RUTCOR
RESEARCH
REPORT

On the Generalization of the MDS Method

Sadegh Niroomand[a] Szabolcs Takács[b]
Béla Vizvári [c]

RRR 22-2012, September 2012

RUTCOR
Rutgers Center for
Operations Research
Rutgers University
640 Bartholomew Road
Piscataway, New Jersey
08854-8003
Telephone: 732-445-3804
Telefax: 732-445-5472
Email: rrr@rutcor.rutgers.edu
http://rutcor.rutgers.edu/∼rrr

[a] Dept. of Industrial Engineering, Eastern Mediterranean University, sadegh.niroomand@cc.emu.edu.tr, Famagusta, Mersin 10, Turkey, tel. +90 392 6301161, fax: +90 392 6302988

[b] Institute of Psychology, Károli Gáspár University, Budapest, tretark@freemail.hu

[c] Department of Industrial Engineering, Eastern Mediterranean University, bela.vizvari@emu.edu.tr

Rutcor Research Report
RRR 22-2012, September 2012

On the Generalization of the MDS Method

Sadegh Niroomand Szabolcs Takács Béla Vizvári

Abstract. The Multi-Dimensional Scaling (MDS) method is used in statistics to detect hidden interrelations among multi-dimensional data and it has a wide range of applications. The method's input is a matrix that describes the similarity/dissimilarity among objects of unknown dimension. The objects are generally reconstructed as points of a lower dimensional space to reveal the geometric configuration of the objects. The original MDS method uses Euclidean distance, for measuring both the distance of the reconstructed points and the bias of the reconstructed distances from the original similarity values. In this paper, these distances are distinguished, and distances other than Euclidean are also used, generalizing the MDS method. Two different distances may be used for the two different purposes. Therefore the instances of the generalized MDS model are denoted as (l_p, l_q) model, where the first distance is the type of distance of the reconstructed points and the second one measures the bias of the reconstructed distances and the similarity values. In the case of l_1 and l_∞ distances mixed-integer programming models are provided. The computational experiences show that the generalized model can catch the key properties of the original configuration, if any exist.

Keywords: Multidimensional Scaling, Mixed Integer Linear Programming, Optimization

Acknowledgements: Please insert the acknowledgement here.

1 Introduction

A crucial function of statistics is to analyze data to interpret relationships of a number of variables. One method for accomplishing this is to map the data into a lower-dimensional space, allowing the interrelation among the investigated objects to become visible. Multi-Dimensional Scaling (MDS) and factor analysis are well-known methods for performing such analyses.

Multidimensional scaling explores the similarities or dissimilarities in the measures of objects (Machado et al. 2010). The data used in MDS can be referred to by several names (dissimilarities, similarities, distances, or proximities). However, the terms "dissimilarity" and "similarity", are the most common. In general, it is assumed that the similarity values are nonnegative and symmetric. The objects are completely identical if the similarity value is zero. A higher (dis)similarity value means higher dissimilarity between the objects.

The same method can be used to reconstruct the geometric configuration of many finite points. Reconstruction consists of finding proper positions in the plane or in space for the points (Niroomand et al. 2011), but the results of the reconstruction must not be geometrically identical to the original structure. Depending on the distance matrix for a set of points, if there is any clear underlying geometric structure, the results of the reconstruction will reflect the structure, as well. Thus, only the relative positions of points in the space are determined.

The traditional MDS method assumes Euclidean distance, i.e., the Euclidean distances between the reconstructed points are compared to the similarity values of the objects. It minimizes the total quadratic error that is, it measures the "distance" between the similarity matrix and the matrix formed by the distances of the reconstructed points on a Euclidean way. If the objects are in Euclidean space and MDS maps the objects into that same dimensional Euclidean space, then MDS provides a perfect reconstruction in the sense that the obtained geometric configuration is congruent, i.e., with appropriate rotation and shifting the new configuration can cover the original one.

MDS has applications in facility layout problems (Niroomand et al. 2011), as well as in the medical (Smallman-Raynor et al. 2001), psychological (Jaworska et al. 2009) and economic (Syrquin 1978) domains, in addition to others. In the case of facility layout problems (Niroomand et al. 2011), the authors reconstructed the Kra30a problem (Hahn et al. 2001) from the Quadratic Assignment Problem Library (QAPLIB), using the MDS method. This problem is in 3-dimensional space with weighted l_1 distances but the weights are unknown in the source of the problem. The reconstruction was performed in 2 and 3-dimensions. In both configurations of points, there were some symmetry and regularity properties.

The main purpose of this paper was to generalize the MDS method. Distances different from l_2 are used for both the distances of objects and for measuring the total error. A problem is of type (l_p, l_q) $(1 \leq p, q \leq +\infty)$ if the distance of the reconstructed objects is of l_p type and the total error is measured in l_q. Thus, the traditional MDS method is of type (l_2, l_2) type. In the case of l_1 and l_∞ distances, integer programming models described the problem correctly.

2 Types of distances used in reconstruction models

The reconstruction models can be used for different types of distances such as l_1, l_p and l_∞ distances. While l_1 is the Manhattan distance between points, l_2 is used to show the Euclidian distance between two points, which is a special case of l_p distance, and l_∞ is the maximum absolute difference between the coordinates of a pair of points.

To find the proper positions of the points, the types of distances between the reconstructed points should be determined in the reconstruction model. Usually this distance is of l_1, l_2 or l_∞ type. Additionally, the bias or tolerance of these distances from those of the similarity matrix should be calculated and minimized. This tolerance (bias) can be the same for all pairs of points. In this case, the bias is of l_∞ type. In other cases, the bias of each pair of points is different if type $l_p\,(1 \leq p < +\infty)$ distance is used in the reconstruction model.

3 General reconstruction model

The two main parts of the reconstruction model are the constraints, which are discussed first, and the objective function, which is the measure of error that must be minimized. The models are elaborated 2-dimensionally. The generalizations, however, are straightforward. The constraints and objective function can be introduced for $l_1, l_p(1 < p < +\infty)$ and l_∞ types of distances, separately. Therefore, there will be 3 types of constraints sets and 3 types of objective functions, which are introduced below.

3.1 l_1 type constraints

A mixed-integer linear programming model is discussed here for the case of 2 dimensions which includes the points on the plane.

The l_1 distance between points (x_j, y_j) and (x_k, y_k) is defined as

$$l_1((x_j, y_j), (x_k, y_k)) = |x_j - x_k| + |y_j - y_k| =$$
$$\max\{x_j - x_k + y_j - y_k, x_j - x_k + y_k - y_j, x_k - x_j + y_j - y_k, x_k - x_j + y_k - y_j\}$$

If the reconstruction of n points on a plane is needed, let $t_jk\,(1 \leq j, k \leq n)$ be the l_1 distance between reconstructed j^{th} and k^{th} points in a square $((0,0),(0,h),(h,h),(h,0))$ where $h > 0$. This value must be at least the highest distance among the known distances of the similarity matrix.

The first set of constraints for each pair of cells will be used to force the four above-mentioned sums to be less than or equal to the reconstructed l_1 distance between the pair of points:

$$x_j - x_k + y_j - y_k \leq t_{jk} \tag{1}$$

$$x_j - x_k + y_k - y_j \leq t_{jk} \tag{2}$$

$$x_k - x_j + y_j - y_k \leq t_{jk} \tag{3}$$

$$x_k - x_j + y_k - y_j \leq t_{jk} \tag{4}$$

$$1 \leq j < k \leq n$$

In the second set of constraints, the opposite inequalities are claimed. At least one of the above-mentioned quantities on the left-hand sides must be greater than or equal to the reconstructed l_1 distance between two points. Let M be a large number, $M = 4h$ is then a proper choice. The constraints are

$$x_j - x_k + y_j - y_k + Mu_{jk1} \geq t_{jk} \tag{5}$$

$$x_j - x_k + y_k - y_j + Mu_{jk2} \geq t_{jk} \tag{6}$$

$$x_k - x_j + y_j - y_k + Mu_{jk3} \geq t_{jk} \tag{7}$$

$$x_k - x_j + y_k - y_j + Mu_{jk4} \geq t_{jk} \tag{8}$$

$$1 \leq j < k \leq n$$

with

$$u_{jk1}, u_{jk2}, u_{jk3}, u_{jk4} \in \{0, 1\} \quad 1 \leq j < k \leq n \tag{9}$$

Mu_{jki} is used as a correction value of i^{th} inequality for the above set of constraints. If $u_{jki} = 1$, then the i^{th} inequality automatically is satisfied. To obtain the l_1 distance, at least one of the constraints must be satisfied without using the correction term. Thus, the cut

$$u_{jk1} + u_{jk2} + u_{jk3} + u_{jk4} \leq 3 \tag{10}$$

must be applied.

The obvious set of constraints is to force the points to be in the square of h:

$$0 \leq x_j, y_j \leq h, \quad j = 1, ..., n \tag{11}$$

3.2 l_∞ type constraints

The l_∞ distance between points (x_j, y_j) and (x_k, y_k) is defined as

$l_\infty((x_j, y_j), (x_k, y_k)) = max\{|x_j - x_k| + |y_j - y_k|\} = max\{x_j - x_k, x_j - x_k, y_j - y_k, y_k - y_j\}$

Assume then that the problem is to reconstruct n points in the above-mentioned square by using l_∞ distance between reconstructed points. The constraint logic is similar to the l_1 case. For each pair of points, the first set of constraints claims that all four of the above terms are less than or equal to the reconstructed l_∞ distance:

$$x_j - x_k \leq t_{jk} \tag{12}$$

$$x_j - x_k \leq t_{jk} \tag{13}$$

$$y_j - y_k \leq t_{jk} \tag{14}$$

$$y_k - y_j \leq t_{jk} \tag{15}$$

$$1 \leq j < k \leq n$$

In the second set of constraints, with the help of binary variables, at least one of the above-mentioned quantities is greater than or equal to the reconstructed l_∞ distance. Using a large number with estimation of $M = 2h$, the constraints are:

$$x_j - x_k + Mu_{jk1} \geq t_{jk} \tag{16}$$

$$x_j - x_k + Mu_{jk2} \geq t_{jk} \tag{17}$$

$$y_j - y_k + Mu_{jk3} \geq t_{jk} \tag{18}$$

$$y_k - y_j + Mu_{jk4} \geq t_{jk} \tag{19}$$

$$1 \leq j < k \leq n$$

where

$$u_{jk1}, u_{jk2}, u_{jk3}, u_{jk4} \in \{0, 1\} \quad 1 \leq j < k \leq n \tag{20}$$

If $u_{jki} = 1$, using Mu_{jki}, the i^th inequality automatically is satisfied. The j^th and k^th points are positioned properly, if at least one of the above-mentioned constraints is satisfied without using the correction term. Thus the cut

$$u_{jk1} + u_{jk2} + u_{jk3} + u_{jk4} \leq 3 \tag{21}$$

must be applied.

Additionally, the points are limited to fall in the square of h:

$$0 \leq x_j, y_j \leq h, \quad j = 1, ..., n \tag{22}$$

3.3 l_p type constraints

The l_p distance between points (x_j, y_j) and (x_k, y_k) is defined as

$$l_p((x_j, y_j), (x_k, y_k)) = (|x_j - x_k|^p + |y_j - y_k|^p)^{\frac{1}{p}}$$

The nonnegative $l_p\,(1 < p < +\infty)$ distance can be expressed by a single equation:

$$|x_j - x_k|^p + |y_j - y_k|^p = t_{jk}^p \tag{23}$$

$$t_j k \geq 0 \tag{24}$$

The points also should be positioned in the square of h:

$$0 \leq x_j, y_j \leq h, \quad j = 1, ..., n \tag{25}$$

Of course, the well-known case of l_p distance is the Euclidean distance if $p = 2$.

3.4 l_1 type of objective function

Before identification of the objective function, the bias between the reconstructed distances and the elements of the similarity matrix for each pair of points should be calculated, e.g., $\tau_j k$ for points j and k. Therefore, in the case of l_1 type of objective function, this bias is separately defined for each pair of points and calculated by the following set of constraints:

$$d_{jk} - t_{jk} \leq \tau_{jk} \tag{26}$$

$$t_{jk} - d_{jk} \leq \tau_{jk} \tag{27}$$

Therefore the objective function will minimize the sum of all tolerances as follows:

$$\min \sum_{j=1}^{n-1} - \sum_{k=j+1}^{n} \tau_{jk} \tag{28}$$

3.5 l_∞ type of objective function

In this type of objective function, the same tolerance of τ is considered for the reconstructed distance and the related element of the similarity matrix for each pair of points. Thus, using the following set of constraints, the tolerance is calculated and subsequently minimized:

$$d_{jk} - t_{jk} \leq \tau \tag{29}$$

$$t_{jk} - d_{jk} \leq \tau \tag{30}$$

$$\min \tau \tag{31}$$

3.6 l_p type of objective function

In l_p type of objective function, the different biases between the reconstructed distance and the distance from the similarity matrix for each pair of points are first calculated. Next the sum of p^{th} power for all tolerances is minimized by use of the following set of constraints and the objective function:

$$d_{jk} - t_{jk} \leq \tau_{jk} \tag{32}$$

$$t_{jk} - d_{jk} \leq \tau_{jk} \tag{33}$$

$$\min \sum_{j=1}^{n-1} \sum_{k=j+1}^{n} \tau_{jk}^{p} \tag{34}$$

3.7 Problem types

Each type of objective function can be used with all types of constraints. This means that 9 possible reconstruction models may be considered.

The general notation of (a, b) is used to reference the utilized model. The first element of the notation signifies the type of constraints and the second element shows the type of objective function that is used in the reconstruction model. a and b can be selected from all above-mentioned distances, e.g., l_1, l_p and l_∞ distances. For example the reconstruction model of (l_1, l_∞) distances, refers to the mathematical model, which includes l_1 type constraints and l_∞ type objective functions.

4 Computational experiments

To perform computational experiments, some test problems were generated. These test problems originally contained special geometric configurations, and each test problem contained the coordinates of a limited number of points along a grid or circuit. Likewise a similarity matrix of each test problem was also obtained from the distance between the points of that test problem. The distance type of similarity matrix is different in each reconstructed model. More details of test problems are presented in Table 1.

XPRESS-IVE optimization software was used to perform the calculations. This software has a high capacity for optimizing the linear, quadratic and nonlinear optimization models.

Table 1. Details of test problems used in computational experiments.

Test No.	Number of Points	Original Configuration
1	4	Square of 2×2 Points
2	9	Square of 3×3 Points
3	10	Boundary of Circuit
4	11	Boundary of Circuit
5	12	Boundary of Circuit
6	13	Boundary of Circuit
7	14	Boundary of Circuit
8	15	Boundary of Circuit
9	16	Square of 4×4 Points
10	16	Boundary of Circuit
11	17	Boundary of Circuit
12	18	Boundary of Circuit
13	19	Boundary of Circuit
14	20	Boundary of Circuit
15	25	Square of 5×5 Points

The reconstruction models, even in the case of exact reconstruction, may give relative positions of points only. For congruent solutions, it may be necessary to rotate and/or shift the points to obtain a perfect cover.

4.1 Computational experiments of the (l_1, l_∞) reconstruction model

To construct the (l_1, l_∞) reconstruction model, the above mentioned l_1 type constraints and l_∞ type objective function were chosen. The distance type of l_2 was used in the similarity matrix. The run times were long, generally taking more than 2 hours except for the first and second test problems. The objective function value of the (l_1, l_∞) reconstruction model is shown in Table 2, which illustrates the unique tolerance of the reconstructed distances from the distances of the similarity matrix and also the optimality status for each problem.

The optimality of only the first four problems was proved. In contrast, the optimality was not proved by XPRESS software in other problems, although a positive lower bound was obtained. The structure of the reconstructed points on the plane has similar configuration but requires further geometric transformation to obtain exactly the original one. Figures 1 and 2 show the reconstructed configuration for test problems of 9 points on a square and 11 points on a circuit respectively.

Figure 1: The original and reconstructed structure for the second test problem. The (l_1, l_∞) model is applied.

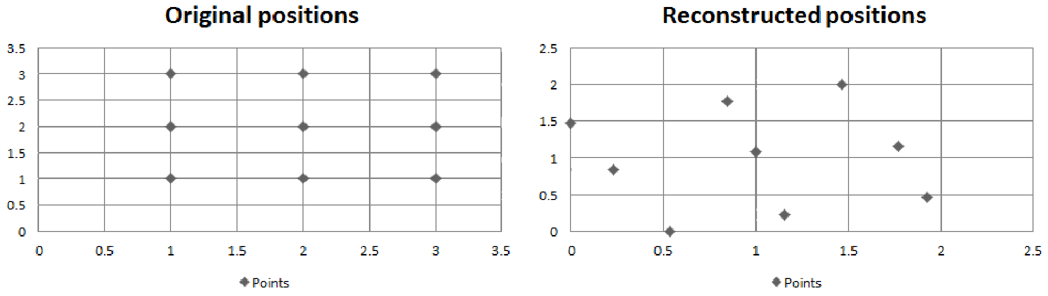


Table 2. Objective function values of the (l_1, l_∞) reconstruction model

Test No.	**Number of Points**	**Objective Function Value**	**Optimality Status**
1	4	0	optimality was proved
2	9	0.1527	optimality was proved
3	10	0.1553	optimality was proved
4	11	0.1376	optimality was proved
5	12	0.2824	optimality was not proved
6	13	0.3755	optimality was not proved
7	14	0.5534	optimality was not proved
8	15	0.2489	optimality was not proved
9	16	0.8980	optimality was not proved
10	16	0.3477	optimality was not proved
11	17	0.5697	optimality was not proved
12	18	0.5570	optimality was not proved
13	19	0.9543	optimality was not proved
14	20	0.5296	optimality was not proved
15	25	2.7190	optimality was not proved

4.2 Computational experiments of the (l_1, l_1) reconstruction model

The (l_1, l_1) reconstruction model consists of l_1 type constraints and objective function. Additionally, the l_2 distance between original points is used in the similarity matrix. Table 3 shows the results of this reconstruction model that were obtained using XPRESS software with long CPU times. In this experiment, only the optimality of the first problem was proved.

In this case the reconstructed configuration again requires some geometric transportation to obtain exactly the original configuration; however, the similarity is clearly recognizable.

Figure 2: The original and reconstructed structure for the 11-points problem on the circuit. The (l_1, l_∞) model is applied.

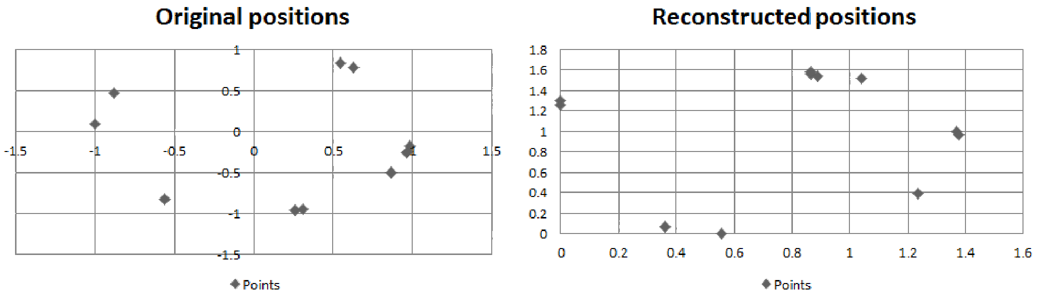
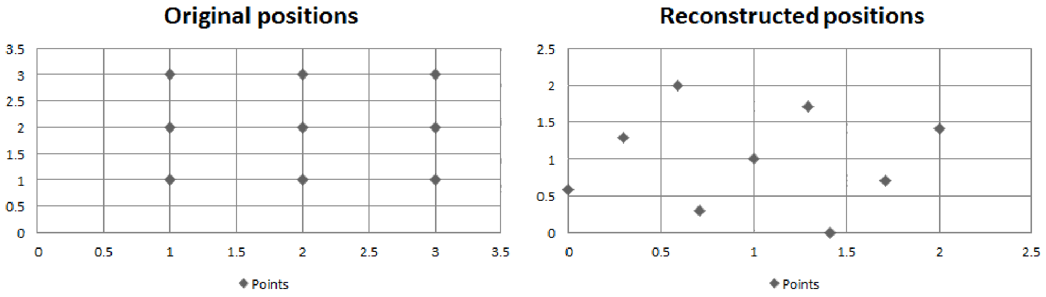


Figure 3: The original and reconstructed structure for the second test problem. The (l_1, l_1) model is applied.



Figures 3 and 4 illustrate the problems of having 9 points on a square and 11 points on a circuit, respectively.

Figure 4: The original and reconstructed structure for the 11 points problem on a circuit. The (l_1, l_1) model is applied.

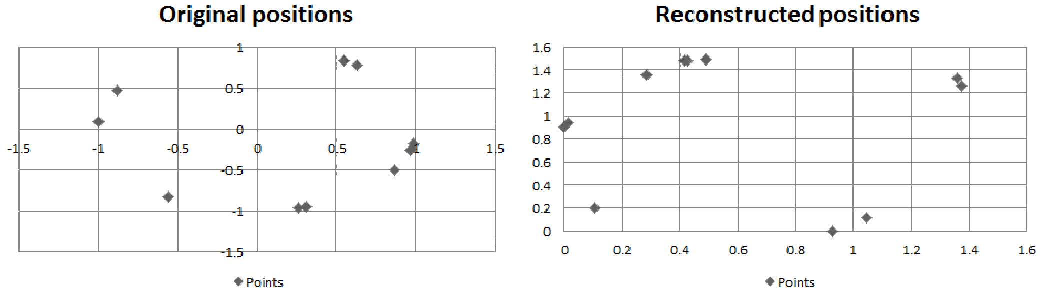


Table 3. Objective function values of the (l_1, l_1) reconstruction model

Test No.	Number of Points	Objective Function Value	Optimality Status
1	4	0	optimality was proved
2	9	2.3430	optimality was not proved
3	10	2.3660	optimality was not proved
4	11	2.7473	optimality was not proved
5	12	5.9751	optimality was not proved
6	13	5.8956	optimality was not proved
7	14	11.6358	optimality was not proved
8	15	6.9336	optimality was not proved
9	16	16.5577	optimality was not proved
10	16	10.1937	optimality was not proved
11	17	11.9211	optimality was not proved
12	18	25.6415	optimality was not proved
13	19	12.7347	optimality was not proved
14	20	26.3420	optimality was not proved
15	25	59.8229	optimality was not proved

4.3 Computational experiments of the (l_2, l_2) reconstruction model

The reconstruction model of (l_2, l_2) is used in this part of the paper. This model demonstrates that the constraints and objective function are of l_2 type. This model is applied with three different similarity matrices. Necessarily, then, the similarity matrix of the same original points of the test problems are used when

1. The similarity matrix contains the l_1 type distance of the original points,

2. The similarity matrix contains the l_2 type distance of the original points,

3. The similarity matrix contains the l_∞ type distance of the original points.

Therefore, the (l_2, l_2) reconstruction model is used to reconstruct the points that are related to the above-mentioned similarity matrices, separately. MATLAB Optimizer is used to solve the (l_2, l_2) reconstruction models. Table 4 shows the objective function values using different similarity matrices.

Similar to previously reconstructed structures, those for the two previous test problems (9 points among a square and 11 points on a circuit), as well as the test for 20 points on a circuit using each similarity matrix are shown in Figures 5 to 7.

Table 4. Objective function values (OBF) of the (l_2, l_2) reconstruction model

Test No.	**No. of Points**	**OBF with l_1 similarity matrix**	**OBF with l_2 similarity matrix**	**OBF with l_∞ similarity matrix**
1	4	0.0032	0.0037	0.0029
2	9	0.0028	0.0041	0.0028
3	10	0.0526	0.0042	0.0547
4	11	0.0616	0.0040	0.1239
5	12	0.0528	0.0043	0.0285
6	13	0.0584	0.0041	0.0316
7	14	0.0438	0.0038	0.0240
8	15	0.0402	0.0031	0.0422
9	16	0.2204	0.0033	0.0652
10	16	0.0735	0.0037	0.0712
11	17	0.0371	0.0043	0.0242
12	18	0.0412	0.0027	0.0470
13	19	0.0388	0.0040	0.0485
14	20	0.0498	0.0027	0.0278
15	25	0.0676	0.0031	0.3569

5 Discussion and conclusion

The similarity values in the mathematical model of MDS are considered more or less to be distance type quantities. Otherwise it would not be possible to reconstruct a geometric configuration. Also supporting this assumption are the conditions starting that for all j and k the equations $d_{jj} = 0$, and $d_{jk} = d_{kj}$ must hold. Obviously the distance which is and must be used most frequently, is the Euclidean distance. However, other types of distances exist both in real life problems and mathematical theory. Moreover, the original MDS method uses

Figure 5: The original and reconstructed structures for the second test problem using (l_2, l_2) reconstruction models.

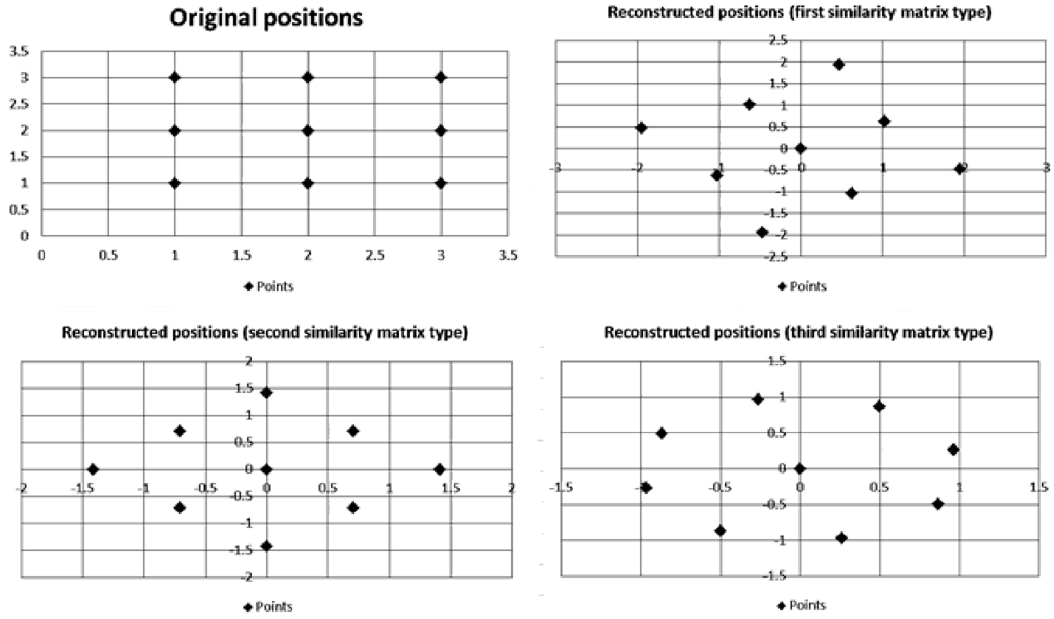


Figure 6: The original and reconstructed structures for the 11 points test problem using (l_2, l_2) reconstruction models.

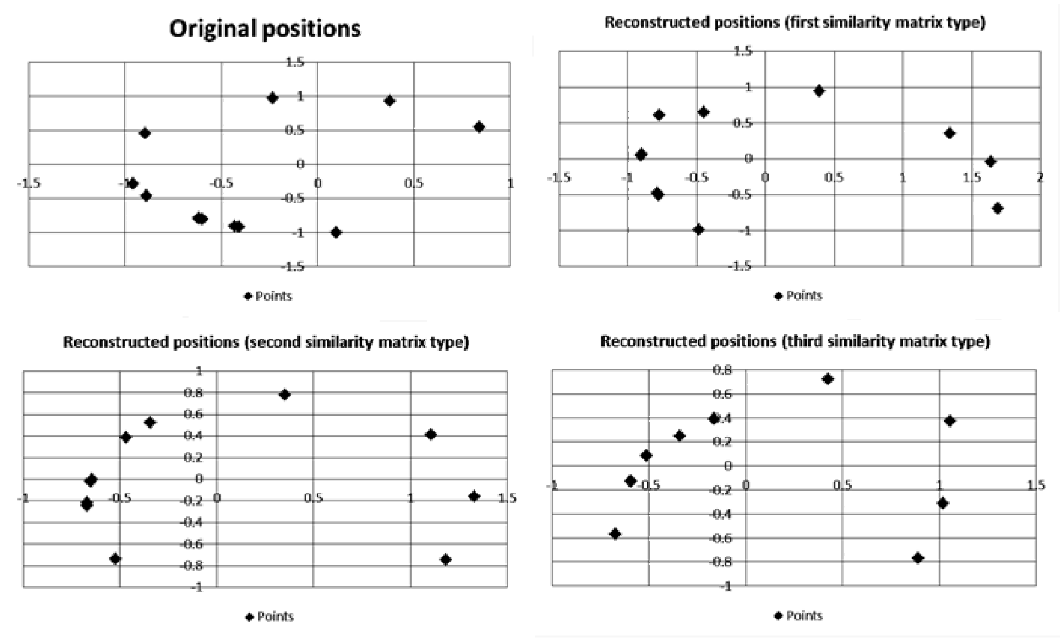
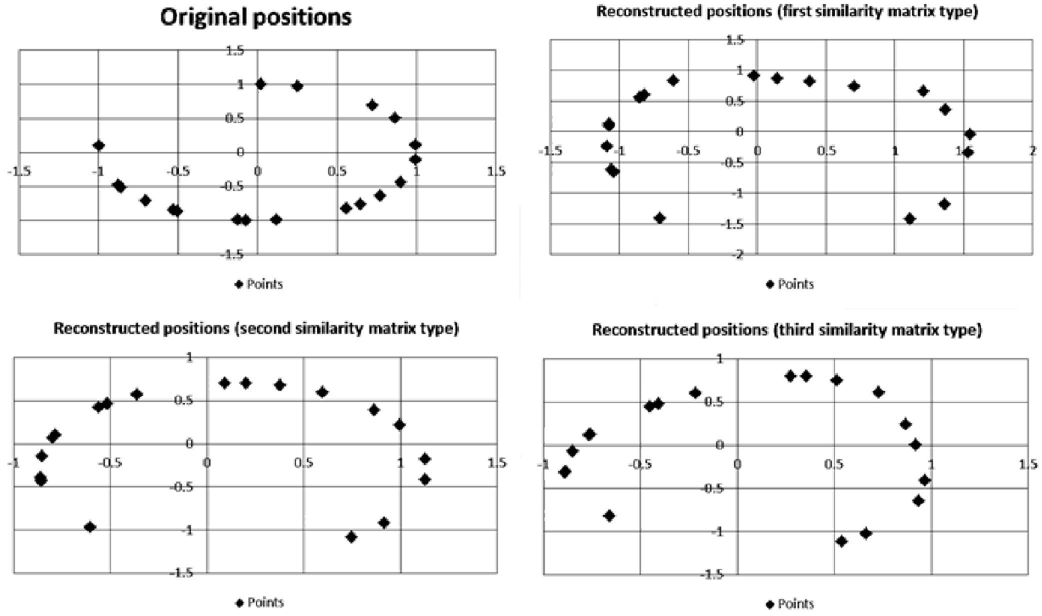


Figure 7: The original and reconstructed structures of the 20 points test problem using (l_2, l_2) reconstruction models.



two distances. One measures the geometric distances between the reconstructed points while the other measures the biases of the geometric distances from the similarity values. This observation then leads to a generalization of the MDS method where a particular version is characterized by the types of the two distances. In the cases of l_1 and l_∞ distances mixed-integer programming models described the problems. In the computational experiences, the distances used in the reconstruction were different from those used to create the similarity matrix. However, while the reconstructed configurations did contain some distortions, the original structure was still recognizable.

References

[1] Hahn, P. M., Krarup, J. (2001). A hospital facility layout problem ?nally solved, Journal of Intelligent Manufacturing, 12, 487496.

[2] J. Tenreiro Machado, Goncalo Monteiro Duarte, Fernando B. Duarte (2010), Identifying Economic Periods and Crisis with the Multidimensional Scaling, Nonlinear Dynamics, 4, 611-622.

[3] Matthew Smallman-Raynor, Andrew D Cliff (2001), Epidemiological spaces: the use of multidimensional scaling to identify cholera diffusion processes in wake of the Philippines insurrection, 18991902, Trans. Inst. Br. Geogr., 26, 288-305.

[4] Moshe Syrquin (1978), The Application of Multidimensional Scaling to the Study of Economic Development, The Quarterly Journal of Economics, 92(4), 621-639.

[5] Natalia Jaworska, Angelina Chupetlovska-Anastasova(2009), A Review of Multidimensional Scaling (MDS) and its Utility in Various Psychological Domains, Tutorials in Quantitative Methods for Psychology, 5(1), 1-10.

[6] Quadratic Assignment Problem Library (QAPLIB) homepages, http://www.opt.math.tu-graz.ac.at/qaplib/ and http://www.seas.upenn.edu/qaplib/ (2011).

[7] Sadegh Niroomand, Szabolcs Takács, Béla Vizvári (2011), To lay out or not to lay out?, Annals of Operations Research, 191, 183-192.