Multidimensional Scaling by Deterministic Annealing with Iterative Majorization algorithm

Seung-Hee Bae, Judy Qiu, Geoffrey C. Fox
Pervasive Technology Institute, School of Informatics and Computing
Indiana University
Bloomington, Indiana 47408
Email: sebae, xqiu, gcf@indiana.edu

***Abstract*—Multidimensional Scaling (MDS) is a dimension reduction method for information visualization, which is set up as a non-linear optimization problem. It is applicable to many data intensive scientific problems including studies of DNA sequences but tends to get trapped in local minima. Deterministic Annealing (DA) has been applied to many optimization problems to avoid local minima. We apply DA approach to MDS problem in this paper and show that our proposed DA approach improves the mapping quality and shows high reliability in a variety of experimental results. Further its execution time is similar to that of the un-annealed approach. We use different data sets for comparing the proposed DA approach with both a well known algorithm called SMACOF and a MDS with distance smoothing method which aims to avoid local optima. Our proposed DA method outperforms SMACOF algorithm and the distance smoothing MDS algorithm in terms of the mapping quality and shows much less sensitivity with respect to initial configurations and stopping condition. We also investigate various temperature cooling parameters for our deterministic annealing method within an exponential cooling scheme.**

I. Introduction

The recent explosion of publicly available biology gene sequences, chemical compounds, and various scientific data offers an unprecedented opportunity for data mining. Among many data mining algorithms, dimension reduction is a useful tool for information visualization of such high-dimensional data to make data analysis feasible for such vast volume and high-dimensional scientific data. It facilitates the investigation of unknown structures of high dimensional data in three (or two) dimensional visualization.

Among the known dimension reduction algorithms, such as Principal Component Analysis (PCA), Multidimensional Scaling (MDS) [1], [2], Generative Topographic Mapping (GTM) [3], and Self-Organizing Maps (SOM) [4], to name a few, multidimensional scaling has been extensively studied and used in various real application area, such as biology [5], stock market analysis [6], computational chemistry [7], and breast cancer diagnosis [8].

In contrast to other algorithms, like PCA, GTM, and SOM, which generally construct a low dimensional configuration based on vector information, MDS aims to construct a new mapping in target dimension on the basis of pairwise proximity (typically dissimilarity or distance) information so that it does not require feature vector information of the real application data to acquire lower dimensional mapping of the given data. Hence, MDS is really useful for data visualization of a certain type of data which is impossible to represent by feature vectors but have pairwise dissimilarity, such as biological sequence data. MDS, of course, is also applicable to the data represented by feature vectors as well.

MDS is a non-linear optimization approach constructing a lower dimensional mapping of high dimensional data with respect to the given proximity information based on objective functions, namely STRESS [9] or SSTRESS [10]. Below equations are the definition of STRESS (1) and SSTRESS (2):

$$\sigma(X) = \sum_{i<j\leq N} w_{ij}(d_{ij}(X) - \delta_{ij})^2 \quad (1)$$

$$\sigma^2(X) = \sum_{i<j\leq N} w_{ij}[(d_{ij}(X))^2 - (\delta_{ij})^2]^2 \quad (2)$$

where $1 \leq i < j \leq N$, w_{ij} is a weight value ($w_{ij} \geq 0$), $d_{ij}(X)$ is a Euclidean distance between mapping results of x_i and x_j, and δ_{ij} is the given original pairwise dissimilarity value between x_i and x_j. SSTRESS is adopted by ALSCAL algorithm (Alternating Least Squares Scaling) [10], and using *squared* Euclidean distances results in simple computation. A more natural choice could be STRESS which is used by SMACOF [11] and Sammon's mapping [12].

Due to non-linear property of MDS problem, an optimization method called iterative majorization is used to solve MDS problem [11]. However, iterative majorization method is a type of Expectation-Maximization (EM) approach [13], and it is well understood that EM method suffers from local minima problem although EM method is widely applied to many optimization problem. In order to overcome local minima issue, we have applied a robust optimization method called Deterministic Annealing (DA) [14], [15] to the MDS problem.

A key feature of the DA algorithm is to endeavour to find global optimum without trapping local optima in *deterministic way* [14] instead of stochastical random approach, which results in long running time, as in Simulated Annealing (SA) [16]. In a physics language, DA uses *mean field approximation* to the statistical physics integrals.

In Section II, we discuss briefly the background and related work. Then, the proposed DA MDS algorithm is explained in Section III. Section IV illustrates performance of the proposed DA MDS algorithm compared to other MDS algorithms followed by conclusion in Section V.

II. Background and Related Work

A. Avoiding Local Optima in MDS

SMACOF is a quite useful algorithm, since it will monotonically decrease the STRESS criterion [11]. However, the well-known problem of the gradient descent approach is to be trapped in a local minima due to its hill-climbing approach. Stochastic optimization approaches, such as simulated annealing (SA) [16] and genetic algorithms (GA) [17], have been used for many optimization problems including MDS problem [18], [19] in order to avoid local optima, but stochastic algorithms are well-known to suffer from long running time due to their Monte Carlo approach. In addition to stochastic algorithms, distance smoothing [20] and tunneling method [21] for MDS problem were proposed to avoid local optima in a deterministic way.

Recently, Ingram et al. introduced a multilevel algorithm called Glimmer [22] which is based on force-based MDS algorithm with restriction, relaxation, and interpolation operators. Glimmer shows less sensitivity to initial configurations than GPU-SF subsystem, which is used in Glimmer [22], due to the multilevel nature. In Glimmer's paper [22], however, SMACOF algorithm shows better mapping quality than Glimmer. Also, the main purpose of Glimmer is to achieve speed up with less cost of quality degrade rather than mapping quality improvement. In contrast, this paper focuses on optimization method which improves mapping quality in deterministic approach. Therefore, we will compare the proposed algorithm to other optimization algorithms, i.e. SMACOF and Distance Smoothing method, in Section IV.

B. Deterministic Annealing Approach (DA)

Since the simulated annealing (SA) was introduced by Kirkpatrick et al. [16], people widely accepted SA and other stochastic maximum entropy approach to solve optimization problems for the purpose of finding global optimum instead of hill-climbing deterministic approaches. SA is a Metropolis algorithm [23], which accepts not only the better proposed solution but even the worse proposed solution than the previous solution based on a certain probability which is related to *computational temperature* (T). Also, it is known that Metropolis algorithm converges to an equilibrium probability distribution known as *Gibbs probability distribution*. If we denote $\mathscr{H}(X)$ as the energy (or cost) function and $\mathscr{F}$ as a *free energy*, then Gibbs distribution density is following:

$$P^G(X) = \exp\left(-\frac{1}{T}(\mathscr{H}(X) - \mathscr{F})\right), \tag{3}$$

$$\mathscr{F} = -T\log\int \exp\left(-\frac{1}{T}\mathscr{H}(X)\right)\mathrm{d}X. \tag{4}$$

and the *free energy ($\mathscr{F}$)*, which is a suggested objective function of SA, is minimized by the Gibbs probability density P^G. Also, free energy $\mathscr{F}$ can be written as following:

$$\mathscr{F}_P = < \mathscr{H} >_P - T\mathscr{S}(P) \tag{5}$$

$$\equiv \int P(X)\mathscr{H}(X)\mathrm{d}X + T\int P(X)\log P(X)\mathrm{d}X \tag{6}$$

where $< H >_P$ represents the *expected energy* and $\mathscr{S}(P)$ denotes *entropy* of the system with probability density P. Here, T is used as a Lagrange multiplier to control the expected energy. With high temperature, the problem space is dominated by the *entropy* term which make the problem space become smooth so it is easy to move further. As temperature is getting cooler, however, the problem space is gradually revealed as the landscape of the original cost function which limits the movement on the problem space. To avoid trapped in local optima, people usually start with high temperature and slowly decrease temperature in the process of finding solution.

SA relies on random sampling with Monte Carlo method to estimate the expected solution, e.g. expected mapping in target dimension for MDS problem, so that it suffers from long running time. Deterministic annealing (DA) [14], [15] can be thought of as an approximation algorithm of SA which tries to keep the merit of SA. DA [14], [15] method actually tries to calculate the expected solution exactly or approximately with respect to the Gibbs distribution as an amendment of SA's long running time, while it follows computational annealing process using Eq. (5), which T decreases from high to low.

DA method is used for many optimization problems, including clustering [14], [15], pairwise clustering [24], and MDS [25], to name a few. Since it is intractable to calculate $\mathscr{F}$ in Eq. (4) exactly, an approximation technique called *mean field approximation* is used for solving MDS problem by DA in [25], in that Gibbs distribution $P^G(X)$ is approximated by a factorized distribution with density

$$P^0(X|\Theta) = \prod_{i=1}^{N} q_i(x_i|\Theta_i). \tag{7}$$

where Θ_i is a vector of mean field parameter of x_i and $q_i(x_i|\Theta_i)$ is a factor serves as a marginal distribution model of the coordinates of x_i. To optimize parameters Θ_i, Klock and Buhmann [25] minimized Kullback-Leibler (KL) divergence between the approximated density $P^0(X)$ and the Gibbs density $P^G(X)$ through EM algorithm [13]. Although, DA-MDS [25] shows the general approach of applying DA to MDS problem, it is not clearly explained how to solve MDS. Therefore, we will introduce the alternative way to utilize DA method to MDS problem in Section III.

III. Deterministic Annealing SMACOF

If we use STRESS (1) objective function as an expected energy (cost) function in Eq. (5), then we can define $\mathscr{H}_{MDS}$ and $\mathscr{H}_0$ as following:

$$\mathscr{H}_{MDS} = \sum_{i<j\leq N}^{N} w_{ij}(d_{ij}(X) - \delta_{ij})^2 \tag{8}$$

$$\mathscr{H}_0 = \sum_{i=1}^{N} \frac{(x_i - \mu_i)^2}{2} \tag{9}$$

where $\mathscr{H}_0$ corresponds to an energy function based on a simple multivariate Gaussian distribution and μ_i represents the average of the multivariate Gaussian distribution of i-th point ($i = 1, \ldots, N$) in target dimension (L-dimension). Also, we define P^0 and $\mathscr{F}_0$ as following:

$$P^0(X) = \exp\left(-\frac{1}{T}(\mathscr{H}_0 - \mathscr{F}_0)\right), \tag{10}$$

$$\mathscr{F}_0 = -T \log \int \exp\left(-\frac{1}{T}\mathscr{H}_0\right) \mathrm{d}X = -T \log (2\pi T)^{L/2} \tag{11}$$

We need to minimize $\mathscr{F}_{MDS}(P^0) = < \mathscr{H}_{MDS} - \mathscr{H}_0 > + \mathscr{F}_0(P^0)$ with respect to μ_i. Since $- < \mathscr{H}_0 > + \mathscr{F}_0(P^0)$ is independent to μ_i, only $< \mathscr{H}_{MDS} >$ part is necessary to be minimized with regard to μ_i. If we apply $< x_i x_i > = \mu_i \mu_i + TL$ to $< \mathscr{H}_{MDS} >$, then $< \mathscr{H}_{MDS} >$ can be deployed as following:

$$< \mathscr{H}_{MDS} > = \sum_{i<j\leq N}^{N} w_{ij}(< \|x_i - x_j\| > - \delta_{ij})^2 \tag{12}$$

$$= \sum_{i<j\leq N}^{N} w_{ij}(\sqrt{\|\mu_i - \mu_j\|^2 + 2TL} - \delta_{ij})^2 \tag{13}$$

$$\approx \sum_{i<j\leq N}^{N} w_{ij}(\|\mu_i - \mu_j\| + \sqrt{2TL} - \delta_{ij})^2 \tag{14}$$

where $\|a\|$ is Norm$_2$ of a vector a. Eq. (13) can be approximated to Eq. (14), since the bigger T, the smaller $\|\mu_i - \mu_j\|$ and the smaller T, the bigger $\|\mu_i - \mu_j\|$.

In [25], Klock and Buhmann tried to find an approximation of $P^G(X)$ with mean field factorization method by minimizing Kullback-Leibler (KL) divergence using EM approach. The found parameters by minimizing KL-divergence between $P^G(X)$ and $P^0(X)$ using EM approach are essentially the expected mapping in target dimension under current problem space with computational temperature (T).

In contrast, we try to find expected mapping, which minimize $\mathscr{F}_{MDS}(P^0)$, directly with new objective function ($\hat{\sigma}$) which is applied DA approach to MDS problem space with computational temperature T by well-known EM-like MDS solution, called SMACOF [11]. Therefore, as T varies, the problem space also varies, and SMACOF algorithm is used to find expected mapping under each problem space at a corresponding T. In order to apply SMACOF algorithm to DA method, we substitute the original STRESS equation (1) with Eq. (14). Note that μ_i and μ_j are the expected mappings we are looking for, so we can consider $\|\mu_i - \mu_j\|$ as $d_{ij}(X_T)$, where X_T represents the embedding results in L-dimension at T and d_{ij} means the Euclidean distance between mappings of point i and j. Thus, the new STRESS ($\hat{\sigma}$) is following:

$$\hat{\sigma} = \sum_{i<j\leq N}^{N} w_{ij}(d_{ij}(X_T) + \sqrt{2TL} - \delta_{ij})^2 \tag{15}$$

$$= \sum_{i<j\leq N}^{N} w_{ij}(d_{ij}(X_T) - \hat{\delta}_{ij})^2 \tag{16}$$

with $\hat{\delta}_{ij}$ defined as following:

$$\hat{\delta}_{ij} = \begin{cases} \delta_{ij} - \sqrt{2TL} & \text{if } \delta_{ij} > \sqrt{2TL} \\ 0 & \text{otherwise} \end{cases} \tag{17}$$

Algorithm 1 DA-SMACOF algorithm

```
Input: $\Delta$ and $\alpha$
Compute $T_0$ and $\widehat{\Delta}_0 = [\hat{\delta}_{ij}]$ based on Eq. (18).
Generate random initial mapping $X_0$.
$k \Leftarrow 0$;
while $T_k \geq T_{min}$ do
    $X_{k+1}$ = output of SMACOF with $\widehat{\Delta}_k$ and $X_k$. $X_k$ is used
    for initial mapping of the current SMACOF running.
    Cool down computational Temperature $T_{k+1} = \alpha T_k$
    Update $\widehat{\Delta}_{k+1}$ w.r.t. $T_{k+1}$.
    $k \Leftarrow k + 1$;
end while

$X$ = output of SMACOF based on $\Delta$ and $X_k$.
return: $X$;
```

In addition, T is a lagrange multiplier so it can be thought of as $T = \hat{T}^2$, then $\sqrt{2TL} = \hat{T}\sqrt{2L}$ and we will use T instead of $\hat{T}$ for the simple notation. Thus, Eq. (17) can be written as following:

$$\hat{\delta}_{ij} = \begin{cases} \delta_{ij} - T\sqrt{2L} & \text{if } \delta_{ij} > T\sqrt{2L} \\ 0 & \text{otherwise.} \end{cases} \tag{18}$$

Now, we can apply SMACOF to find expected mapping with respect to new STRESS (16) which is based on computational temperature T. The MDS problem space could be smoother with higher T than with lower T, since T represents the portion of entropy to the free energy $\mathscr{F}$ as in Eq. (5). Generally, DA approach starts with high T and gets cool down T as time goes on, like physical annealing process. However, if starting computational temperature (T_0) is very high which results in all $\hat{\delta}_{ij}$ become ZERO, then all points will be mapped at origin (O). Once all mappings are at the origin, then the Guttman transform is unable to construct other mapping except the mapping of all at the origin, since Guttman transform does multiplication iteratively with previous mapping to calculate current mapping. Thus, we need to calculate T_0 which makes at least one $\hat{\delta}_{ij}$ is bigger than ZERO, so that at least one of the points is not located at O.

With computed T_0, the $\widehat{\Delta}_0 = [\hat{\delta}_{ij}]$ can be calculated, and we are able to run SMACOF algorithm with respect to Eq. (16). After new mapping generated with T_0 by SMACOF algorithm, say X_0, then we will cool down the temperature in exponential way, like $T_{k+1} = \alpha T_k$, and keep doing above steps until T becomes too small. Finally, we set $T = 0$ and then run SMACOF by using the latest mapping as an initial mapping with respect to original STRESS (1). We will assume the uniform weight $\forall w_{ij} = 1$ where $0 < i < j \leq N$ and it is easy to change to non-uniform weight. The proposed deterministic annealing SMACOF algorithm, called DA-SMACOF, is illustrated in Alg. 1.

IV. Experimental Analysis

For the performance analysis of the proposed deterministic annealing MDS algorithm, called **DA-SMACOF**, we would

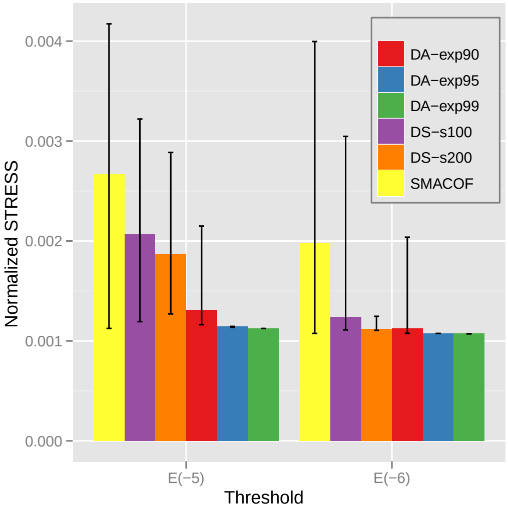


Fig. 1. The normalized STRESS comparison of **iris** data mapping results in 2D space. Bar graph illustrates the average of 50 runs with random initialization and the corresponding error bar represents the minimum and maximum of the normalized STRESS value of SMACOF, MDS-DistSmooth with different smoothing steps ($s = 100$ and $s = 200$) (**DS-s100** and **-s200** hereafter for short), and DA-SMACOF with different cooling parameters ($\alpha = 0.9$, 0.95, and 0.99) (**DA-exp90,-exp95**, and **-exp99** hereafter for short). The x-axis is the threshold value for the stopping condition of iterations (10^{-5} and 10^{-6}).

like to examine DA-SMACOF's capability of avoiding local optima in terms of objective function value (normalized STRESS in (19)) and the sensitivity of initial configuration by comparing with original EM-like SMACOF algorithm and MDS by Distance Smoothing [20] (**MDS-DistSmooth** hereafter for short) which tries to find global optimum mapping. We have tested above algorithms with many different data sets, including well-known benchmarking data sets from UCI machine learning repository[1] as well as some real application data, such as chemical compound data and biological sequence data, in order to evaluate the proposed DA-SMACOF.

Since MDS-DistSmooth requires the number of smoothing steps which affects to the the degree of smoothness and cooling parameter (α) of computational temperature (T) affects the annealing procedure in DA-SMACOF, we examine two different number of smoothing step numbers ($s = 100$ and $s = 200$) for MDS-DistSmooth and three different cooling parameters ($\alpha = 0.9$, 0.95, and 0.99) for DA-SMACOF algorithm, as well. (Hereafter, MDS-DistSmooth with smoothing steps $s = 100$ and $s = 200$ are described by **DS-s100** and **DS-s200** respectively, and DA-SMACOF with temperature cooling parameters $\alpha = 0.9$, 0.95, and 0.99 are represented by **DA-exp90**, **DA-exp95**, and **DA-exp99**, correspondingly.) We also examine two different thresholds for the stopping condition, i.e. $\varepsilon = 10^{-5}$ and $\varepsilon = 10^{-6}$, for tested algorithms.

To compare mapping quality of the proposed DA-SMACOF with SMACOF and MDS-DistSmooth, we measure the normalized STRESS which substitutes w_{ij} in (1) for $1/\sum_{i<j}\delta_{ij}^2$ like following:

$$\sigma(X) = \sum_{i<j\leq N} \frac{1}{\sum_{i<j}\delta_{ij}^2}(d_{ij}(X) - \delta_{ij})^2 \qquad (19)$$

in that the normalized STRESS value denotes the relative portion of the squared distance error rates of the given data set without regard to scale of δ_{ij}.

A. Iris Data

The iris data[2] set is very well-known benchmarking data set for data mining and pattern recognition communities. Each data item consists of four different real values (a.k.a. *4D real-valued vector*) and each value represents an attribute of each instance, such as length or width of sepal (or petal). There are three different classes (*Iris Setosa, Iris Versicolour, and Iris Virginica*) in the iris data set and each class contains 50 instances, so there are total 150 instances in the iris data set. It is known that one class is linearly separable from the other two, but the remaining two are not linearly seperable from each other.

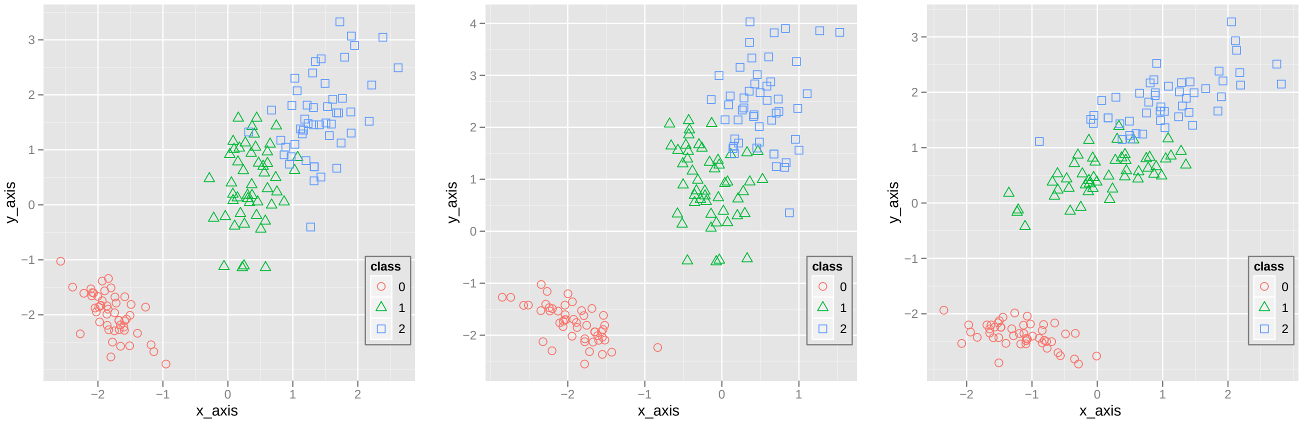
In Fig. 1, The mapping quality of the constructed configurations of iris data by SMACOF, MDS-DistSmooth, and DA-SMACOF is compared by the average, the minimun, and the maximum of normalized STRESS values among 50 random-initial runnings. The proposed DA-SMACOF with all tested cooling parameters, including quite fast cooling parameter ($\alpha = 0.9$), outperforms SMACOF and MDS-DistSmooth in Fig. 1 except DS-s200 case with $\varepsilon = 10^{-6}$. Although DS-s200 with $\varepsilon = 10^{-6}$ is comparable to DA-SMACOF results, DS-s200 takes almost 3 times longer than DA-exp95 with $\varepsilon = 10^{-6}$ which shows more consistent result than DS-s200.

Numerically, DA-exp95 improves mapping quality 57.1% and 45.8% of SMACOF results in terms of the average of STRESS values with $\varepsilon = 10^{-5}$ and $\varepsilon = 10^{-6}$, correspondingly. DA-exp95 shows better mapping quality about 43.6% and 13.2% than even DS-s100, which is the algorithm to find global optimum, with $\varepsilon = 10^{-5}$ and $\varepsilon = 10^{-6}$.

In terms of sensitivity to initial configuration, SMACOF shows very divergent STRESS value distribution for both $\varepsilon = 10^{-5}$ and $\varepsilon = 10^{-6}$ cases in Fig. 1, which means that SMACOF is quite sensitive to the initial configuration (a.k.a. easy to be trapped in *local optima*). In addition, MDS-DistSmooth also shows relatively high sensitivity to the initial configuration with the iris data set although the degree of divergence is less than SMACOF algorithm. In contrast to other algorithms, the proposed DA-SMACOF shows high consistency without regard to initial setting which we could interprete as it is likely to avoid local optima. Since it is well-known that the slow cooling temperature is necessary to avoid local optima, we expected that DA-exp90 might be trapped in local optima as shown in Fig. 1. Although DA-exp90 cases show some variations, DA-exp90 cases still show much better results than SMACOF and MDS-DistSmooth except DS-s200

[1] UCI Machine Learning Repository, http://archive.ics.uci.edu/ml/

[2] Iris Data set, http://archive.ics.uci.edu/mi/datasets/Iris



(a) Iris by EM Median

(b) Iris by DS Median

(c) Iris by DA Median

Fig. 2. The 2D median output mappings of iris data with SMACOF (a), DS-s100 (b), and DA-exp95 (c), whose threshold value for the stopping condition is 10^{-5}. Final normalized STRESS values of (a), (b), and (c) are 0.00264628, 0.00208246, and 0.00114387, correspondingly.

with $\varepsilon = 10^{-6}$ case. In fact, the standard deviation of DA-exp95 with $\varepsilon = 10^{-5}$ result is 1.08×10^{-6} and DA-exp99 with $\varepsilon = 10^{-5}$ and DA-exp95/exp99 with $\varepsilon = 10^{-6}$ shows ZERO standard deviation in terms of STRESS values of 50 random-initial runs. We can also note that difference of DA-SMACOF results between $\varepsilon = 10^{-5}$ and $\varepsilon = 10^{-6}$ is negligible with the iris data, whereas the average of SMACOF and MDS-DistSmooth (DS-s100) with $\varepsilon = 10^{-5}$ is about 35.5% and 81.6% worse than corresponding $\varepsilon = 10^{-6}$ results.

Fig. 2 illustrates the difference of actual mapping outputs among SMACOF, MDS-DistSmooth, and DA-SMACOF. All of the mappings are the median results of stopping condition with 10^{-5} threshold value. The mapping in Fig. 2a is the 2D mapping result of median valued SMACOF, and Fig. 2b represents the median result of MDS-DistSmooth. Three mappings in Fig. 2 are a little bit different to one another, and clearer structure differentiation between class 1 and class 2 is shown at Fig. 2c which is the median STRESS valued result of DA-SMACOF.

B. *Chemical Compound Data*

The second data set is chemical compound data with 333 instances represented by 155 dimensional real-valued vectors. For the given original dissimilarity (Δ), we measure Euclidean distance of each instance pairs based on feature vectors as well as the iris data set.

Fig. 3 depicts the average mapping quality of 50 runs for 333 chemical compounds mapping results with regard to different experimental setups as in the above. For the chemical compound data set, all experimented results of the proposed DA-SMACOF (DA-exp90, DA-exp95, and DA-exp99) show the superior performance to SMACOF and MDS-DistSmooth with both $\varepsilon = 10^{-5}$ and $\varepsilon = 10^{-6}$ stopping conditions. In detail, the average STRESS of SMACOF is 2.50 and 1.88 times larger than corresponding DA-SMACOF results with $\varepsilon = 10^{-5}$ and $\varepsilon = 10^{-6}$ threshold, and the average STRESS of MDS-DistSmooth shows 2.66 and 1.57 times larger than DA-SMACOF algorithm with $\varepsilon = 10^{-5}$ and $\varepsilon = 10^{-6}$. Furthermore, the minimum STRESS values of SMACOF and MDS-DistSmooth experiments are larger than the average of all DA-SMACOF results. One interesting phenomena in Fig. 3 is that the MDS-DistSmooth shows worse performance in average with $\varepsilon = 10^{-5}$ stopping condition than SMACOF and DS-s100 shows better than DS-s200.

As similar as in Fig. 1, all SMACOF and MDS-DistSmooth experimental results show higher divergence in terms of STRESS values in Fig. 3 than the proposed DA-SMACOF. On the other hand, DA-SMACOF shows much less divergence with respect to STRESS values, especially DA-exp99 case.

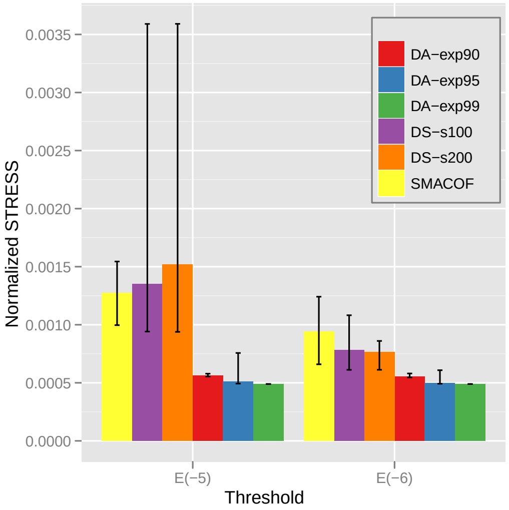


Fig. 3. The normalized STRESS comparison of **chemical compound** data mapping results in 2D space. Bar graph illustrates the average of 50 runs with random initialization and the corresponding error bar represents the minimum and maximum of the normalized STRESS value of SMACOF, DS-s100 and -s200, and DA-exp90,DA-exp95, and DA-exp99. The x-axis is the threshold value for the stopping condition of iterations (10^{-5} and 10^{-6}).

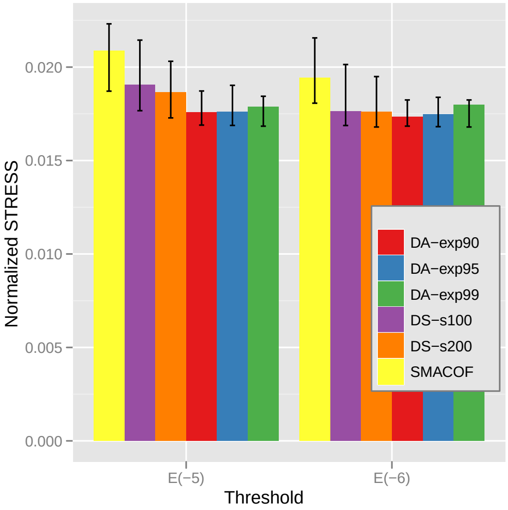


Fig. 4. The normalized STRESS comparison of **breast cancer** data mapping results in 2D space. Bar graph illustrates the average of 50 runs with random initialization and the corresponding error bar represents the minimum and maximum of the normalized STRESS value of SMACOF, DS-s100 and -s200, and DA-exp90,DA-exp95, and DA-exp99. The x-axis is the threshold value for the stopping condition of iterations (10^{-5} and 10^{-6}).

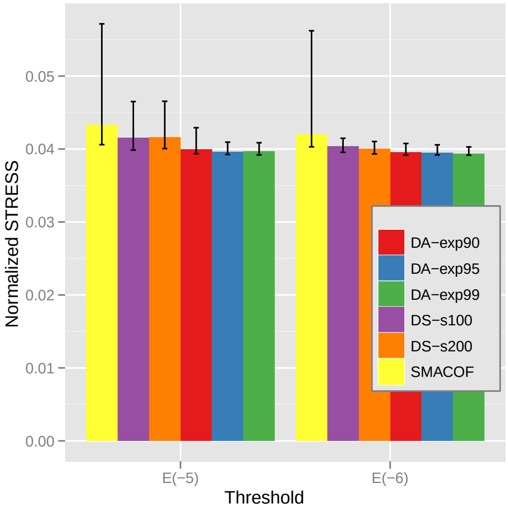


Fig. 5. The normalized STRESS comparison of **yeast** data mapping results in 2D space. Bar graph illustrates the average of 50 runs with random initialization and the corresponding error bar represents the minimum and maximum of the normalized STRESS value of SMACOF, DS-s100 and -s200, and DA-exp90,DA-exp95, and DA-exp99. The x-axis is the threshold value for the stopping condition of iterations (10^{-5} and 10^{-6}).

For the comparison between different cooling parameters, as we expected, DA-exp90 shows some divergence and a little bit higher average than DA-exp95 and DA-exp99, but much less average than SMACOF. Interestingly, DA-exp95 shows relatively larger divergence than DA-exp90 due to outliers. However, those outliers of DA-exp95 happened rarely among 50 runs and the most of DA-exp95 running results are similar as minimum value of corresponding test.

C. Cancer Data

The cancer data[3] set is well-known data set found in UCI Machine Learning Repository. Each data item consists of 11 columns and the first and the last column represent *id-number* and *class* correspondingly, and the remaining 9 columns are attribute values described in integer from 1 to 10. There are two different classes (*benign and malignant*) in the cancer data set. Originally, it contains 699 data items, we use only 683 data points which have every attribute values, since 16 items have some missing information.

Fig. 4 depicts the average mapping quality of 50 runs for 683 cancer data mapping results with regard to different experimental setups as in the above. For the cancer data set, all experimented results of the proposed DA-SMACOF (DA-exp90, DA-exp95, and DA-exp99) show the superior performance to SMACOF and MDS-DistSmooth with $\varepsilon = 10^{-5}$, and better than SMACOF and comparable to MDS-DistSmooth with $\varepsilon = 10^{-6}$ stopping conditions. Interestingly, DA-exp99 case shows worse results than DA-exp95 and DA-exp90 results, although DA-exp99 case find most minimum mapping in terms of normalized STRESS value. In detail, the average STRESS of SMACOF is 18.6% and 11.3% worse than corresponding DA-SMACOF results with $\varepsilon = 10^{-5}$ and $\varepsilon = 10^{-6}$ threshold, and the average STRESS of MDS-DistSmooth shows 8.3% worse than DA-SMACOF with $\varepsilon = 10^{-5}$ and comparable to DA-SMACOF with$\varepsilon = 10^{-6}$.

Although DA-SMACOF experimental results show some divergence in terms of STRESS values in Fig. 4, in contrast to Fig. 1 and Fig. 3, DA-SMACOF experimental results show less divergence of STRESS values than SMACOF and MDS-DistSmooth in Fig. 4.

D. Yeast Data

The yeast data[4] set is composed of 1484 entities and each entity is represented by 8 real-value attributes in addition to the sequence name and class labels.

The normalized STRESS comparison of the yeast mapping results by different algorithms is illustrated in Fig. 5 in terms of the average mapping quality of 50 runs for 1484 points mapping. DA-SMACOF shows better performance than the other two algorithms in this experiments as same as the above experiments. SMACOF keep showing much higher divergence rather than DA-SMACOF with both stopping condition cases. Also, MDS-DistSmooth shows divergent STRESS distribution with $\varepsilon = 10^{-5}$ stopping condition, but not with $\varepsilon = 10^{-6}$ stopping condition. DA-SMACOF shows quite stable results except DA-exp90 case with $\varepsilon = 10^{-5}$ stopping condition, as well as better solution. In terms of best mapping (a.k.a. minimum normalized STRESS value), all DA-SMACOF experiments obtain better solution than SMACOF and MDS-DistSmooth, and even best result of SMACOF is worse than the average of the proposed DA approach.

[3] Breast Cancer Data set, http://archive.ics.uci.edu/mi/datasets/Breast+Cancer+Wisconsin+(Original)

[4] Yeast Data set, http://archive.ics.uci.edu/mi/datasets/Yeast

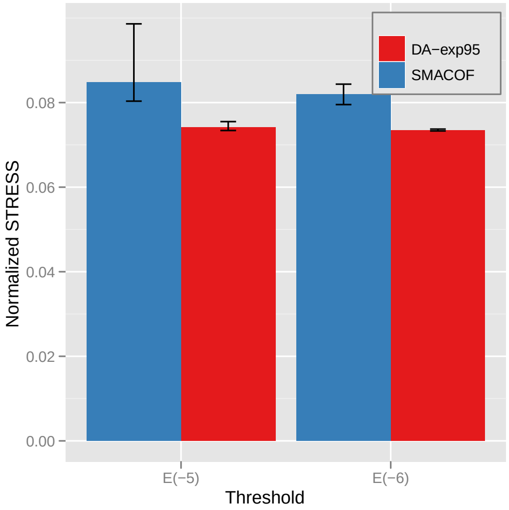


Fig. 6. The normalized STRESS comparison of **metagenomics sequence** data mapping results in 2D space. Bar graph illustrates the average of 10 runs with random initialization and the corresponding error bar of the normalized STRESS value of SMACOF and DA-exp95. The x-axis is the threshold value for the stopping condition of iterations (10^{-5} and 10^{-6}).

E. Metagenomics Data

The last data we used for evaluation of DA-SMACOF algorithm is a biological sequence data with respect to the metagenomics study. Although it is hard to present a biological sequence in a feature vector, people can calculate a dissimilarity value between two different sequences by some pairwise sequence alignment algorithms, like Smith Waterman - Gotoh (*SW-G*) algorithm [26], [27] which we used in this paper.

In contrast to smaller data size as in the above tests, metagenomics data set contains 30,000 points (sequences). Since MDS algorithms requires $\mathcal{O}(N^2)$ main memory, we have to use much larger amount of memory than main memory in a single node for running with 30,000 points. Thus, we use distributed memory version of SMACOF algorithm [28] to run with this metagenomics data.

Fig. 6 is the comparison between the average of 10 random initial runs of DA-SMACOF (DA-exp95) and SMACOF with metagenomics data set. As same as other results, SMACOF shows a tendency to be trapped in local optima by depicting some variation and larger STRESS values, and even the minimum values are bigger than any results of DA-exp95. DA-exp95 results are actually 12.6% and 10.4% improved compared to SMACOF in average with $\varepsilon = 10^{-5}$ and $\varepsilon = 10^{-6}$, correspondingly. As shown in Fig. 6, all of the DA-exp95 results are very similar to each other, especially when stopping condition is $\varepsilon = 10^{-6}$. In contrast to DA-SMACOF, SMACOF shows larger variation in both stopping conditions in Fig. 6.

F. Running Time Comparison

From Section IV-A to Section IV-E, we have been analyzed the mapping quality by comparing DA-SMACOF with SMACOF and MDS-DistSmooth, and DA-SMACOF outperforms SMACOF in all test cases and outperforms or is comparable to MDS-DistSmooth. In this section, we would like to compare the running time among those algorithms. Fig. 7 describes the average running time of each test case for SMACOF, MDS-DistSmooth, and DA-exp95 with 50 runs for the tested data. In order to make a distinct landscape in Fig. 7, we plot the quadrupled runtime results of **iris** and **cancer** data.

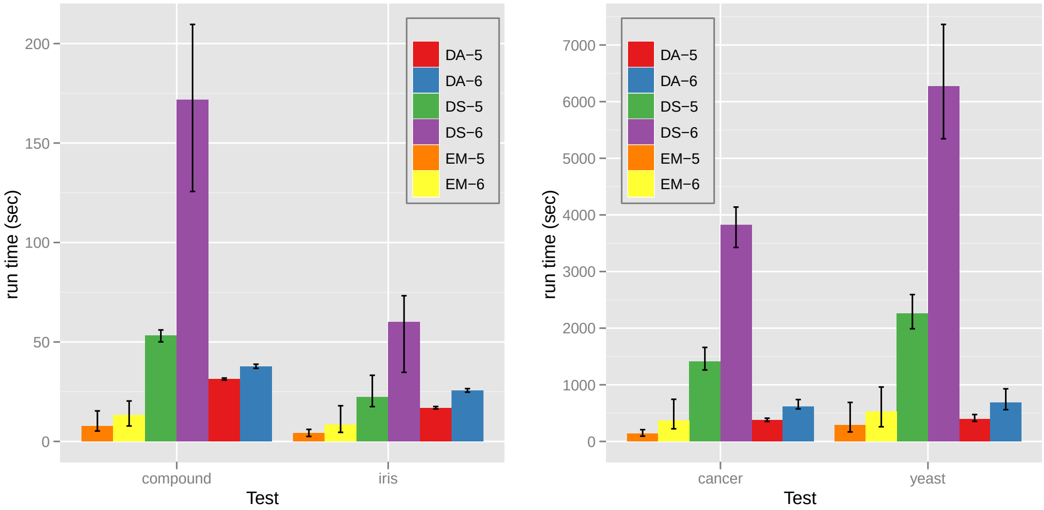
In Fig. 7, all runnings are performed in sequential computing with AMD Opteron 8356 2.3GHz CPU and 16GB memory. As shown in Fig. 7, DA-SMACOF is a few times slower than SMACOF but faster than MDS-DistSmooth in all test cases. In detail, DA-SMACOF takes 2.8 to 4.2 times longer than SMACOF but 1.3 to 4.6 times shorter than MDS-DistSmooth with **iris** and **compound** data set in Fig. 7a. Also, DA-SMACOF takes 1.3 to 2.8 times longer than SMACOF but 3.7 to 9.1 times shorter than MDS-DistSmooth with **cancer** and **yeast** data set in Fig. 7b. For **metagenomics** data set with 30,000 points, we tested with 128 way parallelism by MPI version of SMACOF and DA-SMACOF [28] and DA-SMACOF takes only 1.36 and 1.12 times longer than SMACOF in average. Actually, several SMACOF runs take longer than DA-SMACOF running times, although DA-SMACOF obtains better and reliable mapping results in Fig. 6. Interestingly, the less deviation is shown by DA-SMACOF than other compared algorithms in all cases with respect to running time as well as STRESS values.

V. Conclusion

In this paper, the authors have proposed an MDS solution with deterministic annealing (DA) approach, which utilize SMACOF algorithm in each cooling step. the proposed DA approach outperforms SMACOF and MDS-DistSmooth algorithms with respect to the mapping qualities with several different real data sets. Furthermore, DA-SMACOF exhibits the high consistency due to less sensitivity to the initial configurations, in contrast to SMACOF and MDS-DistSmooth which show high sensitivity to both the initial configurations and stopping condition. With the benefit of DA method to avoid local optima, the proposed DA approach uses slightly longer or comparable running time to SMACOF and shorter running time than MDS-DistSmooth. In addition, we also investigate different computational temperature cooling parameters in exponential cooling scheme and it turns out that it shows some deviation of mapping results when we use faster cooling parameter than necessary (like DA-exp90 case in this paper) but DA-exp90 shows still better than or comparable to the compared algorithms in our experiments. Also, DA-exp95 results are very similar to or even better than DA-exp99 results although DA-exp95 takes shorter time than DA-exp99 case. In future work, we will integrate these ideas with the interpolation technology described in [29] to give a robust approach to dimension reduction of large datasets that scales like $\mathcal{O}(N)$ rather $\mathcal{O}(N^2)$ of general MDS methods.

Acknowledgment

Authors appreciate Mina Rho and Haixu Tang for providing the metagenomics data.



(a) Small Data Runtime

(b) Large Data Runtime

Fig. 7. The average running time comparison between SMACOF, MDS-DistSmooth ($s = 100$), and DA-SMACOF (DA-exp95) for 2D mappings with tested data sets. The error bar represents the minimum and maximum running time. **EM-5/EM-6** represents SMACOF with $10^{-5}/10^{-6}$ threshold, and **DS-5/DS-6** and **DA-5/DA-6** represents the runtime results of MDS-DistSmooth and DA-SMACOF, correspondingly, in the same way.

References

[1] J. B. Kruskal and M. Wish, *Multidimensional Scaling*. Beverly Hills, CA, U.S.A.: Sage Publications Inc., 1978.

[2] I. Borg and P. J. Groenen, *Modern Multidimensional Scaling: Theory and Applications*. New York, NY, U.S.A.: Springer, 2005.

[3] C. Bishop, M. Svensén, and C. Williams, "GTM: The generative topographic mapping," *Neural computation*, vol. 10, no. 1, pp. 215–234, 1998.

[4] T. Kohonen, "The self-organizing map," *Neurocomputing*, vol. 21, no. 1-3, pp. 1–6, 1998.

[5] J. Tzeng, H. Lu, and W. Li, "Multidimensional scaling for large genomic data sets," *BMC bioinformatics*, vol. 9, no. 1, p. 179, 2008.

[6] P. Groenen and P. Franses, "Visualizing time-varying correlations across stock markets," *Journal of Empirical Finance*, vol. 7, no. 2, pp. 155–172, 2000.

[7] D. Agrafiotis, D. Rassokhin, and V. Lobanov, "Multidimensional scaling and visualization of large molecular similarity tables," *Journal of Computational Chemistry*, vol. 22, no. 5, pp. 488–500, 2001.

[8] H. Lahdesmaki, X. Hao, B. Sun, L. Hu, O. Yli-Harja, I. Shmulevich, and W. Zhang, "Distinguishing key biological pathways between primary breast cancers and their lymph node metastases by gene function-based clustering analysis," *International journal of oncology*, vol. 24, no. 6, pp. 1589–1596, 2004.

[9] J. B. Kruskal, "Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis," *Psychometrika*, vol. 29, no. 1, pp. 1–27, 1964.

[10] Y. Takane, F. W. Young, and J. de Leeuw, "Nonmetric individual differences multidimensional scaling: an alternating least squares method with optimal scaling features," *Psychometrika*, vol. 42, no. 1, pp. 7–67, 1977.

[11] J. de Leeuw, "Applications of convex analysis to multidimensional scaling," *Recent Developments in Statistics*, pp. 133–146, 1977.

[12] J. W. Sammon, "A nonlinear mapping for data structure analysis," *IEEE Transactions on Computers*, vol. 18, no. 5, pp. 401–409, 1969.

[13] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society. Series B*, pp. 1–38, 1977.

[14] K. Rose, E. Gurewitz, and G. C. Fox, "A deterministic annealing approach to clustering," *Pattern Recognition Letters*, vol. 11, no. 9, pp. 589–594, 1990.

[15] K. Rose, "Deterministic annealing for clustering, compression, classification, regression, and related optimization problems," *Proceedings of IEEE*, vol. 86, no. 11, pp. 2210–2239, 1998.

[16] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983.

[17] J. H. Holland, *Adaptation in natural and artificial systems*. Ann Arbor, MI: University of Michigan Press, 1975.

[18] M. Brusco, "A simulated annealing heuristic for unidimensional and multidimensional (city-block) scaling of symmetric proximity matrices," *Journal of Classification*, vol. 18, no. 1, pp. 3–33, 2001.

[19] R. Mathar and A. Žilinskas, "On global optimization in two-dimensional scaling," *Acta Applicandae Mathematicae: An International Survey Journal on Applying Mathematics and Mathematical Applications*, vol. 33, no. 1, pp. 109–118, 1993.

[20] P. Groenen, W. Heiser, and J. Meulman, "Global optimization in least-squares multidimensional scaling by distance smoothing," *Journal of classification*, vol. 16, no. 2, pp. 225–254, 1999.

[21] P. Groenen and W. Heiser, "The tunneling method for global optimization in multidimensional scaling," *Psychometrika*, vol. 61, no. 3, pp. 529–550, 1996.

[22] S. Ingram, T. Munzner, and M. Olano, "Glimmer: Multilevel MDS on the GPU," *IEEE Transactions on Visualization and Computer Graphics*, vol. 15, no. 2, pp. 249–261, 2009.

[23] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *The journal of chemical physics*, vol. 21, no. 6, pp. 1087–1092, 1953.

[24] T. Hofmann and J. M. Buhmann, "Pairwise data clustering by deterministic annealing," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, pp. 1–14, 1997.

[25] H. Klock and J. M. Buhmann, "Data visualization by multidimensional scaling: a deterministic annealing approach," *Pattern Recognition*, vol. 33, no. 4, pp. 651–669, 2000.

[26] T. F. Smith and M. S. Waterman, "Identification of common molecular subsequences," *Journal of molecular biology*, vol. 147, no. 1, pp. 195–197, 1981.

[27] O. Gotoh, "An improved algorithm for matching biological sequences," *Journal of Molecular Biology*, vol. 162, no. 3, pp. 705–708, 1982.

[28] J. Y. Choi, S.-H. Bae, X. Qiu, and G. Fox, "High performance dimension reduction and visualization for large high-dimensional data analysis," in *Proceedings of the 10th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID) 2010*, May 2010.

[29] S.-H. Bae, J. Y. Choi, X. Qiu, and G. Fox, "Dimension reduction and visualization of large high-dimensional data via interpolation," in *Proceedings of the ACM International Symposium on High Performance Distributed Computing (HPDC) 2010*, Chicago, Illinois, June 2010.