J Glob Optim (2007) 38:581–596
DOI 10.1007/s10898-006-9097-x

ORIGINAL ARTICLE

Two level minimization in multidimensional scaling

Antanas Žilinskas · Julius Žilinskas

Received: 14 July 2006 / Accepted: 19 September 2006 / Published online: 27 October 2006
© Springer Science+Business Media B.V. 2006

Abstract Multidimensional scaling with city block norm in embedding space is considered. Construction of the corresponding algorithm is reduced to minimization of a piecewise quadratic function. The two level algorithm is developed combining combinatorial minimization at upper level with local minimization at lower level. Results of experimental investigation of the efficiency of the proposed algorithm are presented as well as examples of its application to visualization of multidimensional data.

Keywords Multilevel optimization · Multidimensional scaling · Metaheuristics · Global optimization

1 Introduction

Multidimensional scaling (MDS) is a technique for analysis of multidimensional data widely usable in different applications (Borg and Groenen 1997, Cox and Cox 2001). Theoretical and algorithmic aspects of MDS are considered, by Borg and Groenen (1997), Cox and Cox (2001), Groenen (1993), De Leeuw and Heiser (1982), Mathar (1997) among others. Let us give a short formulation of the problem. The dissimilarity between pairs of n objects is given by the matrix $(\delta_{ij}), i, j = 1, \ldots, n$, and it is supposed that $\delta_{ij} = \delta_{ji}$. The points in an m-dimensional embedding space $x_i \in R^m, i = 1, \ldots, n$, should be found whose interpoint distances fit the given dissimilarities. Most frequently a two-dimensional ($m = 2$) embedding space is considered, for example, aiming to visualize the results of MDS. Different measures of accuracy of fit can be chosen defining different images of the considered set of objects. In the case the

A. Žilinskas (✉) · J. Žilinskas
Institute of Mathematics and Informatics, Akademijos 4, 08663, Vilnius, Lithuania
e-mail: antanasz@ktl.mii.lt

J. Žilinskas
e-mail: julius.zilinskas@mii.lt

objects are points in a high-dimensional vector space such images can be interpreted as different nonlinear projections of the set of points in high-dimensional space to an embedding space of lower dimensionality. The problem of construction of images of the considered objects is reduced to minimization of an accuracy of fit criterion, e.g. of the most frequently used least squares STRESS function

$$S(X) = \sum_{i<j} w_{ij}(d_{ij}(X) - \delta_{ij})^2, \tag{1}$$

where $X = (x_{11}, \ldots, x_{n1}, x_{12}, \ldots, x_{nm})^T$; $d_{ij}(X)$ denotes the distance between the points x_i and x_j; it is supposed that the weights are positive: $w_{ij} > 0, i, j = 1, \ldots, n$.

Since different distances $d_{ij}(X)$ can be defined, the formula (1) defines a class of accuracy criteria. To define a particular criterion a norm in R^m should be chosen implying the particular formula for calculating distances $d_{ij}(X)$. The most frequently used norm is Euclidean. However, MDS with other Minkowski norms in embedding space can be even more informative than MDS with Euclidean norm (Groenen et al. 1995). Results of MDS with different norms can be useful to grasp different properties of the considered objects. For example, the pictures in Fig. 2 presenting results of MDS with Euclidean and city block distances show different properties of a multidimensional hypercube.

In the present paper MDS algorithms based on STRESS criterion with city block norm in the embedding space are considered. Since the non-differentiability of (1) in this case cannot be ignored, MDS with city block distances is especially difficult. The minimization problem of (1) is high-dimensional: $X \in R^N$ where the number of variables is equal to $N = n \times m$. STRESS function can have many local minima. Therefore MDS is a difficult global optimization problem.

Global optimization methods are developed for various classes of multimodal problems (Törn and Žilinskas 1989, Horst et al. 1995). Different global optimization methods have been applied to MDS, e.g. tunneling method by Groenen (1993), evolutionary methods by Mathar and Žilinskas (1993), Groenen et al. (2000), Everett (2001), simulated annealing by Brusco (2001), Leng and Lau (2004), Klock and Buhmann (1999), D.C. algorithm by An and Tao (2001). In the present paper global minimization of (1) with city block norm is considered. Several two level methods are investigated where local minimization is a lower level task, and combinatorial optimization is an upper level task.

2 On differentiability of STRESS at local minimizer

Majority of publications on MDS consider STRESS with Euclidean distances $d_{ij}(X)$ which are special case of Minkowski distances

$$d_{ij}(X) = \left(\sum_{k=1}^{m} \left| x_{ik} - x_{jk} \right|^p \right)^{1/p},$$

with $p = 2$. However, recently also increased interest to the methods based on city block distances, i.e. Minkowski distances with $p = 1$,

$$d_{ij}(X) = \sum_{k=1}^{m} \left| x_{ik} - x_{jk} \right| \tag{2}$$

(see, e.g. Brusco 2001, Leng and Lau 2004). For a review of MDS with city-block distances we refer to Brusco (2001).

Many global optimization methods for minimization of (1) with Euclidean distances include auxiliary local minimization algorithms. Differentiability of an objective function at minimum point is an important factor for a proper choice of local minimization algorithm. The well known result by De Leeuw (1984) on differentiability of (1) with Euclidean distances at local minimizer is generalized for general Minkowski distances in (Groenen et al. 1995). However the latter result does not cover the case of city block distances, i.e. the case of Minkowski distances with $p = 1$.

Let X be a local minimizer of $S(\cdot)$. Then a directional derivative with respect to an arbitrary directional (unit) vector Y is not negative: $D_Y S(X) \geq 0$. Therefore the inequality

$$D_Y S(X) + D_{-Y} S(X) \geq 0, \tag{3}$$

holds for an arbitrary vector Y. The expression of $D_Y S(X)$,

$$D_Y S(X) = \sum_{i<j} 2w_{ij}(d_{ij}(X) - \delta_{ij}) \cdot D_Y d_{ij}(X), \tag{4}$$

includes $D_Y d_{ij}(X)$ whose compact expression can be obtained using the following formula

$$D_{Y_{ijk}} d_{ij}(X) = \begin{cases} y_{ik} - y_{jk}, & \text{if } x_{ik} - x_{jk} > 0, \\ -(y_{ik} - y_{jk}), & \text{if } x_{ik} - x_{jk} < 0, \\ |y_{ik} - y_{jk}|, & \text{if } x_{ik} - x_{jk} = 0, \end{cases} \tag{5}$$

where Y_{ijk} denotes a vector whose all components are equal to zero except those corresponding to x_{ik}, x_{jk}, $k = 1, \ldots, m$. Formula (5) can be written in the following shorter form

$$D_{Y_{ijk}} d_{ij}(X) = |y_{ik} - y_{jk}| \cdot \text{sign}(x_{ik} - x_{jk})(y_{ik} - y_{jk}), \tag{6}$$

where sign$(\cdot)$ denotes non-symmetric signum function: sign$(t) = 1$, for $t \geq 0$, and sign$(t) = -1$ for $t < 0$. Substitution of $D_Y d_{ij}(X)$ in (4) with its expression based on (6) gives the following formula

$$D_Y S(X) = \sum_{k=1}^{m} |y_{ik} - y_{jk}| \cdot \text{sign}((x_{ik} - x_{jk})(y_{ik} - y_{jk})). \tag{7}$$

From (3), (4), and (7) it follows the inequality

$$4 \sum_{k=1}^{m} \sum_{(i,j) \in Q_k} w_{ij}(d_{ij}(X) - \delta_{ij}) \cdot |y_{ik} - y_{jk}| \geq 0, \tag{8}$$

where $Q_k = \{(i,j) : x_{ik} = x_{jk}\}$.

Since the inequality (8) is not satisfied in a case $d_{ij}(X) = 0$, $d_{ij}(Y) > 0$ and $d_{rs}(Y) = 0$, $(rs) \neq (i,j)$, then at local minimum point X the inequalities $d_{ij}(X) > 0$ should hold for all $i \neq j$. The positiveness of distances $d_{ij}(X) > 0$ means that the points in embedding space (images of the considered objects) do not coincide. Our proof is similar to that in (Groenen et al. 1995), but some modification was needed since their formulae do not cover the case of city block distances.

The positiveness of distances between image points corresponding to a local minimizer of (1) does not imply differentiability of (1) at the minimizer. Such a conclusion distinguishes MDS version with the city block distances from all other MDS versions with Minkowski ($p > 1$) distances. On the other hand it does not prove the existence of cases with non-differentiable local minima. A simple example illustrating possibility of non differentiable local minimum is presented below.

Let us consider an example of two-dimensional scaling where data is the following

$$\delta_{12} = \delta_{14} = \delta_{23} = \delta_{34} = 1, \qquad \delta_{13} = \delta_{24} = 3 \tag{9}$$

and $w_{ij} = 1$. The set of vertices of the square centered at origin, and with side equal to 4/3 is a potential image of the considered objects. This image corresponds to the eight-dimensional ($n \times m = 8$) point $\bar{X}$ where

$$\bar{x}_{11} = \bar{x}_{21} = \bar{x}_{12} = \bar{x}_{42} = -\frac{2}{3}, \qquad \bar{x}_{31} = \bar{x}_{41} = \bar{x}_{22} = \bar{x}_{32} = \frac{2}{3}. \tag{10}$$

We will show that $\bar{X}$ is a local minimizer of $S(X)$. The directional derivative of $S(X)$ with respect to an arbitrary directional vector Y at the point $\bar{X}$ is equal to

$$D_Y S(\bar{X}) = 2(|y_{11} - y_{21}| + |y_{12} - y_{42}| + |y_{22} - y_{32}| + |y_{31} - y_{41}|)/3 \geq 0. \tag{11}$$

It is obvious that

$$D_Y S(\bar{X}) > 0 \tag{12}$$

unless all summands in (11) are equal to zero. In the latter case the directional vector should satisfy the following equalities

$$y_{11} = y_{21},\ y_{12} = y_{42}, \qquad y_{22} = y_{32},\ y_{31} = y_{41}, \tag{13}$$

implying differentiability of $S(\bar{X} + tY)$ with respect to t. The rather long initial expression of $\frac{\mathrm{d}^2}{\mathrm{d}t^2} S(\bar{X} + tY)$ using elementary algebra can be reduced to the following one

$$\begin{aligned}\frac{\mathrm{d}^2}{\mathrm{d}t^2} S(\bar{X} + tY) = 2[&(y_{12} - y_{22})^2 + (y_{11} - y_{31} + y_{12} - y_{32})^2 \\ &+ (y_{11} - y_{41})^2 + (y_{21} - y_{31})^2 \\ &+ (y_{21} - y_{41} + y_{22} - y_{42})^2 + (y_{32} - y_{42})^2],\end{aligned} \tag{14}$$

implying the inequality

$$\frac{\mathrm{d}^2}{\mathrm{d}t^2} S(\bar{X} + tY)|_{t=0} > 0 \tag{15}$$

for all directional vectors satisfying (13), unless all summands in (14) are equal to zero. Let Y satisfy equality $\frac{\mathrm{d}^2}{\mathrm{d}t^2} S(\bar{X} + tY) = 0$ and (13), then the components of Y also satisfy the equalities

$$y_{11} = y_{21} = y_{31} = y_{41}, \qquad y_{12} = y_{22} = y_{32} = y_{42}. \tag{16}$$

But $S(X)$ is invariant with respect to translation of coordinates implying that

$$S(\bar{X}) = S(\bar{X} + Y). \tag{17}$$

Therefore the inequalities (12) and (15) prove that $\bar{X}$ is a local minimizer of $S(X)$.

Because STRESS function can be non differentiable at local minimizer, application of local descent methods with high-convergence rate, e.g. of different versions of Newton method, seems questionable. In the subsequent sections we either apply search methods for STRESS minimization, or reformulate the problem introducing constraints.

3 Two level optimization of STRESS

We consider two-dimensional ($m = 2$) embedding space because our aim is to visualize multidimensional data for heuristic analysis. Using city block distances $d_{ij}(X)$, STRESS (see (1)) can be redefined as

$$s(X) = \sum_{i<j} \left(\left| x_{i1} - x_{j1} \right| + \left| x_{i2} - x_{j2} \right| - \delta_{ij} \right)^2 , \tag{18}$$

where weights w_{ij} are supposed equal to 1.

Let A_{gh} denotes a set such that

$$A_{gh} = \{X : x_{i1} \leq x_{j1} \quad \text{for } g_i < g_j,\ x_{i2} \leq x_{j2} \quad \text{for } h_i < h_j\}, \tag{19}$$

where g and h are permutations of $1, \ldots, n$.

For $X \in A_{gh}$ (18) can be rewritten in the following form

$$s(X) = \sum_{i<j} \left(\left(x_{i1} - x_{j1} \right) g_{ij} + \left(x_{i2} - x_{j2} \right) h_{ij} - \delta_{ij} \right)^2 ,$$
$$g_{ij} = \text{sign}(g_i - g_j), \qquad h_{ij} = \text{sign}(h_i - h_j). \tag{20}$$

Since function $s(X)$ is quadratic over polyhedron $X \in A_{gh}$ the minimization problem

$$\min_{X \in A_{gh}} s(X) \tag{21}$$

is a quadratic programing problem. The structure of the minimization problem (21) supposes a two level minimization algorithm: to solve a combinatorial problem at upper level, and to solve a quadratic programing problem (21) at lower level:

$$\min_{g,h} s\left(X\left(g, h\right)\right), \tag{22}$$

$$\text{where } X\left(g, h\right) = \arg \min_{X \in A_{gh}} s\left(X\right). \tag{23}$$

The upper level (22) objective function is defined over the set of (g, h) where g and h are permutations of $1, \ldots, n$. Properties of the objective function are not known, therefore optimal solution can not be found by means of an efficient algorithm with a guarantee. A reasonable alternative is a metaheuristic optimization, for example by means of evolutionary search.

Alternatively the function (18) can be minimized directly. Methods combining selection of starting points with local minimization can be also considered as implicit two level algorithms, where, e.g. a genetic algorithm performs combinatorial search in the space of basins of attraction of local minima, although this space is not explicitly defined. In the case of MDS with Euclidean distances an algorithm combining local descent and evolutionary search is proposed in (Mathar and Žilinskas 1993). Such an algorithm is shown to be most reliable of known MDS algorithms experimentally tested in (Groenen et al. 2000), and (Mathar 1996). Therefore it seems reasonable to investigate similar algorithm also for the case of city block distances. It is only necessary to replace a gradient based descent algorithm with a search algorithm.

The disadvantage of the two level optimization problem with quadratic problems at lower level is enormous number of potential solutions of a combinatorial problem at upper level. Although the quadratic programing problems can be solved easily, the upper level problem theoretically is intractable since the favorable properties of the objective function of the combinatorial problem are not known. The enormous number of potential solutions at upper level seems a bit artificial since solutions of a quadratic programing subproblem are not necessary local minimizers of the original problem.

From the theoretical point of view the disadvantage of the second version of two level algorithm for minimization of STRESS with city block distances is a difficult local minimization problem at lower level; the advantage is much smaller number of potential solutions at upper level.

Both versions of two level algorithms seem prospective candidates for practical applications of MDS based on city block distances. Implementations of both versions combine a metaheuristic algorithm for upper level problems with a local minimization at lower level. We compare these versions experimentally using artificially difficult and standard test problems.

4 Algorithms for lower level problems

For the first version of the two level algorithm the problem at lower level is solved using sophisticated local algorithm combining quadratic programming algorithm and the search algorithm by Powell presented in Press et al. (2002). The extended form of lower level quadratic programming problem (23) is presented below:

$$\min -d^T X + \frac{1}{2} X^T D X, \tag{24}$$

$$\text{s.t.} A_1 X = 0, \tag{25}$$

$$A_2 X \geq 0, \tag{26}$$

$$A_3 X \geq 0, \tag{27}$$

$$\text{where } D = \begin{pmatrix} n-1 & -1 & & \sum g_{1j}h_{1j} & -g_{12}h_{12} & \\ -1 & n-1 & \dots & -g_{12}h_{12} & \sum g_{2j}h_{2j} & \dots \\ & \vdots & & & \vdots & \\ \sum g_{1j}h_{1j} & -g_{12}h_{12} & & n-1 & -1 & \\ -g_{12}h_{12} & \sum g_{2j}h_{2j} & \dots & -1 & n-1 & \dots \\ & \vdots & & & \vdots & \end{pmatrix},$$

$$d = \begin{pmatrix} \sum g_{1j}\delta_{1j} \\ \sum g_{2j}\delta_{2j} \\ \vdots \\ \sum h_{1j}\delta_{1j} \\ \sum h_{2j}\delta_{2j} \\ \vdots \end{pmatrix},$$

$$A_1 = \begin{pmatrix} 1\,1\ldots 0\,0\ldots \\ 0\,0\ldots 1\,1\ldots \end{pmatrix},$$

$$A_{2ij} = \left\{\begin{array}{ll} 1, & \text{if } g_j = i+1 \\ -1, & \text{if } g_j = i \\ 0, & \text{otherwise} \end{array}\right| , \quad i = 1,\ldots,n-1, \; j = 1,\ldots,2n,$$

$$A_{3ij} = \left\{\begin{array}{ll} 1, & \text{if } h_{j-n} = i+1 \\ -1, & \text{if } h_{j-n} = i \\ 0, & \text{otherwise} \end{array}\right| , \quad i = 1,\ldots,n-1, \; j = 1,\ldots,2n.$$

STRESS function is invariant with respect to translation, i.e. addition of constant values to x_{i1} or/and x_{i2}, $i = 1,\ldots,n$. This disadvantage can be eliminated centering the solution of quadratic programing problem with respect to x_{i1} and x_{i2} by means of equality constraints (25). The latter ensure that the sums of x_{i1} and x_{i2} are both equal to 0: a $(2 \times 2n)$ matrix A_1 multiplied by X is a vector of two sums $(\sum_{i=1}^{n} x_{i1}, \sum_{i=1}^{n} x_{i2})$. Polyhedron $X \in A_{gh}$ is defined by linear inequality constraints (26) and (27). The dimensionality of matrices A_2 and A_3 is $((n-1) \times 2n)$. They contain one element equal to 1 and one element equal to -1 in each row; the other elements are equal to 0. The ith row of A_2 represents $x_{\{j|g_j=i+1\}1} - x_{\{j|g_j=i\}1}$, and the corresponding constraint ensures that $x_{\{j|g_j=i+1\}1} \geq x_{\{j|g_j=i\}1}$. Similarly A_3 ensures the desired sequencing of x_{i2}.

The lower level problem (24)–(27) can be tackled by a standard quadratic programing method. In this case the results below are indicated by '*qp*'. However, solution of a quadratic programing problem is not necessary a local minimizer of the initial problem, i.e. of minimization of STRESS (1). If a solution of a quadratic programing problem is on the boundary, most likely that a local minimizer of the initial problem is in the neighboring subregion. Therefore we have tested two extended versions of local minimization. In the first version, a quadratic programing problem is solved in the neighboring subregion on the opposite side of the active inequality constraints. Minimization by means of quadratic programing is repeated while better values are found and some inequality constraints are active. In the description of the experimental results this type of local minimization is denoted by '*q*'. In the second extended version of local minimization (denoted by '*l*') search is continued by means of Powell's method.

For the second version of the two level algorithm the problem at lower level can be solved by an algorithm supposed for non-differentiable objective functions. Below we investigate such a two level algorithm using the version of local search algorithm by Powell presented in Press et al. (2002).

5 Genetic algorithm for upper level problem

The upper level problem is a combinatorial optimization problem which can be formulated either explicitly or implicitly. In both cases an evolutionary approach is applicable (Michalewicz 1996). The idea is to maintain a population of p suboptimal solutions whose crossover can generate better solutions. An offspring is produced by a combination of crossover and local search operators; some authors call such algorithms memetic (Corne et al. 1999). An initial population is generated performing local search from random starting points. The population evolves generating offsprings of randomly mated parents. The fitness of an individual is defined by the optimal value of the corresponding lower level problem, and an elitist selection is applied.

In the first version of the two level algorithm the chromosome of an individual is represented by a pair of permutations (g, h) of natural numbers $1, \ldots, n$ defining a feasible region of the quadratic programming problem (21). In the second version the chromosome is represented by a local minimizer X implicitly defining a basin of attraction. General structure of an evolutionary algorithm is presented in Fig. 1.

Two versions of general structure of the algorithm in Fig. 1 should be implemented taking into account different encodings of chromosomes. While the initial population for the second version of the algorithm is formed directly from the found local minimizers, in the first case a population of permutations' pairs (g, h) should be formed according to the order of coordinates of the found minimizers $x_{11}, \ldots, x_{n1}$ and $x_{12}, \ldots, x_{n2}$.

The two point crossover operators are similar in both cases where parents are chosen at random. In the first version of the algorithm the chromosomes of parents are denoted $(\hat{g}, \hat{h})$ and $(\check{g}, \check{h})$, where the first corresponds to the better fitted parent. A two point crossover reproducing an offspring (g, h) is defined by the following formula

$$
\begin{aligned}
(g, h) = MIN((&\hat{g}_1, \ldots, \hat{g}_{\xi_1}, \tilde{g}_{\xi_1+1}, \ldots, \tilde{g}_{\xi_2-1}, \hat{g}_{\xi_2}, \ldots, \hat{g}_n), \\
&(\hat{h}_1, \ldots, \hat{h}_{\xi_1}, \tilde{h}_{\xi_1+1}, \ldots, \tilde{h}_{\xi_2-1}, \hat{h}_{\xi_2}, \ldots, \hat{h}_n)),
\end{aligned}
$$

where ξ_1, ξ_2 are two integer random numbers with uniform distribution over $1, \ldots, n$; MIN is a lower level search operator described above; and $\tilde{g}_i$ are numbers from the set $1, \ldots, n$ not included into the set $(\hat{g}_1, \ldots, \hat{g}_{\xi_1}, \hat{g}_{\xi_2}, \ldots, \hat{g}_n)$, and ordered in the same way as they are ordered in $\check{g}_1, \ldots, \check{g}_n$. The sequence of $\tilde{h}_i$ is defined in similar way.

In the second version of the algorithm the crossover operator is defined by the following formula

$$
\begin{aligned}
X = \min((&\hat{x}_{11}, \ldots, \hat{x}_{\xi_1 1}, \check{x}_{\xi_1+1\,1}, \ldots, \check{x}_{\xi_2-1\,1}, \hat{x}_{\xi_2 1}, \ldots, \hat{x}_{n1}), \\
&(\hat{x}_{12}, \ldots, \hat{x}_{\xi_1 2}, \check{x}_{\xi_1+1\,2}, \ldots, \check{x}_{\xi_2-1\,2}, \hat{x}_{\xi_2 2}, \ldots, \hat{x}_{n2})),
\end{aligned}
$$

where X is the chromosome of the offspring; $\hat{X}$ and $\check{X}$ are chromosomes of the selected parents; $\min(Z)$ denotes an operator of calculation of the local minimizer of (1) from the starting point Z using Powell's algorithm; ξ_1, ξ_2 are two integer random numbers with uniform distribution over $1, \ldots, n$; and it is supposed that the parent $\hat{X}$ is better fitted than the parent $\check{X}$ with respect to the value of STRESS.

As it follows from the general structure of the algorithm in Fig. 1 an elitist selection rule is implemented, and search terminates after fixed in advance number of crossovers N_c.

Generate N_{init} uniformly distributed random points. Keep p best.
Perform local search from p random points.
do N_c times
    Randomly with uniform distribution select two parents from a current population.
    Produce an offspring by means of crossover and local minimization.
    If the offspring is better fitted than the worst individual of the current population,
        then the offspring replaces the latter.

Fig. 1 The structure of a genetic algorithm with parameters $(p, N_{\text{init}}, N_{\text{c}})$

6 Experimental investigation

Theoretical comparison of different MDS methods, especially of visualization methods, is difficult. There are many ways to represent features of data by geometric properties of data images. Human perception of geometric images is also ambiguous. Therefore it is difficult to assess efficiency of the main component of a MDS algorithm, i.e. an optimization method used for minimization of STRESS function. We investigate efficiency of the developed optimization algorithm by means of traditional experimental approach in optimization although it does not fully assess quality of visualization.

We start with visualization of well understood geometric objects: vertices of multidimensional cubes and simplices of different dimensionality. Such data is difficult for MDS since the geometric objects extending in all dimensions are aimed to visualize as two-dimensional figures. $n = \mathrm{dim} + 1$ vertices of multidimensional simplex may be defined by

$$v_{ij} = \begin{cases} 1, & \text{if } i = j+1, \\ 0, & \text{otherwise,} \end{cases} \quad i = 1, \ldots, \mathrm{dim}+1, \ \ j = 1, \ldots, \mathrm{dim}.$$

The coordinates of ith vertex of a dim-dimensional hypercube are equal either to 0 or to 1, and they are defined by binary code of $i = 0, \ldots, n-1$, $n = 2^{\mathrm{dim}}$.

For both types of objects symmetric location of vertices is characteristic. In the image of a simplex a special central location of the 'zero' vertex is expected. The other vertices are expected to be shown equally with respect to each other. All vertices of a hypercube are equally far from the center and compose clusters containing 2^d points, where d is any integer number between 1 and dim. Such clusters correspond to edges, faces, etc. In this experiment we compare the images corresponding to the best known STRESS values ignoring computational expenditure.

Dissimilarities between vertices can be measured by Euclidean and city block distances. Figs. 2 and 3 show influence of norm in original and embedding space to the result of MDS, where upper index CB means city block norm and ED means Euclidean norm, e.g. δ^{ED} and d^{CB} means that dissimilarities between vertices are measured by the Euclidean distances in the original space and distances in the embedding space correspond to the city block norm. The vertices are shown as circles. To make representations more visual, adjacent vertices are joined by lines. Lines are darker if they come from 'zero' vertex in the case of simplex and if they come from one of two opposite vertices in the case of hyper-cube.

The images of the hypercube corresponding to city block norm in embedding space well visualize equal location of all vertices of the hypercube with respect to the center. This property is not visible from the images corresponding to Euclidean norm in embedding space. On the other hand, the latter images show the structure composed of 2^d points, as it is the case of the original hypercube.

As expected, the 'zero' vertex of multidimensional simplex is shown at the center of images corresponding to all combinations of norms. The images corresponding to city block norm in embedding space well visualize equal location of other vertices with respect to 'zero' vertex. This property is not highlighted by images corresponding to Euclidean norm in embedding space.

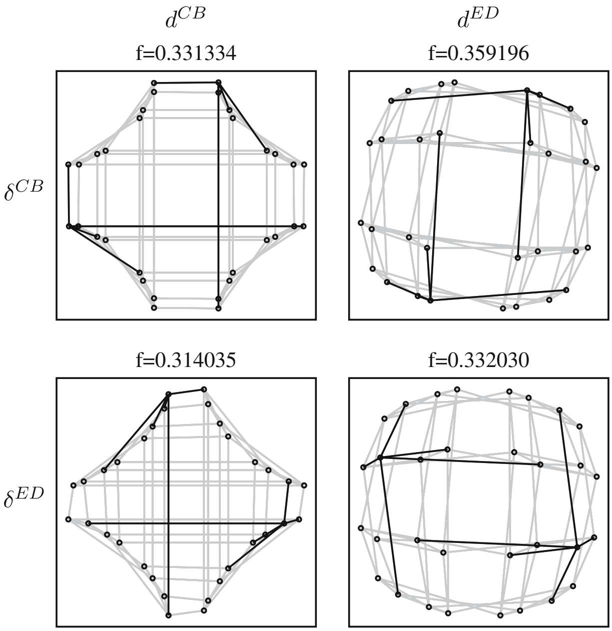

Fig. 2 5-dimensional hyper-cube visualized using different norms in original and embedding spaces

Besides of qualitative assessment of informativeness of the images it is interesting to compare 'visualization errors' quantitatively. To exclude impact of scales a relative error

$$f(X) = \sqrt{\frac{S(X)}{\sum_{i<j} \delta_{ij}^2}}$$

is used for comparisons. The values of $f(X)$ are presented in Figs. 2 and 3 to compare the visualization quality not only heuristically but also with respect to the quantitative precision criterion. In both cases the least error is obtained in the case of Euclidean norm in original space and city block norm in embedding space. This conclusion is consistent with known results on different structure of distances in spaces of different dimensionality (see e.g. Žilinskas 2003).

The examples of visualization of the well known multidimensional geometric objects show that the images corresponding to city block norm in embedding space can be more informative than the images corresponding to Euclidean norm. The development of efficient algorithms for minimization of STRESS with city block distances (1) is an urgent problem since city block distances based MDS methods are underdeveloped with respect to that based on Euclidean distances. The structure of minimization problem (1) suggests two level methods for MDS with city block norm: metaheuristic optimization at upper level and local minimization exploiting piecewise quadratic structure of the objective function at lower level. Some results of quantitative assessment of such two level algorithms are presented below. Test data correspond to vertices of a 4-dimensional hyper-cube and a 12-dimensional simplex. The middle size problems have been chosen for investigation because they hardly can be solved with unsophisticated methods but still can be solved with specially tailored methods

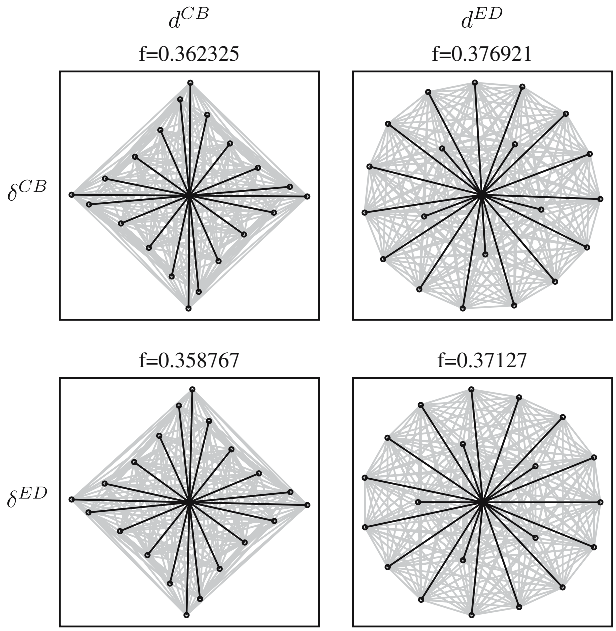

Fig. 3 20-dimensional simplex visualized using different norms in original and embedding spaces

in time acceptable to collect representative statistics for comparison. An algorithm has been run 100 times with each set of parameters to evaluate reliability and speed. Personal computer with AMD Duron 500 MHz processor and RedHat 9 Linux has been used in the experiments.

Several versions of the algorithm were tested. At the upper level a genetic algorithm with $p = 60$, $N_{\text{init}} = 6000$, and $N_{\text{c}} = 1200$ has been used. At lower level different local minimization algorithms described above have been used. The results are summarized in Table 1 and Fig. 4 (related to the visualization of the hypercube) and in Table 2 and Fig. 5 (related to the visualization of the simplex). To assess the performance minimal, average, and maximal running times in seconds ($t_{\min}$, t_{mean}, $t_{\max}$) are estimated from 100 runs. Similarly, minimal, average, and maximal estimates of global minimum in 100 runs ($f^*_{\min}, f^*_{\text{mean}}, f^*_{\max}$) are presented in the tables to show quality of found solutions. The percentage of runs where the best known estimate

Table 1 Minimization results related to MDS of 4-dimensional hyper-cube

qp	q	l	$t_{\min}$	t_{mean}	$t_{\max}$	$f^*_{\min}$	f^*_{mean}	$f^*_{\max}$	perc
Version with quadratic programing									
+			9.29	10.37	11.57	0.2965	0.2965	0.2969	97
+	+		21.81	26.93	31.15	0.2965	0.2965	0.2965	100
+	+	+	61.43	99.54	117.46	0.2965	0.2965	0.2965	100
+		+	57.19	97.06	117.79	0.2965	0.2965	0.2965	100
Version without quadratic programing									
			42.88	57.85	85.44	0.2965	0.2966	0.2970	34

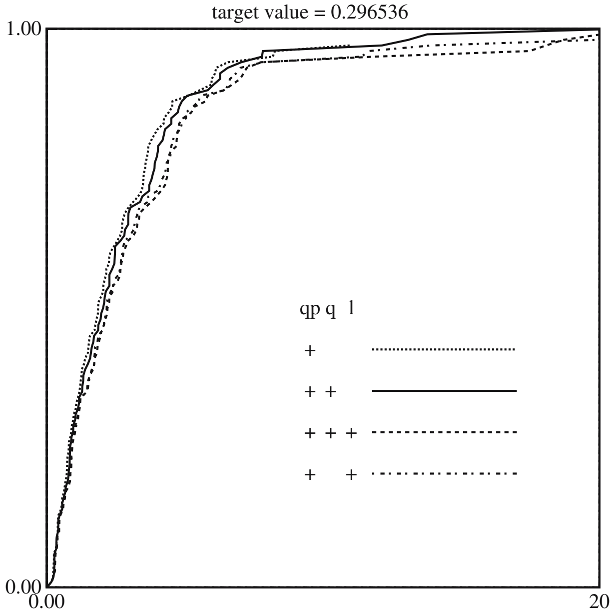

Fig. 4 Time to target in case of 4-dimensional hyper-cube

of global minimum has been found (perc) is presented in the tables as a criterion of reliability of different versions of the algorithm.

The dynamic of minimization is illustrated using plots of time to target (Festa et al. 2002). To evaluate time to target, algorithm is run r times recording the running time when function value at least as good as the target value is found. Let t_i denotes a sequence of time moments, and r_i denotes number of runs where target value is found no later than t_i. The target plot is a plot of r_i/r against t_i. Several target plots presented in the same figure show comparative efficiency of the corresponding algorithms: the graph above the others indicates the most efficient algorithm.

For the problem of visualization of the hypercube the version of the algorithm indexed by '*qp*-q' is most efficient. The other versions of the algorithm taking into account piecewise quadratic structure of STRESS are of similar efficiency. However, the version of the algorithm not taking into account piecewise quadratic structure of

Table 2 Minimization results related to MDS of 12-dimensional simplex

qp	q	l	$t_{\min}$	t_{mean}	$t_{\max}$	$f^*_{\min}$	f^*_{mean}	$f^*_{\max}$	perc
Version with quadratic programing									
+			3.24	3.60	4.04	0.3249	0.3250	0.3259	94
+	+		4.12	5.41	7.23	0.3249	0.3249	0.3259	98
+	+	+	11.70	16.35	23.30	0.3249	0.3249	0.3249	100
+		+	11.30	15.61	22.70	0.3249	0.3249	0.3249	100
Version without quadratic programing									
			16.39	25.53	35.57	0.3249	0.3249	0.3249	100

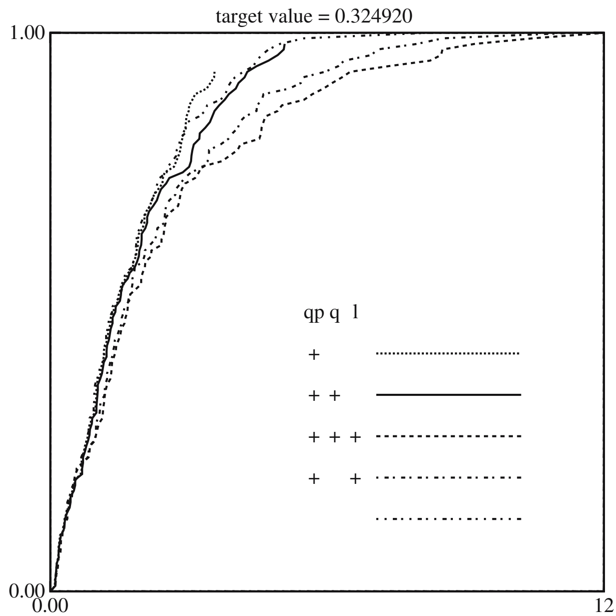

Fig. 5 Time to target in case of 12-dimensional simplex

STRESS is not sufficiently reliable, best known estimate of global minimum has been found only in 34% runs. For the problem of visualization of the simplex the version of the algorithm indexed by 'qp-q' is again most efficient. In this case the performance of versions taking into account piecewise quadratic structure of STRESS does not differ so much from the version not taking into account piecewise quadratic structure of STRESS.

An alternative algorithm for MDS with city block norm in embedding space is based on simulated annealing minimization of (1) (Brusco 2001). For experimental testing of the latter algorithm Morse code confusion data was used. Originally the Morse code confusion data is presented by a proximity matrix (Borg and Groenen 1997). Dissimilarity can be defined via proximity in different ways. We have used a dissimilarity matrix calculated from the proximity matrix according to the formula of (Brusco 2001). The best found value of $s(X)$ reported in (Brusco 2001) is equal to 153.24.

Dimensionality of the minimization problem related to MDS of Morse code confusion data is $n = 64$. Therefore larger values of parameters of our algorithm than in the experiments above have been chosen: $p = 10^2$, $N_{\text{init}} = 10^6$, and $N_c = 10^4$, and the version 'qp-q' for local search has been used. The algorithm has been run ten times. The best value found was 153.001, while the average and maximal estimates of global minimum were equal to 153.380 and 154.435 correspondingly. The average minimization time was $t_{\text{mean}} = 5657$. Two level algorithm has found better value than 153.01 in 60% cases, and with respect to this criterion it outperforms simulated annealing (Brusco 2001) which finds the value 153.24 only in one case out of ten.

Visualization of the results of MDS corresponding to $s(X) = 153.001$ is presented in Fig. 6. It is interesting to note that image of Morse code confusion data resembles

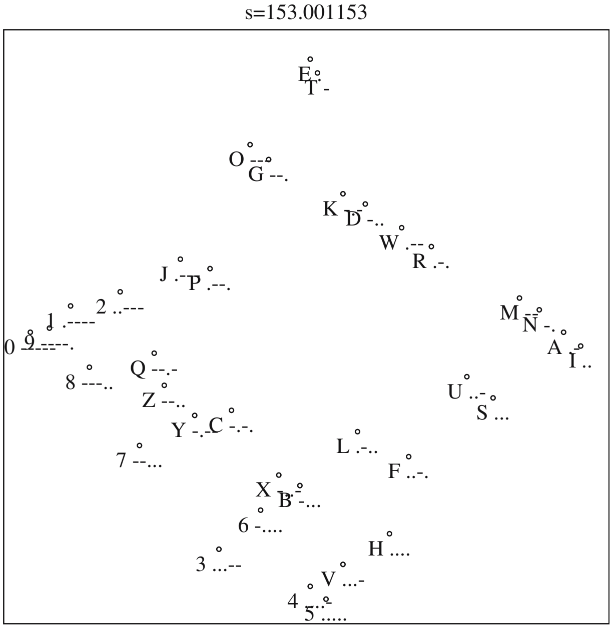

Fig. 6 Image of Morse code data

the image of a hypercube. Such a similarity can be considered as an advantage of MDS with city block norm since Morse codes indeed is a mixture of vertices of different dimensionality.

A disadvantage of the proposed two level method is a large computing time. However, rather good estimate of minimum can be found terminating search after rather a small number of generations $N_c = 10^3$. With such a parameter of genetic algorithm the best value found was 153.082, while the average and maximal estimates of global minimum were equal to 153.635 and 155.074 correspondingly. The average minimization time was $t_{\text{mean}} = 843$. Better values than 153.24 (the record value of Brusco (2001)) have been found in 60% cases.

Experimental results of multidimensional scaling of larger geometric problems are presented in Table 3. In this case larger values of $p = 100$, $N_{\text{init}} = 10^6$, N_c (shown in the table), and the local minimization version ‘qp-q’ have been used. The parameters have been chosen empirically to find the best known $f(X)$ value at least in 30% of runs. Optimization has been repeated 10 times for each problem. The computing time increased essentially. Complexity of the minimization problem seems to be growing faster for simplex than for hypercube.

A two level minimization combining genetic search at upper level and local minimization exploiting piecewise quadratic structure of the objective function at lower level is an efficient algorithm for middle size MDS problems with city block norm in embedding space. Further development of the algorithm targeting larger problems seems prospective. A general idea to enhance performance of an evolutionary search is to start with better genetic material. In the case of MDS problems an initial population can be composed of simple projections from the original space to the embedding space, e.g. by the method of principal components. Local minimization

Table 3 Minimization results related to MDS of larger geometric problems

N_c	t_{min}	t_{mean}	t_{max}	f^*_{min}	f^*_{mean}	f^*_{max}	perc
20-dimensional simplex							
1,000,000	185	731	3,685	0.3623	0.3624	0.3625	30
5-dimensional hyper-cube							
1,000	610	641	673	0.3313	0.3313	0.3314	90

can be improved by a more sophisticated exploitation of piecewise quadratic structure of (1). The computing time can be reduced by means of parallelization, since the developed version of the algorithm can be parallelized rather easily.

7 Conclusions

The MDS methods with city block norm in an embedding space can better visualize some properties of multidimensional objects than Euclidean norm based methods. Two level structure with evolutionary search at upper level and local minimization at lower level is prospective for development of city block norm based MDS algorithms. Solutions found by such methods are sufficiently close to global minima however the solution time of large problems is rather long. To reduce the computing time, the piecewise quadratic structure of STRESS function can be further exploited as well as parallelization of computations.

Acknowledgments The authors are grateful to the unknown referee for the valuable comments and suggestions. The research is supported by Lithuanian State Science and Studies Foundation. The work of the second co-author is supported by the NATO Reintegration grant CBP.EAP.RIG.981300.

References

An, L., Tao, P.: D.C. programing approach to the multidimensional scaling problem. In: Migdalas, A., Pardalos P., Varbrand P. (eds.) From Local to Global Optimization, Kluwer, Dodrecht, pp. 231–276. (2001)

Borg, I., Groenen, P.: Modern Multidimensional Scaling. Springer, New York (1997)

Brusco, M.J.: A simulated annealing heuristics for unidimensional and multidimensional (city block) scaling of symmetric proximity matrices. J. Classif. **18**, 3–33 (2001)

Corne, D., Dorigo, M., Glover, F. (eds.) New Ideas in Optimization. McGraw-Hill, Maidenhead, England (1999)

Cox, T., Cox, M.: Multidimensional Scaling. Chapman and Hall/CRC, Boca Raton (2001)

De Leeuw, J.: Differentiability of Kruskal's stress at a local minimum. Psychometrika **149**, 111–113 (1984)

De Leeuw, J., Heiser, W.: Theory of multidimentional scaling. In: Krishnaiah, P.R. (ed) Handbook of Statistics, vol. 2, pp. 285–316. North Holland, Amsterdam (1982)

Everett, J.: Algorithms for multidimensional scaling. In: Chambers, L. (ed) The practical handbook of genetic algorithms, pp. 2003–2233. Chapman and Hall/CRC, Boca Raton (2001)

Festa, P., Pardalos, P.M., Resende, M.G.C., Ribeiro, C.C.: Randomized heuristics for the max-cut problem. Optim. Methods Softw. **7**, 1033–1058 (2002)

Groenen, P.: The Majorization Approach to multidimentional scaling, p. 110. DSWO, Amsterdam (1993)

Groenen, P., Mathar, R., Heiser, W.: The majorization approach to multidimensional scaling for minkowski distances. J. Classif. **12**, 3–19 (1995)

Groenen, P., Mathar, R., Trejos, J.: Global optimization methods for MDS applied to mobile communications. In: Gaul, W., Opitz, O., Schander, M. (eds.) Data Analysis: Scientific Models and Practical Applications, pp. 459–475. Springer, (2000)

Horst, R., Pardalos, P., Thoai, N.: Introduction to global optimization. Kluwer, Dodrecht (1995)

Klock, H., Buhmann, J.: Data visualization by multidimensional scaling: a deterministic annealing approach. Pattern Recogn. **33**(4), 651–669 (1999)

Leng, P.L., Lau, K.: Estimating the city-block two-dimensional scaling model with simulated annealing. Eur. J. Oper. Res. **158**, 518–524 (2004)

Mathar, R.: Multidimensionale Skalierung. Teubner, Stuttgart (1997)

Mathar, R.: A hybrid global optimization algorithm for multidimensional scaling. In: Klar, R., Opitz, O. (eds.) Classification and knowledge organization, 63–71. Springer, Berlin (1996)

Mathar, R., Žilinskas, A.: On global optimization in two-dimensional scaling. Acta Appl. Math. **33**, 109–118 (1993)

Michalewicz, Z.: Genetic Algorithms + Data Structures = Evolution Programs. Springer, Berlin (1996)

Press, W. et al.: Numerical Recipes in C++. Cambridge University Press, Cambridge (2002)

Törn, A., Žilinskas, A.: Global Optimization, Lecture Notes in Computer Science, vol. 350, pp. 1–250 (1989)

Žilinskas, A.: On the distribution of the distance between two points in a cube. Random Oper. Stoch. Eqs. **11**, 21–24 (2003)