxation Method for Nonlinear Mapping in Cluster Analysis. IEEE Transactions on Systems, Man and Cybernetics, Vol. 3, 1973, No. 2, pp. 197–200.

[7] Cox, T. F.—Cox, M. A. A.: Multidimensional Scaling. CRC Press, Boca Raton, FL, USA. 2001.

[8] Dzwinel, W.: How to Make Sammon's Mapping Useful for Multidimensional Data Structures Analysis. Pattern Recognition, Vol. 27, 1994, No. 7, pp. 949–959.

[9] Dzwinel, W.—Blasiak, J.: Method of Particles in Visual Clustering of Multidimensional and Large Data Sets. Future Generation Computers Systems, Vol. 15, 1999, pp. 365–379.

[10] Espaol, P.—Warren, P.: Statistical Mechanics of Dissipative Particle Dynamics. Europhysics Letters, Vol. 30, 1995, No. 4, pp. 191–196.

[11] Frank, A.—Asuncion, A.: UCI Machine Learning Repository. University of California, Irvine, School of Information and Computer Sciences 2010, `http://archive.ics.uci.edu/ml`.

[12] Frey, P. W.—Slate, D. J.: Letter Recognition Using Hollandstyle Adaptive Classifiers. Machine Learning, Vol. 6, 1991, No. 2, pp. 161–182.

[13] Girard, R. A.—Cliff, N.: A Monte Carlo Evaluation of Interactive Multidimensional Scaling. Psychometrika, Vol. 41, 1976, No. 1, pp. 43–64.

[14] Graveley, B. R.—Brooks, A. N.—Carlson, J. W.—Duff, M. O.—Landolin, J. M.—Yang, L. et al.: The Developmental Transcriptome of Drosophila Melanogaster. Nature, Vol. 471, 2011, pp. 473–479.

[15] Green, R. S.—Bentler, P. M.: Improving the Efficiency and Effectiveness of Interactively Selected Mds Data Designs. Psychometrika, Vol. 44, 1979, No. 1, pp. 115–119.

[16] Griffiths-Jones, S.—Bateman, A.—Marshall, M.—Khanna, A.—Eddy, S. R.: RFAM: An RNA Family Database. Nucleic Acids Research, Vol. 31, 2003, No. 1, pp. 439–441.

[17] Griffiths-Jones, S.—Moxon, S.—Marshall, M.—Khanna, A.—Eddy, S. R.—Bateman, A.: RFAM: Annotating Noncoding RNAs in Complete Genomes. Nucleic Acids Research, Vol. 33, 2005, pp. D121–D124.

[18] Hoogerbrugge, P. J.—Koelman, J. M. V. A.: Simulating Microscopic Hydrodynamic Phenomena with Dissipative Particle Dynamics. Europhysics Letters, Vol. 19, 1992, No. 3, pp. 155–160.

[19] Lee, R. C. T.—Slagle, J. R.—Blum, H.: A Triangulation Method for the Sequential Mapping of Points from n-Space to Two-Space. IEEE Transactions on Computers, Vol. C-26, 1977, No. 3, pp. 288–292.

[20] Mathar, R.—Zilinskas, A.: On Global Optimization in Two-Dimensional Scaling. Acta Applicandae Mathematicae, Vol. 33, 1993, No. 1, pp. 109–118.

[21] Needleman, S. B.—Wunsch, C. D.: A General Method Applicable to the Search for Similarities in the Amino Acid Sequence of Two Proteins. Journal of Molecular Biology, Vol. 48, 1970, No. 3, pp. 443–453.

[22] Oten, R.—de Figueiredo, R. J. P.: Topological Dimensionality Determination and Dimensionality Reduction Based on Minimum Spanning Trees. In: Proceedings

of the 1998 IEEE International Symposium on Circuits and Systems, ISCAS '98, Monterey, CA, USA, June 1998, Vol. 3, pp. 366–369.

[23] Pawliczek, P.—Dzwinel, W.: Visual Analysis of Multidimensional Data Using Fast MDS Algorithm. In: Proceedings of SPIE, Vol. 6937 Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments, December 2007.

[24] Pevsner, J.: Bioinformatics and Functional Genomics. Second edition, Wiley Blackwell 2009.

[25] Sammon, J. W.: A Non-Linear Mapping for Data Structure Analysis. IEEE Transactions on Computers, Vol. C-18, 1969, No. 5, pp. 401–409.

[26] Trapnell, C.—Williams, B. A.—Pertea, G.—Mortazavi, A.—Kwan, G.—van Baren, M. J.—Salzberg, S. L.—Wold, B. J.—Pachter, L.: Transcript Assembly and Quantification by RNASeq Reveals Unannotated Transcripts and Isoform Switching During Cell Differentiation. Nature Biotechnology, Vol. 28, 2010, No. 5, pp. 511–515.

[27] Young, F. W.—Cliff, N.: Interactive Scaling with Individual Subjects. Psychometrika, Vol. 37, 1972, No. 4, pp. 385–415.

Marcin Kurdziel received his Ph. D. degree in computer science from the AGH University of Science and Technology in 2010. His research interests focus on unsupervised pattern recognition methods, in particular data clustering and visualization algorithms, and their implementations on large, massively parallel computers. Currently, he is an Assistant Professor at the Department of Computer Science, Faculty of Computer Science, Electronics and Telecommunications, AGH University of Science and Technology.

Krzysztof Boryczko received his Ph. D. degree in computer science in 1992 and D. Sc. degree in computer science in 2004, both from the AGH University of Science and Technology, where he is now an Associate Professor at the Department of Computer Science, Faculty of Computer Science, Electronics and Telecommunications. His research interests focus on large-scale simulations with particle methods and on implementations of particle methods on Graphical Processing Units. He is also interested in scientific visualization, feature extraction and clustering algorithms for analysis of simulation data.

Witold Dzwinel holds Full Professor position in the Department of Computer Science at AGH University of Science and Technology in Krakow. As a postdoctoral computer science fellow and visiting scientist he conducted work on diagnostic system of IBR-2 nuclear reactor at the Joint Institute of Nuclear Research in Dubna (Russia). Holding research-scholar position at the University of Minnesota (Minnesota Supercomputer Institute) he published many papers about applications of dissipative particle dynamics for mesoscopic HPC simulations of complex fluids, including blood dynamics in micro vessels. Now, his main research activities are focused on discrete particle methods and so called complex automata paradigm developed for simulation of microscopic and macroscopic phenomena. He is the author and co-author of about 175 papers in computer science and computational sciences.