Proceedings of the 7th WSEAS International Conference on Evolutionary Computing, Cavtat, Croatia, June 12-14, 2006 (pp12-19)

PST*: A new Evolutionary Approach to topographic mapping

MASSIMO BUSCEMA, STEFANO TERZI
Semeion Research Centre of Science of Communication
Via Sersale 117, 0128 Rome
ITALY
http://www.semeion.it

Abstract: In this paper we approach the problem of Topographic Mapping. We introduce a new algorithm based on an evolutionary approach with very competitive performances in alternative to some algorithms found in literature. The Pick and Squash algorithm (P.S.T.) is also able to manage incomplete data and it is metric independent.

Keywords: Topographic Mapping, Multi Dimensional Scaling, Evolutionary Algorithms,

**The PST system presented in this article is conceived by Massimo Buscema and protected under international patent n. PCT/EP2004/051190, receipt 26.06.04, applicant Semeion Research Centre*

1 The Problem

The impossibility to supply efficient visualizations of the relationships between records and variables in a dataset is one of the major limits in the analysis of multidimensional data.

More in general, it is very useful an instrument able to compress the information contained in a data set from a K-dimensional space to a P-dimensional space (with P<<K and P=2 or P=3 in the case in which the objective is the graphic representation).

The problem that we want to approach is as follows: given a measure of the relationships existing between elements of a given space, we want to project these elements in a different space with less dimensions and minimizing the distortion of the original relationships.

This is a problem similar to the one solved by the Self Organizing Map (SOM) [9]. The important difference is in the approach: in the case of SOM we can speak of vector projection, that is, all the information in the process of projection are not considered for every step, that is, the positions of the other points, but a synthesis of this information, that is, the codebooks of the cells of the SOM. On the contrary, the proposed model in the paper considered implements the one that we might define as coordinated projection, which means, the positioning of the original points in a sub-space considering the concurrent presence of all the others. A problem of this kind is defined as "Topographic Mapping" in literature (TM) and there are several algorithms to treat it. The objective is to preserve the characteristics of the "geometrical structure" of data in a representation with reduced dimensionality.

The concept of "geometrical structure" is connected to the concept of distance. Therefore, to preserve the geometric structure of the original data means that, after applying the algorithm of mapping, the elements that used to be "close" in the original space find themselves to be close in the sub-space too. It means also those distant elements in the original space turn out to be still separated in the final destination space.

In the case in which the preservation of the specific metric the original space is important, the objective consists in aiming at the *isometry* between the original space and the resulting map.

We can also consider to be only relevant the conservation of the topological order between the two spaces; in this case, we tend only to keep the order of the distances between the points between the original and the final space.

In the case in which the objective of the analysis is the preservation of the metric of the original distances, the problem can be formalized as follows: *given a square matrix A of dimension N, symmetrical and with null diagonal, we want to position in a P-dimensional space, generally with P<K, N points so that we minimize an error function E that takes into account the dissimilarity between the matrix A and the matrix of distances between the*

N points projected in the P-dimensional subspace.

The problem can be presented in the following form: given N points $X = \{x_1,....,x_N\}$, or their distances in a K-dimensional space, find the distribution of these points $Y = \{y_1,....,y_N\}$ in a *P-dimensional* space with $P<K$, so that we minimize the "difference" between the original distances and those in the projected space.

If we define:

- the matrix of the map distances $Md(Y): Md_{ij} = D^P(y_i, y_j)$,
- the matrix of the original distances $Rd(X): Rd_{ij} = D^K(x_i, x_j)$,

and a measure of the dissimilarity between the two matrices $E = E(Md, Rd)$,

then the target function consists in finding a configuration of points $Y' = \{y'_1,....,y'_N\}$, such that:

$$E^* = \min\left[E(Md(Y'), Rd(X))\right] \quad (1)$$

This general problem generates an ample spectrum of sub-problems depending on the choices made for the distance *D* and on the target function *E*.

For the distance D, in particular, we can distinguish metric and non-metric distances (those that do not satisfy the triangular inequality).

More choices are possible for the target function too, for example:

$$E = \frac{2}{N(N-1)} \sum_{i=1}^{N} \sum_{j=i}^{N} \left| Md_{ij} - Rd_{ij} \right| \quad (2a)$$

$$E = \frac{2}{N(N-1)} \sum_{i=1}^{N} \sum_{j=i}^{N} \frac{\left| Md_{ij} - Rd_{ij} \right|}{\left| Rd_{ij} \right|} \quad (2b)$$

In the case of the target function (2a), the estimate of the error in the data projection on the map considers all the distances equally, in the second case (2b), instead, the distances among closer points "weigh" more in the evaluation of the quality of a solution. The equation (2b) is the notorious 'stress' equation used in many algorithms that deals with topographic mapping [13].

This problem can be still further classified as linear and non linear mapping according to the relationship existing between the coordinates of the original and projected points: the linear mapping is a mapping from an original space to a final space which maintains a linear relation between the axes of the original space and the axes of the final space.

The more common algorithms of linear projections are the PCA (Principal Component Analysis) [8] and the ICA (Independent Component Analysis) [4]; the former requires a gaussian distribution of data, while the latter does not require any specific distribution.

If the relationships between variables are non linear, the methods described above are not able to preserve, with adequate accuracy, the geometrical structure of the original space. In the compression, in fact, much important information is lost. It is evident, then, the necessity of both eliminating the requirement of linearity between axes and of finding a non linear mapping between the original and final space capable of preserving as much as possible the relationships between variables.

2 The PST Algorithm

To face the problem of the non-linear mapping, we choose an approach of an evolutionary kind, named as "Pick and Squash Tracking" (PST).

The novelty of many approaches to the Topographic Mapping (TM) is more in the cost function minimized for the projection, than in the algorithms used to approach the problem.

In terms of the optimization theory, the problem can be formalized, in general, as a problem of non-linear minimization; our approach does not assume the use of a particular measure of distance or of an error function. Our approach aims to underline the efficiency of a specific evolutionary algorithm in solving the problem on minimization. This ability is characterized by the known properties of evolutionary approach:

a. Parallel capability of the process;
b. Flexibility in the choice of the function to be minimized;
c. Absence of bounds;
d. Capability of avoiding local minima;
e. Efficiency in the optimization of multiple target functions [7,11].

We evaluate now the geometric and the algorithmic complexity of this problem and how to approach it with PST.

A possible way to evaluate the geometric complexity of this problem consists in making the hypothesis of the existence of an exact solution and, then, evaluates the complexity of the constructive algorithm needed to find it. We define:

a. state of the map, *S*, a configuration of points in the projected space that can be rotated and/or translated;
b. tolerance angle, α, the possible positions on the circumference that fix the distance between the point P_i and the point P_i+1 to be positioned.

If we have two points to project, the problem is trivial: every point along a circumference with

radius equal to the distance to the two points is a correct solution.

When the number of point is bigger than two the complexity grows dramatically. We find that the number of possible states S, considering the distances among N points, is equal to:

$$S = \alpha^{M} \qquad M = N - 2;$$

If we now define T the number of tests needed to verify the distances among N points in a state, then, we have:

$$T = \frac{M \cdot (M+1)}{2} \tag{3}$$

The equation is the sum of the number of comparison between the distances that is necessary to control in order to verify the consistency of every possible state of the map [5]. For example, for 4 points and a tolerance angle of 360°, the possible states of the map are 3602 and in order to verify each of these it is necessary to make [2·(2+1)]/2 comparisons. The number of possible tests for all the states of the network is then $Q = S \cdot T$

The PST is an evolutionary algorithm based on the algorithm GenD [1]. The space of projection is discretised, with a number of intervals sufficient to reach the optimal approximation to the solution. An analysis of the worst case shows that the error introduced by the discretisation for every point, with respect to the optimal solution in the continuum, is given by:

$$err_i = \sqrt{\sum_{k=1}^{M} \frac{l_k}{2}}$$

Where l_k is the dimension along the *k* axis of the discrete cell (Figure 1) (L_k dimension of the space of projection along the *k* axis divided by the number of elements of discretisation).

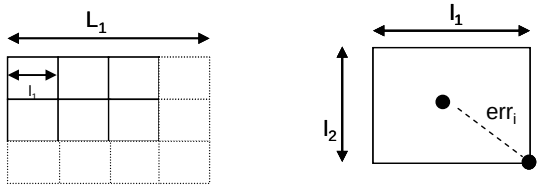


Figure 1

The total error for the projection of *N* points is given by:

$$ErrQ = \sum_{i=1}^{N} err_i = N \cdot \sqrt{\sum_{k=1}^{M} \frac{l_k}{2}} \tag{4}$$

Thus, it is possible to choose a discrete interval such that the error introduced by it is negligible.

As far as the choice of the dimension of the projected space is concerned it is sufficient to guarantee that in the destination space it can be inscribed an iperspheroid of diameter equal to the maximum distance present in the original data; we choose, for practical reasons, an hypercube of side equal to the maximum distance found in the data (MaxDim).

The stated problem has, then, an algorithmic complexity equal to the dispositions *n* over *k*, in other words, the ordered *k-uple* that can be built utilizing (without repetition) *k* among the n given objects, where *k* is the number of points to be placed on the map and *n* are the possible discrete cells where to place the points. Therefore, in order to obtain the projection of 10 points from a 3-dimensional space to a 2-dimensional space, quantized in squares of unitary side and with a quantization error of one thousandth of unity, we have:

$$l_1 = l_2 = l$$

$$ErrQ = 0.001 = N \cdot \sqrt{M \cdot \frac{l}{2}}$$

$$l = \frac{2}{M}\left(\frac{ErrQ}{N}\right)^2 = 6.6 \cdot 10^{-9}$$

The total number of elements, cells, that make up the discretised space is $\frac{1}{l}$=1.5*10^8.

The number of possible states, that is, of possible configurations of N points in the discrete space composed by *1/l* elements, is then:

$$S = D_N^{1/l} = \frac{(1/l)!}{(1/l - N)!} \tag{5}$$

with 1.5*10^8 cells, from which the possible solutions are the dispositions of 10 objects over 1.5*10^8 cells, that is 5.7*10^{81} possible states. Once coded in a discrete way the space of solutions it is, now, possible to define the coding individuals of the evolutionary algorithm (Figure 4):

a. Every individual encodes a complete solution of the problem, i.e. a set of *K* coordinates of the points on the projected space;
b. Every individual is constituted by *P* genomes, one for each dimension, each of length k;
c. The alphabet of each gene is equal and goes from 0 to MaxDim.

It is important to highlight that the operators of the evolutionary algorithm act independently in each dimension, because operations on genes involving different dimensions are not allowed.

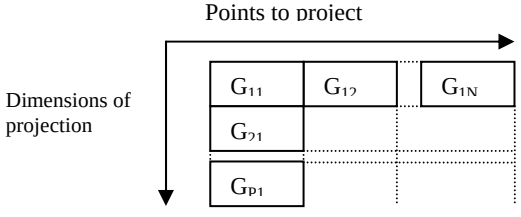


Figure 2: Individual coding

The evolutionary algorithm that we have chosen in order to approach the TM is GenD [1].

This choice is dictated by the particular efficiency shown by this algorithm in approaching problems with complex space of solutions.

The advantage of GenD in approaching complex optimization problems is due to one to its structural properties: the link of feedback between the average of the fitness of the population and the criteria of selection of the individuals that participate in the crossover [1].

The algorithm PST enriches GenD in two key points:

a. The creation of a new kind of crossover defined constructive crossover, in addition to the standard crossover already known ;

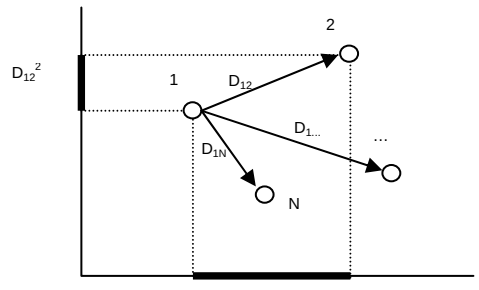
b. The creation of a new operator of optimization, called look around**,** is linked to the necessity of a more energetic exploitation of the local information.

Experimental results show that, thanks to the introduction of the operator "look around" and of the constructive crossover, the PST increases the velocity of convergence, and a bigger independence from the initialization conditions.

The constructive crossover is based on a construction of the kind greedy as a function of the local error.

We define "local error" of a gene the sum of the errors, calculated as differences of the original distances of each point from the distances of each projected point.

In the constructive crossover the new genome is built choosing from two "parents" the genes that locally correspond to a point with less local error (Figure 3).



Local Error of the point: $E_n = \sum_{i=1, i\neq n}^{N} D_{ni}$

Local Error of the gene: $E_{nj} = \sum_{i=1, i\neq n}^{N} D_{ni}^{j}$

Figure 3

The operator "look around" is an operator that implements a local search: it tries to explore in 8 directions, according to a random gradient, the new possible positions for each point (Figure 4). When a certain position increases the global fitness, then that position is kept. It is necessary to highlight that the already mentioned independence from the initial conditions is linked to this operator which allows, on the one side, to alter the boundaries defined by the initialization of the points and, on the other side, to modify the limits from 0 to MaxDim, making then the alphabet dynamic and virtually infinite.

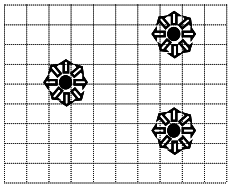


Figure 4: The potential displacement of the algorithm look around

---

Constructive Crossover:

1) Random choice of the first gene from one of the two parents;
2) Choice of one of the genes left (i);
3) Evaluation of the local error of the gene of the father with respect to the new genome ($E_{ij}(p)$);
4) Evaluation of the local error of the gene of the mother with respect to the new genome ($E_{ij}(m)$);
5) If $E_{ij}(p)<E_{ij}(m)$ copy the gene from the father else copy the gene of the mother;
6) If it has not used all the genes go to point 2)

Operator "look-around"

Repeat n times:

1) Random choice of one of the points;
2) Random choice of one of the 8 possible displacements;
3) Evaluation of the fitness of the new solution;
4) If the fitness is better than the previous one use the new genome.

3 Experimentations

Given the ample variability of objectives that we can set in approaching a problem of TM, it is useful to analyze the practical usage of these algorithms in order to allow, at least partially, a comparative analysis.

One of the most common usages of these algorithms is the compression of the dimension of the data in order to allow visualization on a 2-dimensional plan. In order to evaluate the performances of the PST, we have chosen a set of

problems of 2-dimensional projection and we have compared the results of the PST with those of some algorithms used traditionally to approach this kind of problems.

For each problem we have done ten experiments using, as criteria of evaluation, the measure $Fit = 1 - E$;

where $E = \frac{2}{N(N-1)} \sum_{i=1}^{N} \sum_{j=i}^{N} \frac{\left| Md_{ij} - Rd_{ij} \right|}{Rd_{ij}}$.

In the results we report:

a. The result of the best experiment, measure of the ability of solution of the algorithm.

b. The mean and the variance of the ten experiments, measure of the robustness of the algorithm.

The following datasets have been chosen as problems for the benchmark:

- ItalianCity10 – The highway distances of 10 Italian cities.
- Uk23 – The highway distances of 23 English cities.
- Usa12 – The air distance route among 12 USA cities.
- Food – A small database of the food consumption of European countries.
- Gang – An example database generated from the characters of the film "West Side Story"[12].
- Molecola25 – A problem of molecular structure [10].
- Iris150 – The famous database on the of Fischer's Iris [6].

The algorithms with which we compared the PST are:

- The Principal Component Analysis (PCA) [8]**:** algorithm of linear projection. The first two principal components have been used as projection axis. The comparison with the PCA has been done to highlight the different compression capability of the original information by a linear projection with respect to a non-linear.
- Sammon's mapping [13]: this algorithm tends to guarantee the preservation of topology. Sammon defines the error function:

$$E_{Sammon} = \frac{1}{\sum_{i,j=1}^{N} d_{i,j}^{k}} \left(\sum_{i,j=1}^{N} \frac{\left(d_{i,j}^{k} - d_{i,j}^{p} \right)^2}{d_{i,j}^{k}} \right)$$

where $d_{i,j}^{k}$ and $d_{i,j}^{p}$ are the distances between the *i-th* and the *j-th* vector, respectively in the *k*-dimensional space of immersion and the *p* dimensional space of projection. Given the error function, the optimal projection is calculated using the decreasing gradient algorithm. The only difference with respect to the objective function used by PST is a scaling factor that makes fully comparable the results obtained with the two different approaches.

- Direct Search [7]: is a method for solving the problem of optimization that does not require any information about the gradient of the objective function. A direct search algorithm analyses a set of solutions in the neighborhood of the current solutions, trying to find a solution with objective function better than the current one. In terms of flexibility of the objective function it is, then, comparable to PST and for this reason a comparison of the results is significant.

In the first instance we show the comparison among different algorithms (Table 1), showing the best result obtained in each of the test datasets:

BEST	**Ds**	**Pst**	**Pca**	**Sammon**
UK 23	0.964	**0.964**	*	0.9582
USA 12	0.987	**0.987**	*	0.9809
Italy 10	0.934	**0.974**	*	0.9662
Food 9	0.877	**0.879**	0.818	0.854
Food 16	0.877	**0.877**	0.813	0.8624
Gang 14	0.841	**0.842**	0.792	0.8036
Gang 27	0.809	**0.814**	0.741	0.7831
Molecola 25	0.94	**0.982**	*	0.9809
Iris 150	0.339	**0.928**	0.902	0.9247

Table 1**:** Best results reached by each algorithm in 10 tests (* indicates the impossibility of the algorithm analyzed to solve the problem being available only the matrix of the distances and not the original points)

The significant performances of the PST is shown by the comparison that highlights the average result that each algorithm has obtained in each of the test (Table 2):

Average	**Ds**	**Pst**	**Pca**	**Sammon**
UK 23	0.955	**0.964**	*	0.9495
USA 12	0.975	**0.987**	*	0.9707
Italy 10	0.868	**0.973**	*	0.9556
Food 9	0.867	**0.879**	0.818	0.8401
Food 16	0.874	**0.877**	0.813	0.8524
Gang 14	0.836	**0.842**	0.792	0.7982
Gang 27	0.796	**0.814**	0.741	0.7787
Molecola 25	0.926	**0.98**	*	0.9707
Iris 150	0.285	**0.928**	0.902	0.9175

Table 2 Average results of each algorithm in 10 tests.

Significativity of mean difference is reported in Table 3.

Pst_p	DS		PCA		Sammon	
	M. Diff	*Pvalue*	*M. Diff*	*Pvalue*	*M. Diff*	*Pvalue*
Uk23	0.01	0.071			0.007	0
Usa12	0.012	0.016			0.017	0
Ita10	0.105	0			0.018	0
Food9	0.012	0.016	0.061	0	0.039	0
Food16	0.003	0.001	0.064	0	0.024	0
Gang14	0.006	0.057	0.05	0	0.044	0
Gang27	0.018	0	0.072	0	0.035	0
Mol25	0.055	0			0.01	0.002
Iris150	0.643	0	0.026	0	0.01	0

Table 3: Significativity of mean differences between PST algorithm and other algorithms. *MeanDiff* is the difference between means of PST results and other algorithm results; *Pvalue*, is the p-value associated with the t-statistic:

$$T = \frac{\bar{x} - \bar{y}}{s\sqrt{\frac{1}{n} + \frac{1}{m}}}$$

where s is the pooled sample standard deviation and n and m are the numbers of observations in the x and y samples.

3.1 PST versus PCA

The comparison with the PCA brings out two particular aspects:

a. On the one side, as we expected from the bigger freedom in the projection on the map, the PST shows a net improvement in the measure of the fitness with respect to PCA. In fact, the PST, as shown in figures 5 and 6 for the problem FOOD, allows a better compression of the useful information;

b. On the other side, the PCA is not applicable starting from a simple matrix of the distances.

PCA

(Variables Space)

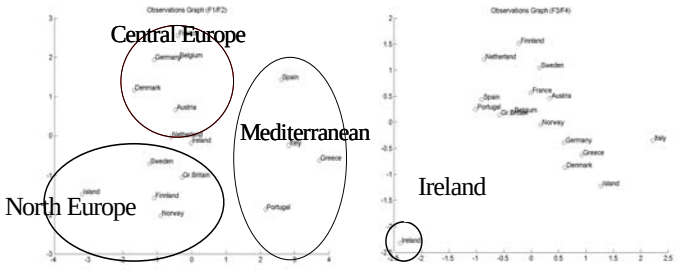


Figure 5

PST

(Variables Space)

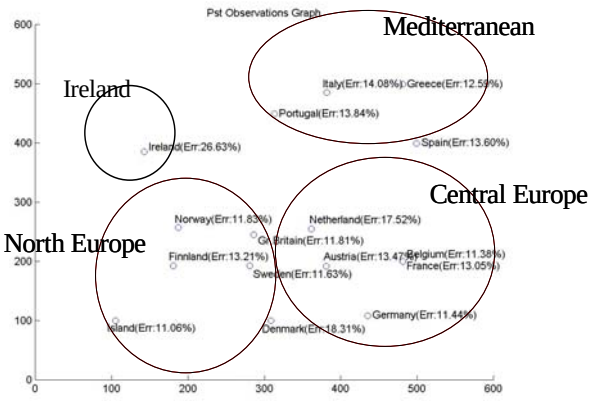


Figure 6

3.2 PST versus Sammon

The Sammon Mapping uses a functional cost very similar to the one used by PST. The results show that on all the analyzed datasets, PST obtains the best performances, both from the point of view of result and with respect to the robustness (in two cases, indeed, the variance of the results of the Sammon mapping is different from zero).

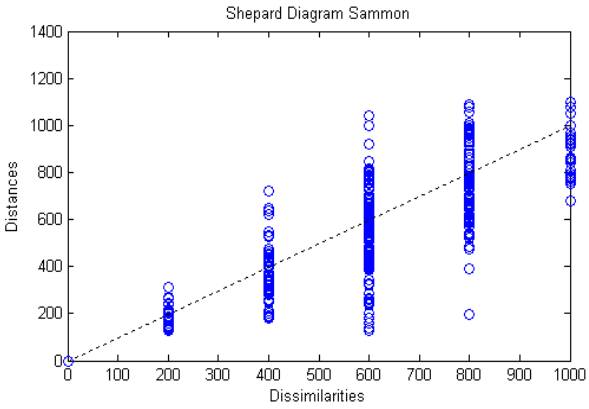


Figure 7a

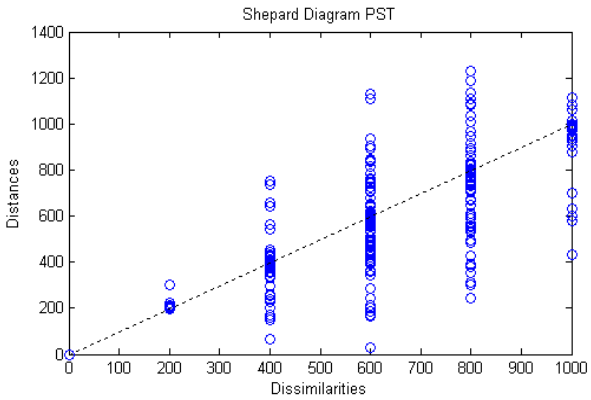


Figure 7b

Figure 7a and 7b report the comparison on the dataset gang27, using the Sammon mapping and the PST; the best results from the two systems are reported. From the Shepard diagram, scatter diagram which reports on the x axis the distances in the original space and on the y axis the distances of the mapping, we see that the PST as clearly a better performance on the short distances.

3.3 PST versus Direct Search

In the case of the Direct Search, the comparison is linked to the analyses of efficiency in the solution of the problem of optimization.

The results are interesting and highlight that on 10 tests on each data set the best results are sometimes comparable (even though PST shows better performances on many datasets, in particular on the problem of the 150 Iris).

In addition, the analyses of the average of the results shows a difference even more marked in favor of PST.

In conclusion, the choice of an evolutionary algorithm, in particular of GenD, is a more efficient choice (better overall results), more robust (better repeated results) and more powerful (better repeated results obtained on very different datasets) than the Direct Search.

An additional edge element of PST, with respect to the two algorithms discussed (also with respect to others, like CCA and the CDA, which have not been taken into consideration because they optimize different cost function), is the high flexibility in the choice of typology of distance used for the mapping and in the cost function. PST shares this advantage with the Direct Search.

For instance, if the optimizations just carried out should be repeated with a cost function that does not optimize the metric distance but only the topology of the points (the order in the neighborhood of each point from all the others), then, the PST would not imply any modification, allowing then a projection composed exclusively of variables of ordinal kind.

3.4 PST On molecule conformation problem

In this task we use PST to solve a Molecule Conformation Problem: the target is to arrange the N atoms of a molecule in a way that the distances between specified pairs of atoms match experimental data. We compare PST performances with an algorithm for minimization of an unconstrained multivariable function (fminunc) offered by Matlab Optimization Toolbox [10], this algorithm is a subspace trust region method and is based on the interior-reflective Newton method described in [2],[3]. Each iteration involves the approximate solution of a large linear system using the method of preconditioned conjugate gradients (PCG).

This task is different from the formers due to the lack of some of the distances, not all the distances among atoms were available. All the fitness and performance measures are calculated on the available values.

Molecole	Fminunc 2D	PST 2D	PST 3D
MeanAbsErr	0.0173	0.009374	0.003262
MeanPercErr	0.0778	0.018	0.007

Table 4 Results of molecule conformation problem

In table 4 has been reported the performances of both the systems: the fitness criteria of PST (MeanPercErr) and the mean of $[\|y_i - y_j\|^2 - Rd_{ij}^2]^2$ used as error criteria by the matlab function.

In figure 8 we can see the differences on performance in a graphic way; using both measures performance of PST are clearly better.

We have also done a reconstruction of the problem in the original 3D space using PST getting an error very low.

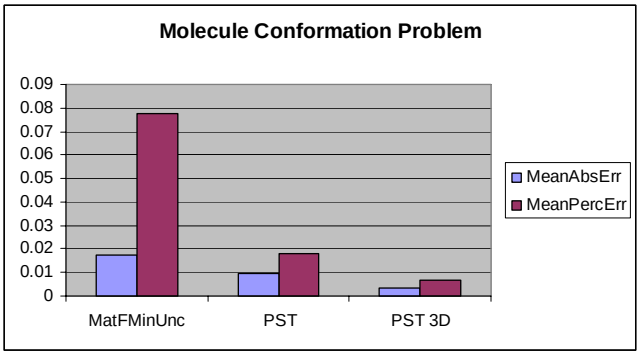


Figure 8 Bar plot of results on Molecule conformation problem

4 Conclusions and Future Perspectives

The approach shown to tackle the general problem of Topographic Mapping, presents two important elements: on the one side, it emphasizes the advantage to tackle the problem from the point of view of optimization, on the other side, it suggests the usage of a particularly efficient evolutionary algorithm to solve it.

The experimental results show, in empirical way, that the space of solutions of the problem can be particularly complex and depends strongly on the data and on the function with which we measure the distortion of the original distances from those on the map. For this reason, the usage of an algorithm, that

has excellent capacity to tackle different optimization problems and is not bound to specific properties of the problem itself, becomes fundamental.

GenD, the evolutionary algorithm used, shows the characteristic advantages of the algorithm of its class, elevated robustness on very different problems and, as underlined, allows hybridization that increase in a sensible way the velocity of convergence to the solution, traditional limit of the evolutionary algorithm.

The very positive results of the reported experiments suggest that the usage of the evolutionary algorithms in the field of Topographic mapping is a road to follow extending the analysis to other kind of distances and objective functions.

We have not done benchmark versus Isomap[14] and CCA/CDA that can be considered the latest algorithm on the topic because they are mostly involved in measuring the distances on the "right" manifold. These can be seen as a kind of preprocessing followed by a "classical" MDS algorithm; so PST process can, for example, replace classical MDS after the "on graph" distance measure carried out in Isomap. A possible improvement of PST would be to find a fitness function able to gain similar results to Isomap and CCA without the need of a manifold structure identification process; to this task we will devote our future work on PST.

Furthermore, another necessary element to complete the approach discussed is the development of a method to use the projection generated on new data, in other words, on an algorithm capable of projecting new values as a function of the solution produced by GenD and, thus, to allow, for example, the usage of this instrument also as a step of pre-processing of data for the compression of the cardinality of the input of the systems of classification and functional approximation.

In the case of very big set of data, of the order of thousands of records, this approach, as others discussing in literature, has problems in terms of computation time and difficulty of convergence; in this sense it is possible to tackle the problem using the SOM, for the projection of the entire dataset and recover part of the information that the SOM is not able to produce, applying the PST on the codebooks generated by the SOM. In this case, it is possible to visualize on a plan not the relation of each single element but the relations of groups of elements which belong to the same cell of the SOM.

References:

[1] Buscema, M. (2004). Genetic Doping Algorithm (GenD). Theory and Applications. *Expert Systems*, 21 (2): 63-79.

[2] Coleman, T.F. and Y. Li, "An Interior, Trust Region Approach for Nonlinear Minimization Subject to Bounds, *SIAM Journal on Optimization*, Vol. 6, pp. 418-445, 1996.

[3] Coleman, T.F. and Y. Li, "On the Convergence of Reflective Newton Methods for Large-Scale Nonlinear Minimization Subject to Bounds," *Mathematical Programming*, Vol. 67, Number 2, pp. 189-224, 1994.

[4] Comon, P. (1994). Indipendent component analysis – a new concept? *Signal Processing*, 36: 287-314.

[5] Dennis, J. and Torczon, V. (1994). Derivative-free pattern search methods for multidisciplinary design problems, paper AIAA-94-4349 in *Proceedings of the 5th AIAA/ USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Panama City, FL, Sept. 7-9, pages 922-932.

[6] Fisher, R. A. (1936). The Use of Multiple Measurements in Axonomic Problems. *Annals of Eugenics 7*, 179-188.

[7] Goldberg, D.E. (1989) *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley.

[8] Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, *24*.

[9] Kohonen, T. (2001). *Self-Organizing Maps*, Springer.

[10] Matlab, Copyright 1984-2005 Mathworks,Inc.

[11] Mitchell M. (1996). *An Introduction to Genetic Algorithms*, MIT Press Cambridge, MA, London, England.

[12] Rumelhart, J.L., McClelland, D.E. (1986). *Parallel Distributed Processing*, The MIT Press, Cambridge, MA, London, England.

[13] Sammon, J. W. (1969). A nonlinear mapping for data structure analysis. *IEEE Transactions on computers C*, 18 (5): 401- 409.

[14] J. B. Tenenbaum, V. De Silva and J. C. Langford (2000). A global geometric framework for nonlinear dimensionality reduction. *Science* 290 (5500), 2319-2323