Les Cahiers du GERAD ISSN: 0711–2440

A Variable Neighborhood Search Algorithm for Multidimensional Scaling in Arbitrary Norm

G. Caporossi
S. Taboubi

G–2005–60

August 2005
Revised: May 2006

A Variable Neighborhood Search Algorithm for Multidimensional Scaling in Arbitrary Norm

Gilles Caporossi

Sihem Taboubi

GERAD and HEC Montréal
3000, chemin de la Côte-Sainte-Catherine
Montréal (Québec) Canada H3T 2A7
{gilles.caporossi, sihem.taboubi}@hec.ca

August 2005
Revised: May 2006

Les Cahiers du GERAD

G–2005–60

Abstract

The multidimensional scaling (MDS) aims at finding coordinates for a set of n objects in a (low) q dimensional space that best fits dissimilarity information. To build a perception map to represent the relative positions of objects under study, the mapping dimension used is generally 2. The application area for MDS is mostly psychology and marketing. Unfortunately, the procedure is very sensitive to the parameters used and to the quality of the optimization performed. In this paper, we propose a reliable and flexible algorithm that allows the researcher to study the impact of the Minkowsky parameter (used to define distance measure between objects) on the results. The core of the algorithm is the use of the Variable Neighborhood Search (VNS) metaheuristic. The algorithm developed is tested with benchmark data (Morse code data). The choice of the best Minkowsly parameter is then discussed through the use of generated data and real data collected in an experiment where respondents had to rate pairs of national and private brands on the basis of their dissimilarities. The impact of distance measure on the quality of the perceptual maps is evaluated through the computation of Stress. Two perceptual maps are generated with different distance measures in order to examine their differences.

Key Words: Multidimensional scaling, Algorithm Metaheuristic, Norm selection, Marketing application.

Résumé

Le "multidimensional scaling" a pour but de trouver des coordonnées pour des objets en se basant sur les informations relatives à leurs distances ou dissimilarités. En considérant les coordonnées des n objets dans l'espace à p dimensions comme des variables, le "multidimensional scaling" peut être considéré comme un problème d'optimisation globale non convexe avec variables continues. Dans cet article, nous proposons un algorithme basé sur la Recherche à Voisinages Variables (RVV) pour résoudre ce problème. L'algorithme est testé sur un problème étalon classique à l'aide de diverses mesures de distance ou d'erreurs; les résultats sont ensuite décrits et discutés.

1 Introduction

The multidimentional scaling (MDS) aims at finding coordinates for a set of n objects in a (low) q dimensional space that best fits dissimilarity (or similarity) information. The application area for such technique is mostly psychology and marketing. Indeed, according to Carroll and Green (1997), the first application of MDS in marketing research is attributed to Torgerson (1958), a psychometrician that examined consumers' perceptions of new silver products in New England in the 50s. The graphical representation of this kind of information in a map was then used by Stefflre (1969) to illustrate, in a simple manner, the positioning of products and brands according to their perceived similarities. MDS was then applied by many marketing researchers to the area of innovation and brand (and store) positioning.[1] The success of MDS was followed by a skeptical movement toward other multivariate techniques[2] (i.e. conjoint analysis, multiple regressions). A parallel stream of research criticized the methodological issues related to MDS at the stages of data collection (Bijmolt and Wedel, 1995), and dimensionality (Kruskal, 1964).

The methodological stream of research that dealt with the issue of distance estimation in MDS was mentioned by Kruskal (1964). It highlighted the fact that the optimization problem underlying MDS is non-convex and has a set of local optima (even for relatively small problems) that lead to biased perceptual maps, when the euclidian distance is used as an estimation method of distances between the objects in the map.

Many authors suggested the use of different algorithms in order to solve this optimization problem. Hubert *et al.* (1992) recommended the use of combinatorial methods, while Groenen and Heiser (1996) and Groenen *et al.* (1999) developed heuristics *i.e.,* tunneling and distance smoothing, and simulated annealing (Brusco, 2001 ; Leung and Lau, 2004). Unfortunately, none of these methods takes into account the recent developments in the area of global and combinatorial optimization.

In this paper, we propose a reliable and flexible algorithm that allows the researcher to study the impact of the distance measure (known as the Minkowsky parameter) on the results.[3]

The core of the algorithm is the use of the Variable Neighborhood Search (VNS) metaheuristic (Hansen and Mladenović, 2001, Mladenović and Hansen 1997, Caporossi and Hansen 2005).

This algorithm is then applied to a marketing problem related to the positioning of store brands compared to national brands. Many perceptual maps are produced by using different distance measures computed from data obtained in an experiment. The experiment is

[1] For a comprehensive review of marketing applications of MDS, see Cooper (1983)
[2] Carroll and Green, (1997)
[3] The program is available from the authors upon request

done with a sample of 45 respondents (mainly students) that had the task of indicating the levels of dissimilarities between 8 brands providing acetaminophen 325 mg caplets (5 store brands and 3 national brands). Among these 45 respondents, only those familiar with all the brands (16 respondents) were selected to estimate distances and generate perceptual maps.

The first section highlights the issue of choosing the right parameter when computing distances in perceptual maps. The second section describes the problem and its formulation. The third explains the VNS metaheuristic and its implementation. The fourth section is devoted to the evaluation of the performance of the algorithm and numerical results. The fifth section describes the numerical impacts of the variation of the Minkowsky parameter p on the error (Stress). A real data analysis is proposed on the sixth section and the last one concludes.

2 The distance measure issue

MDS provides a graphical representation of objects on the basis of distances computed from the positions of the objects on the map. Distances should reflect the similarities and dissimilarities between them.

To evaluate the distance between pairs of objects, a version of the Minkowsky formula is usually used. Let $X = \{x_{ik}\}$ be the coordinates matrix where the x_{ik} is the k^{th} coordinate of the object i. If the dimension of the space is q, the Minkowsky distance d_{ij} between objects i and j is defined as follows:

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

where p is called the Minkowsky parameter. Particular cases of the Minkowsky distance arise when $p = 1$, $p = 2$ or $p = \infty$; the corresponding distances are city-bloc (or Manhattan) distance, Euclidian distance or maximum difference between coordinates. If the Euclidian distance is mainly used in practice, there is no evidence that it best corresponds to the perception people have. For instance, consider a study on objects having only two attributes, say the shape and the color. If we would like to analyze the relative perception of a blue square, a red square and a red disk, we would most likely find that the distance between the blue square and the red square is the same as the distance between the red square and the red disk: 1 as they only differ by one attribute. The distance between the blue square and the red disk will probably be 2 rather than $\sqrt{2}$. This simplistic example gives the idea that we cannot be sure our brain uses euclidian distance (in fact, we don't know which distance it uses). On the other hand, there is no evidence either that the space in which it is evaluated is 2, or the number of attributes the objects under study has.

3 Problem formulation

If the dissimilarity between objects i and j is noted δ_{ij}, a version of the MDS problem could be to minimize the following generalization of Stress S:

$$\min S(X) = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} w_{ij} |f(\delta_{ij}) - d_{ij}|^l \tag{2}$$

where w_{ij}, the weight associated to the pair of objects i and j reflects the confidence one have in the corresponding δ_{ij}, $f()$ is a non decreasing transformation function, and l the objective criterion value; if $l = 1$, the objective is to minimize the sum of absolute deviations and if $l = 2$, the objective is to minimize the sum of squared deviations (sum of squared errors). In the present paper, we will concentrate on the case where $w_{ij} = 1\ \forall i, j$, $f(x) = x$ and $l = 1$ or $l = 2$.

4 Resolution approach: the Variable Neighborhood Search

This $n \times q$ continuous variables (x_{ik}) optimization problem is unfortunately not convex and may have a large number of local optima. For this reason, we decided to use a modern metaheuristic to solve it: the Variable Neighborhood Search (VNS).

The Variable Neighborhood Search is a metaheuristic, *i.e.,* a framework for building heuristics that may be applied to a wide variety of problems. Its principle is to alternate local search and various perturbations of the solution. In order to get better results, the local search part could involve some special strategies such as using a succession of local search algorithms (variable neighborhood descent) or nesting heuristic methods. In the current algorithm, we alternate a simple variable neighborhood search that has weak diversification properties, called Improved Local Search (ILS), with a true perturbation scheme P. The first may be viewed as a search intensification phase while the other is the diversification phase. Depending on the value of the magnitude parameter k, the perturbation will be more or less important, constraining the algorithm to first explore the close neighborhood of the best known solution to date (and to take advantage of the previous optimization unlike the multistart algorithm would do) and explore wider neighborhoods only if the close neighborhood fails.

The VNS implementation used here is the following:

Initialization
Let $k = 0$, choose a random initial solution X_0.
Set $S(X^*) \leftarrow S(X_0)$
While the stopping criterion is not met, do:
$X' \leftarrow ILS(X)$ (the solution obtained from X by improved local search).

If $S(X') < S(X^*)$:
$X^* \leftarrow X'$
$k \leftarrow 0$
else,
$k \leftarrow k + 1$
if $k \geq k_{max}$: $k \leftarrow 0$
$X \leftarrow P_k(X^*)$ (a solution obtained from X^* by a perturbation of magnitude k).
done.

4.1 Improved local search

The improved local search works in the same way as VNS, except that the perturbation is not intended to explore the neighborhood of the best known solution; its goal is rather to quickly drive the search to better solutions. If the moves considered for the "perturbations" at this step are useful to improve the search, they do not have the property to theoretically allow any solution to be found by the algorithm, even if they involve a certain level of randomization. This property is not required at this step but is necessary for the perturbation phase.

The first improved local search strategy, involves a *Barycenter move* and is defined as follows:

Initialization:
choose the value of m (2 here), k_ils (2 here), set $n\_ils = 0$ and set $s\_ils = S(X)$.

While $n\_ils < k\_ils$ **do:**
$s\_cur = LS(X)$
if $s\_cur < s\_ils$ do:
$s\_ils \leftarrow s\_cur$ and $n\_ils \leftarrow 0$
else
$n\_ils \leftarrow n\_ils + 1$
apply n_ils times:
Let *obj* be a random object
Move *obj* to the barycenter of its m nearest neighbors according to dissimilarity.
done.

The second improved local search strategy involves the *Line* move and is defined as follows:

Initialization:
choose the value of k_ils (2 here), set $n\_ils = 0$ and set $s\_ils \leftarrow S(X)$.
While $n\_ils < k\_ils$ **do:**

$s\_cur = LS(X)$
if $s\_cur < s\_ils$ do:
$s\_ils \leftarrow s\_cur$ and $n\_ils \leftarrow 0$
else
$n\_ils \leftarrow n\_ils + 1$
apply n_ils times:
Let i be a random object
Let j be another random object
Move i in the direction of j (forward or backward) so that $d_{ij} = \delta_{ij}$.
done.

The local search $LS(X)$ involved as the central part of any improved local search strategy is a gradient based steepest descent with decreasing stepsize as follows:

Initialization:
Set lr, lr_{min}, η_1, η_2 and max_imp.
While $lr \leq lr_{min}$ **do:**
compute $\overrightarrow{grad(X)}$
$X' \leftarrow X - lr\ \overrightarrow{grad(X)}$
If $S(X') \leq S(X)$
$imp \leftarrow imp + 1$
If $imp > max\_imp$ $\eta \leftarrow \eta_2$
else $\eta \leftarrow \eta_1$
$X \leftarrow X'$
else $\eta \leftarrow \eta_1$ and $imp = 1$
$lr \leftarrow \eta \times lr$
done.

Here we choose $lr = 0.1$, $lr_{min} = 1^{-8}$, $\eta_1 = 0.9$, $\eta_2 = 1.25$ and $max\_imp = 5$. If the solution is improved for more than 5 consecutive steps, the step size is increased, but is reduced when the solution gets worse. The principle of reducing the step size during optimization is a standard technique (used for example in neural networks) to help the search finding a local optima. Here, we also consider increasing the step size if it is too small (if the solution is improved each time, we are in the good direction and have no reason to be slow).

4.2 Perturbation Scheme

In variable neighborhood search, the perturbation scheme consists in applying increasing magnitude perturbations to the best known solution $P_k(X^*)$. As slightly moving the positions of each objects would not change the structure of the solution, it was decided that the magnitude of the perturbation would not be associated to the change in the coordinates of the objects but on the number of objects to move.

The following two perturbations are considered:

- The **STD** perturbation consists in applying k times the following: *Choose randomly an object obj and randomly change its coordinates.*
- The **EXC** perturbation: *Choose randomly two objects* obj_1 *and* obj_2 *and permute them.*

In practice, we noticed that small perturbations ($k_{max} = 2$ or 3) are enough to quickly improve the solution, even if this number needs to be increased a little for better performance on large problems.

5 Performance of the algorithm

5.1 Testing procedure

All the numerical results given in this section have been obtained on a SUN computer with AMD opteron 250 CPUs running at 2.4 GHz. The benchmark data is a transformed version of the "Morse code data" (Rothkopf, 1957). The symmetric dissimilarity matrix is deduced from the confusion matrix C by the same transformation as in Hubert, Arabie and Meulman (1997) and in Brusco (2001). The transformation is the following: $\delta_{ij} = \delta_{ji} = 2 - c_{ij} - cji$.

5.2 Numerical results

The following tables give a synthesis of the errors obtained for the Morse code data after 10 runs of 180 seconds CPU time. Multistart (MS) is compared to different implementations of Variable Neighborhood Search (involving the EXC or STD perturbations). Different improved locals search (BAR and LIN) are also compared to the basic local search (Basic). Each table shows the comparison for a different parameter p ($p = 1$ or $p = 2$) or an error criterion (sum of absolute deviation or sum of squared errors).

The numerical results obtained show that the Variable Neighborhood Search implementations significantly improves the quality of the solutions; this is particularly true if the "LIN" local search scheme is used together with the "STD" perturbation.

From the optimization point of view, these results show that minimizing the sum of squared errors is easier than minimizing the sum of the absolute deviations. Also, solving the problem with the Euclidian distance is easier than it is with the Manhattan distance. In the case of the minimization of the sum of squared errors with the Euclidian distance, the best known solution was found in almost every case (and when it was missed, the solution was very close).

Table 1: Results for 10 runs minimizing the sum of absolute deviation for the Morse code data with Minkowsky parameter 1

LS PERT	BAR EXC	LIN EXC	Basic EXC	BAR STD	LIN STD	Basic STD	BAR MS	LIN MS	Basic MS
Min	203.494	203.647	203.864	203.363	203.486	203.685	205.519	204.145	221.509
Median	203.582	203.79	213.065	203.412	203.6	208.117	208.423	207.377	228.002
Max	212.376	207.262	217.16	211.291	203.925	217.838	211.300	209.330	233.035
Avg	204.994	204.346	211.651	204.777	203.663	209.623	208.564	207.076	228.162

Table 2: Results for 10 runs the sum of squared errors for the Morse code data with Minkowsky parameter 1

LS PERT	BAR EXC	LIN EXC	Basic EXC	BAR STD	LIN STD	Basic STD	BAR MS	LIN MS	Basic MS
Min	153.474	154.061	153.763	153.806	153.634	153.599	154.338	155.335	157.742
Median	154.286	155.05	155.434	154.198	154.179	155.098	156.758	157.183	163.008
Max	155.125	157.43	160.965	158.063	158.29	160.702	159.775	158.309	166.781
Avg	154.385	155.388	155.689	154.953	154.732	156.599	156.99	157.204	162.779

Table 3: Results for 10 runs minimizing the sum of squared errors for the Morse code data with Minkowsky parameter 2

LS PERT	BAR EXC	LIN EXC	Basic EXC	BAR STD	LIN STD	Basic STD	BAR MS	LIN MS	Basic MS
Min	159.045	159.045	159.045	159.045	159.045	159.045	159.045	159.045	159.045
Median	159.045	159.045	159.045	159.045	159.045	159.045	159.045	159.045	159.045
Max	159.045	159.045	159.045	159.045	159.045	159.045	159.098	150.098	159.098
Avg	159.045	159.045	159.045	159.045	159.045	159.045	159.055	159.05	159.055

Table 4: Results for 10 runs minimizing the sum of absolute deviation for the Morse code data with Minkowsky parameter 2

LS PERT	BAR EXC	LIN EXC	Basic EXC	BAR STD	LIN STD	Basic STD	BAR MS	LIN MS	Basic MS
Min	244.716	245.653	245.009	244.7	245.57	245.042	244.701	245.952	247.676
Median	244.728	246.007	245.687	244.708	245.858	245.626	244.739	246.092	248.844
Max	244.857	246.249	246.575	244.771	246.139	246.004	245.497	246.562	250.308
Avg	244.749	245.958	245.739	244.721	245.841	245.667	244.926	246.228	249.023

6 Impact of the Minkowsky parameter upon the error measure

6.1 The Morse code data

As the program described above seems reliable and could handle any value for the Minkowsky parameter, it may be used to evaluate the impact of the Minkowsky parameter and find the value that best fits the data. The "Morse code" problem was solved for $p = 0.6$ to 3.6 with 0.1 increment. For $p < 1$, the measure could no more be considered as a distance but was however computed for the purpose of studying the response curve. In order to get best results (not to study the performance of the algorithm), the program was run for more than 10 minutes with various strategies. The obtained curve is not convex and has a local maxima for $p = 2$ as shows the Figure 1.

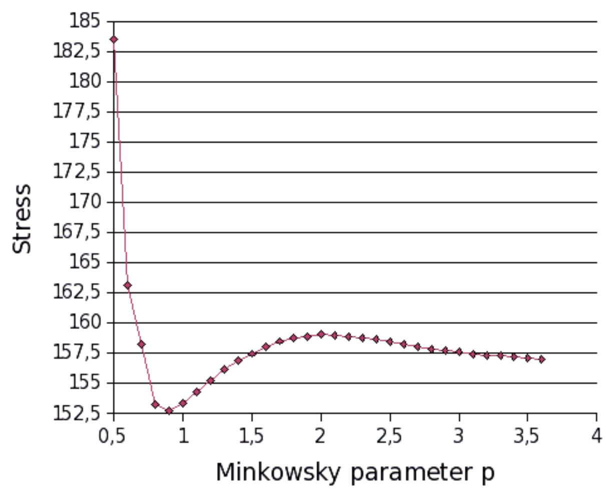

Figure 1: Stress according to the Minkowsky parameter p for the Morse code data

6.2 Generated data

In order to understand this phenomenon and to see wether it is due to the data itself, to a use of an inappropriate dimension, or to any other feature that may have an incidence on the results, the same test was then achieved with simulated data where all the external parameters have been eliminated.

Six data sets have been generated, each consisting in 20 objects with coordinates between 0 and 10 in a two dimensional space (in order to avoid the bias due to the dimension reduction) from which dissimilarities δ_{ij} were computed as distances either using Minkowsky parameter 1 or 2. The dissimilarities have then been randomly noised in following way: $\delta'_{ij} = \delta_{ij} * (1 - \frac{\alpha}{2} + u\alpha)$ where u is a uniform 0-1 random variable and $\alpha = 0.0, 0.1$ or 0.2. Figure 2 shows the results obtained according to the Minkowsky parameter.

7 Real data application

An experiment with a sample of 45 persons has been used in order to collect the necessary data to test the algorithm and to produce the different perceptual maps. Most of the

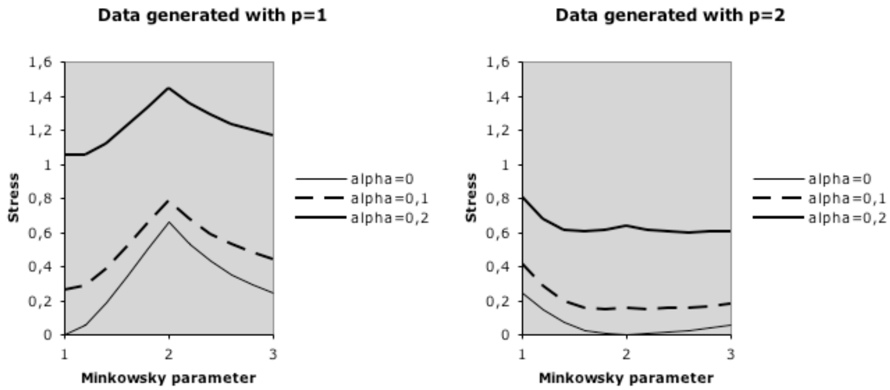

Figure 2: Stress according to the Minkowsky parameter p for the data generated with $p = 1$ (left) and $p = 2$ (right)

respondents are university students that accepted to contribute to the study as volunteers. The experiment was conducted on an individual basis. Each respondent had the task to evaluate the level of dissimilarity between pairs of objects on a scale that varies from 1 (very similar objects) to 100 (very different objects). The objects represent 8 brands of acetaminophen 325 mg caplets: 3 of them were national brands (Atasol, Tylenol and Excedrin) and the rest of them were private brands sold in Canadian pharmacies (Personnelle, Life, Equate, Truly, Option). In order to facilitate the task of data collection, a software (Simili)[4] has been developed allowing pictures of the products to be displayed with some details about the brand. A scale was presented for each pair of brands, and the respondent had to click with the mouse, on the level that corresponds the best to his dissimilarity perception. At each stage of the procedure, the data provided is directly recorded in a matrix.

Respondents were asked also to fill a section of the questionnaire with socio-demographic data and some data related to the use and the familiarity with the different brands. A graduate student had the task of explaining the procedure to the different respondents in order to guarantee the data quality.

The Stress is computed for the Minkowsky parameter $p = 1 \dots 3$ and the results are shown on Figure 3. From this graphic, one does not only notice that the Stress is minimized for $p = 1$, but comparing the shape of the curve to the curves of Figure 2 suggests that $p = 1$ is a better choice.

In order to see the impact of the Minkowsky parameter on the map, two perceptual maps are drawn, one with $p = 1$ (Figure 4) and $p = 2$ (Figure 5). Indeed these two maps

[4]The software is available from the authors upon request.

are close one to the other, but already with only 8 objects, one of them (Personnelle) has a significantly different position. The choice of a distance measure instead of the other could be misleading (and the change would increase with the number of objects).

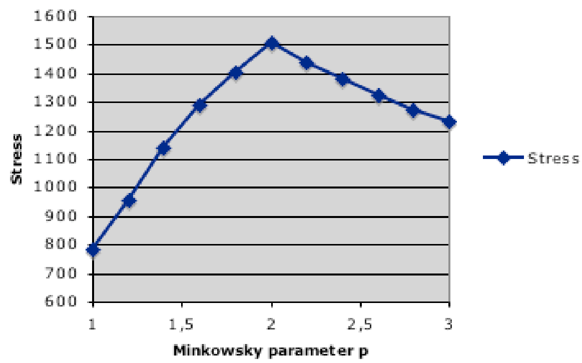

Figure 3: Stress according to the Minkowsky parameter p for the brand data

Figure 4: Map obtained with the stores/national brands data using $p = 1$

Figure 5: Map obtained with the stores/national brands data using $p = 2$

8 Conclusion and future research avenue

The algorithm proposed in this paper may easily handle various parameters and provides reliable results. These properties were useful when studying the impact of the Minkowsky parameter on the final Stress. The first remark was that the multidimensional scaling problem is easier to solve when $p = 2$ than it is for other values and minimizing the sum of squared errors is also easier to solve than the other error functions. The second remark is that the results seems to be better when $p \neq 2$ (the final error is smaller). Surprisingly, this second remark holds for each of the tests we did except for the noise free data ($\alpha = 0$) generated with $p = 2$ (in all the other cases, a local maximum is found when solving the problem with $p = 2$). If this is difficult to prove, it is probably due to the insensitiveness of the euclidian distance to the rotations. For any Minkowsky parameter other than 2, rotating the whole map changes the distances and may reduce the Stress. From the optimization point of view, the insensitiveness of the euclidian distance to the rotations is certainly an asset as any random solution could be expected to be closer to an optimal configuration than it would be if $p \neq 2$. In the case of noisy data, which is a reasonable assumption in psychology or in marketing, the Stress value could not be used alone to find out the best Minkowsky parameter. We therefore suggest to build the curve of Stress value as a function of p. If no strong conclusion could be made, it is likely that choosing $p = 2$ is not always the best choice. New experiments should be achieved to study the choice of the value of p, for example, it would be interesting to see wether the respondents themselves think the map obtained with $p = 1$ is better or not than the map obtained with $p = 2$. As a complement to this study, it would be interesting to see which meaning they give to each axis.

References

[1] T.H.A Bijmolt and M. Wedel The effects of alternative methods of collecting similarity data for Multidimensional Scaling *International Journal of Research in Marketing*, 12:363–371, 1995.

[2] M.J. Brusco. Morph-based local-search heuristics for large scale combinatorial data analysis. *Journal of Classification*, 16:163–180, 1999.

[3] M.J. Brusco. A simulated annealing heuristic for unidimensional and multidimensional (city-block) scaling of symmetric proximity matrices. *Journal of Classification*, 18:3–33, 2001.

[4] G. Caporossi and P. Hansen. Variable neighborhood search for least squares clusterwise linear regression. In *Proceedings of the 18th Mini Eurp Conference on Variable Neighborhood Search*, Tenerife, Spain, November 23–25, 2005.

[5] J.D. Carroll and P.E. Green. Psychometric methods in marketing research: Part II, multidimensional scaling. *Journal of Marketing Research*, 34(2):193–204, 1997.

[6] L.G. Cooper. A Review of Multidimensional Scaling in Marketing Research. *Applied Psychological Measurement*, 7:427–50, 1983.

[7] J. De Leeuw. Convergence of the majorization method for multidimensional scaling. *Journal of Classification*, 5:163–180, 1988.

[8] P.J.F. Groenen and W.J. Heiser. The tunneling method for global optimization in multidimensional scaling. *Psychometrika*, 61(3):529–550, 1996.

[9] P.J.F. Groenen, W.J. Heiser, and J.J. Meulman. Global optimisation in least-squares multidimensional scaling by distance smoothing. *Journal of Classification*, 16:225–254, 1999.

[10] P. Hansen and N. Mladenović. Variable Neighborhood Search: Principles and Applications. *European Journal of Operations Research*, 130:449–467, 2001.

[11] L. Hubert, P. Arabie, and M. Hesson-Mcinnis. Multidimensional scaling in the city-block metric: A combinatorial approach. *Journal of Classification*, 9:211–236, 1992.

[12] L. Hubert, P. Arabie, and J. Meulman. Linear and Circular Unidimentional Scaling for Symmetric Proximity Matrices. *British Journal of Mathematical and Statistical Psychology*, 50:253–284, 1997.

[13] J.B. Kruskal. Nonmetric multidimensional scaling: a numerical method. *Psychometrika*, 29:115–129, 1964.

[14] P.L. Leung and K. Lau. Estimating the city-block two-dimensional scaling model with simulated annealing. *European Journal of Operational Research*, 158:518–524, 2004.

[15] N. Mladenović and P. Hansen. Variable neighborhood search. *Computers and Operations Research*, 24:1097–1100, 1997.

[16] E. Z. Rothkopf. A Measure of Stimulus Similarity and Errors in Some Paired-Associate Learning Tasks. *Journal of Experimental Psychology*, 53:94–101, 1957.

[17] V.J. Stefflre. Market structure studies: new products for old markets and new markets (foreign) for old products. In *Application of the Sciences in Marketing*, F.M. Bass, C.W. King, and E.A. Pessimer, (eds.), New York: John Wiley & Sons, 251–268, 1969.

[18] W.S. Torgerson. *Theory and Methods of Scaling.* New York: John Wiley & Sons, 1958.