Chapter 9
Solution Space Visualization

9.1 Introduction

Visualization is the discipline of analyzing and designing algorithms for visual representations of information to reinforce human cognition. It covers many scientific fields like computational geometry or data analysis and finds numerous applications. Examples reach from biomedical visualization and cyber-security to geographic visualization, and multivariate time series visualization. For understanding of optimization processes in high-dimensional solution spaces, visualization offers useful tools for the practitioner. The techniques allow insights into the working mechanisms of evolutionary operators, heuristic components, and their interplay with fitness landscapes.

In particular, the visualization of high-dimensional solution spaces is a task not easy to solve. The focus of this chapter is the mapping with a dimensionality reduction function F from a high-dimensional solution space $\mathbb{R}^d$ to a low-dimensional space $\mathbb{R}^q$ with $q = 2$ or 3 that can be visualized. Modern dimensionality reduction methods like ISOMAP [1] and LLE [2] that have proven well in practical data mining processes allow the visualization of high-dimensional optimization processes by maintaining distances and neighborhoods between patterns. These properties are particularly useful for visualizing high-dimensional optimization processes, e.g., with two-dimensional neighborhood maintaining embeddings.

Objective of this chapter is to show how ISOMAP can be employed to visualize evolutionary optimization runs. It is structured as follows. Section 9.2 gives a short introduction to ISOMAP. In Sect. 9.3, the dimensionality reduction-based visualization approach is introduced. Section 9.4 presents related work on visualizing evolutionary runs. Exemplary runs of ES are shown in Sect. 9.5. Conclusions are drawn in Sect. 9.6.

O. Kramer, *Machine Learning for Evolution Strategies*,
Studies in Big Data 20, DOI 10.1007/978-3-319-33383-0_9

9.2 Isometric Mapping

ISOMAP is based on multi-dimensional scaling (MDS) [3] that estimates the coordinates of a set of points, of which only the pairwise distances δ_{ij} with $i, j = 1, \ldots, N$ and $i \neq j$ are known. ISOMAP uses the geodesic distance, as the data often lives on the surface of a curved manifold. The geodesic distance assumes that the local linear Euclidean distance is reasonable for close neighboring points, see Fig. 9.1. First, ISOMAP determines all points in a given radius ϵ, and looks for the k-nearest neighbors. The next task is to construct a neighborhood graph, i.e., to set a connection to a point that belongs to the k-nearest neighbors and set the corresponding edge length to the geodesic distance. As next step, ISOMAP computes the shortest paths between any two nodes using Dijkstra's algorithm. In the last step, the low-dimensional embeddings are computed with MDS using the previously computed geodesic distances. ISOMAP is based on MDS, which estimates the coordinates of a set of points, while only the distances are known. Let $\mathbf{D} = (\delta_{ij})$ be the distance matrix of a set of patterns with δ_{ij} being the distance between two patterns $\mathbf{x}_i$ and $\mathbf{x}_j$. Given all pairwise distances δ_{ij} with $i, j = 1, \ldots, N$ and $i \neq j$, MDS computes the corresponding low-dimensional representations. For this sake, a matrix $\mathbf{B} = (b_{ij})$ is computed with

$$b_{ij} = -\frac{1}{2}[\delta_{ij}^2 - \frac{1}{N}\sum_{k=1}^{N}\delta_{kj}^2 - \frac{1}{N}\sum_{k=1}^{N}\delta_{ik}^2 + \frac{1}{N^2}\sum_{k=1}^{N}\sum_{l=1}^{N}\delta_{kl}^2]. \tag{9.1}$$

The points are computed via an eigendecomposition of $\mathbf{B}$ with Cholesky decomposition or singular value decomposition resulting in eigenvalues λ_i and corresponding eigenvectors $\gamma_i = (\gamma_{ij})$. It holds $\sum_{j=1}^{N}\gamma_{ij}^2 = \lambda_i$. The embeddings in a q-dimensional space are the eigenvectors of the q-largest eigenvalues $\hat{\mathbf{x}}_i = \gamma_i\sqrt{\lambda_i}$. Figure 9.2 shows an example of MDS estimating the positions of a set of 10-dimensional candidate positions based on distances between the points. The original points have been generated by a (1+1)-ES optimizing the Sphere function. The blue estimated points are

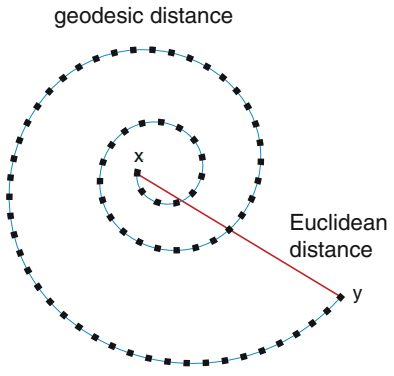

Fig. 9.1 Illustration of geodesic distance measure

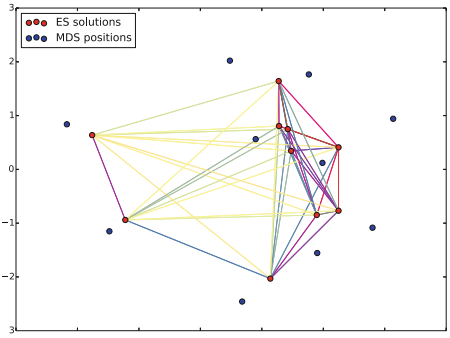

Fig. 9.2 MDS on 10-dimensional patterns generated during ES-based optimization. The *red dots* show the original ES positions of the candidate solutions (first two dimensions), the *blue dots* show the MDS estimates

located close to the original points generated by the ES concerning the first two dimensions. However, the estimation differs from the original positions, as it also considers the remaining eight dimensions.

An example for the application of MDS is depicted in Fig. 9.3. The left part shows the Swiss Roll data in three dimensions, while the right part shows the embedded points computed with MDS. Points that are neighboring in data space are neighboring in the two-dimensional space. This is an important property for our visualization approach.

ISOMAP does not compute an explicit mapping from the high-dimensional to the low-dimensional space. Hence, the embedding of further points is not possible easily. An extension with this regard is incremental ISOMAP [4]. It efficiently updates the solution of the shortest path problem, if new points are added to the data set. Further, the variant solves an incremental eigendecomposition problem with increasing distance matrix.

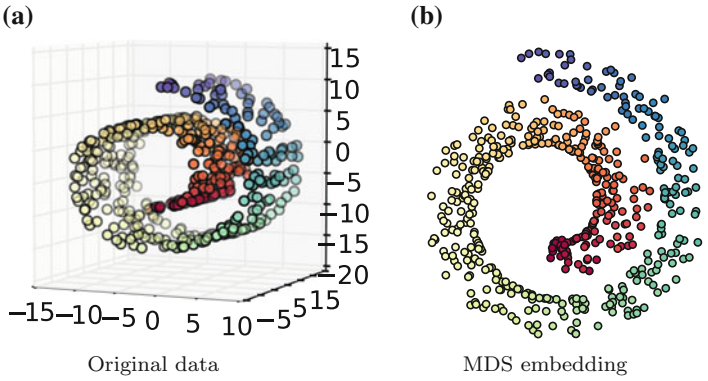

Fig. 9.3 Illustration of dimensionality reduction with MDS: **a** Swiss roll data set and **b** its MDS embedding

Similar to PCA in the previous chapter, SCIKIT-LEARN allows the easy integration of ISOMAP, which is sketched in the following.

- `from sklearn import manifold` imports the SCIKIT-LEARN manifold package that contains ISOMAP, LLE, and related methods.
- `manifold.Isomap(...).fit_transform(X), n_neighbors` fits ISOMAP to the training set of patterns `X` and maps them to a q-dimensional space with neighborhood size `n_neighbors`, which corresponds to k in our previous description.

In the following, ISOMAP is used to map the solution space to a two-dimensional space that can be visualized. Mapping into a three-dimensional space for visualization in 3d-plots is also a valid approach.

9.3 Algorithm

Dimensionality reduction methods map patterns from a high-dimensional data space, e.g., $\mathbb{R}^d$, to a low-dimensional latent space $\mathbb{R}^q$ with $q \ll d$. Objective of the dimensionality reduction process is to maintain most information like distances and neighborhoods of patterns. This is also discussed in Chap. 8. Again, we treat the solution space $\mathbb{R}^d$ as data space to visualize the optimization process in two dimensions. The dimensionality reduction-based visualization process is shortly sketched in the following algorithm.

Algorithm 7 VIS-ES

1: (1+1)-ES on $f \rightarrow \{\mathbf{x}_i\}_{i=1}^N$
2: dim. red. (ISOMAP) $F(\mathbf{x}_i) \rightarrow \{\hat{\mathbf{x}}_i\}_{i=1}^N$
3: convex hull of $\{\hat{\mathbf{x}}_i\}_{i=1}^N \rightarrow H$
4: generate meshgrid in $H \rightarrow \Gamma$
5: train $\hat{f}$ with $(\hat{\mathbf{x}}_1, f(\mathbf{x}_1)) \ldots, (\hat{\mathbf{x}}_N, f(\mathbf{x}_N))$
6: interpolate contour plot $\hat{f}(\gamma) : \forall \gamma \in \Gamma$
7: track search with lines $L_i : \mathbf{x}_i$ to $\mathbf{x}_{i+1}$

Algorithm 7 shows the pseudocode of our visualization approach, which we call VIS-ES in the following. First, the (1+1)-ES performs the optimization run on fitness function f. We visualize the last N generations, which constitute the training set of patterns $\{\mathbf{x}_i\}_{i=1}^N$. The patterns are embedded into a two-dimensional latent space ($q = 2$) with ISOMAP and neighborhood size k. Other dimensionality reduction methods can be used as well, e.g., we compare to PCA and LLE in [5].

The embedding process results in a set of low-dimensional representations $\{\hat{\mathbf{x}}_i\}_{i=1}^N$ of all candidate solutions. In our visualization approach, embeddings are part of the plots and drawn with circles that are colored according to their fitness.

The next steps of the VIS-ES aim at computing an interpolation of the contour plot based on the fitness in the low-dimensional space. The task is to interpolate from the non-uniformly spaced points to a mesh-grid of points to allow a continuous contour plot of fitness values within the convex hull of embedded candidate solutions. For this sake, first a convex hull H for set $\{\hat{\mathbf{x}}_i\}_{i=1}^N$ is computed. A meshgrid, i.e., equally distributed points within the maximum and minimum of coordinates of all $\hat{\mathbf{x}}_i$, serves as basis of the contour lines. All points of this meshgrid within H are basis of the contour line point set Γ. The low-dimensional representations and the corresponding fitness values of their solution space pendants are basis of the training set $(\hat{\mathbf{x}}_1, f(\mathbf{x}_1)), \ldots, (\hat{\mathbf{x}}_N, f(\mathbf{x}_N))$ for the interpolation model $\hat{f}$. This is a regression function that interpolates the surface of fitness landscape f based on the non-uniformly spaced patterns, i.e., $\hat{f}(\gamma)$ is computed for all $\gamma \in \Gamma$. Our implementation of the contour line computation is based on the MATPLOTLIB [6] method GRIDDATA, which can use natural neighbor interpolation based on Delaunay triangulation and linear interpolation. This is also a regression task, but the stack of algorithms is tailored to interpolation in two dimensions.

Last, a set of lines L_i is computed that tracks the evolutionary run by connecting solutions $(\mathbf{x}_i, \mathbf{x}_{i+1})$ of consecutive generations i and $i+1$ with $i = 1, \ldots, N-1$. When employing populations, we recommend to connect the best solutions of consecutive generations like in [5].

9.4 Related Work

An early overview of visualization techniques for evolutionary methods is presented by Pohlheim [7]. He proposes to employ MDS to visualize evolutionary processes. This is demonstrated for visualizing examples of candidate solutions of a population and of non-dominated solutions in multi-objective optimization. MDS-based visualization is appropriate to visualize evolution paths, but lacks from a visualization of the surrounding solution space.

A further early work in this line of research is the visualization of evolutionary runs with self-organizing maps (SOMs) [8]. Also Lotif [9] employs SOMs to visualize evolutionary optimization processes with a concentration on different parameter configurations. SOMs do not compute pattern-by-pattern embeddings like ISOMAP and related methods, but deliver a vector-quantization like placement of weight vectors in data space. These can be used to visualize important parts of data space on the trained map.

Volke et al. [10] introduce an approach for visualizing evolutionary optimization runs that can be applied to a wide range of problems and operators including combinatorial problems. It is based on steepest descent and shortest distance computations in the solution space.

In [11], an adaptive fitness landscape method is proposed that employs MDS. The work concentrates on distance measures that are appropriate from a genetic operator perspective and also for MDS. Masuda et al. [12] propose a method to visualize multi-objective optimization problems with many objectives and high-dimensional decision spaces. The concept is based on distance minimization of reference points on a plane.

Jornod et al. [13] introduce a visualization tool for PSO for understanding PSO processes for the practitioner and for teaching purposes. The visualization capabilities of the solution space are restricted to selecting two of the optimization problem's dimensions at a time, but allow following trajectories and showing fitness landscapes.

Besides visualization, sonification, i.e., the representation of features with sound, is a further way to allow humans the perception of high-dimensional optimization processes, see e.g., Grond et al. [14]. Dimensionality reduction methods can also be directly applied in evolutionary search. For example, Zhang et al. [15] employ LLE in evolutionary multi-objective optimization exploiting the fact that a Pareto set of a continuous multi-objective problem lives in piecewise continuous manifolds of lower dimensionality.

9.5 Experimental Analysis

The VIS-ES is experimentally analyzed in the following. We show the visualization results on our set of benchmark problems. The visualization methods we employ are based on MATPLOTLIB methods, e.g., PLOT and SCATTER. The following figures show the search visualization of a (1+1)-ES with a constant mutation strength of $\sigma = 1.0$ and Rechenberg's step size control on the Sphere function, on the Cigar, on Rosenbrock, and on Griewank.

The visualization is based on the dimensionality reduction of the last $N = 20$ candidate solutions with ISOMAP and neighborhood size $k = 10$. Figure 9.4 shows that the spherical conditions of the optimization process on the 10-dimensional Sphere function is conveniently visualized in the two-dimensional space. On the Cigar function, the walk in the valley can clearly be followed, in case of the Rechenberg variant, see Fig. 9.5, the solution space is significantly narrower. Also on Rosenbrock and on Griewank, the walk in the solution space can appropriately be visualized (Figs. 9.6 and 9.7).

Visualization techniques are difficult to evaluate. To support the observation that the proposed approach maintains high-dimensional properties, we analyze the embeddings w.r.t. the co-ranking matrix measure $E_{NX}(K) \in [0, 1]$ by Lee and Verleysen [16]. It measures the fraction of neighbors of each pattern that occur in a K-neighborhood in data and in latent space. High values for E_{NX} show that the high-dimensional neighborhood relations are preserved in latent space. Our analysis assumes that the maintenance of high-dimensional neighborhoods of evolutionary runs in the low-dimensional plots is an adequate measure for their quality. We analyze the co-ranking matrix measure of the last 100 solutions of 1000 generations of

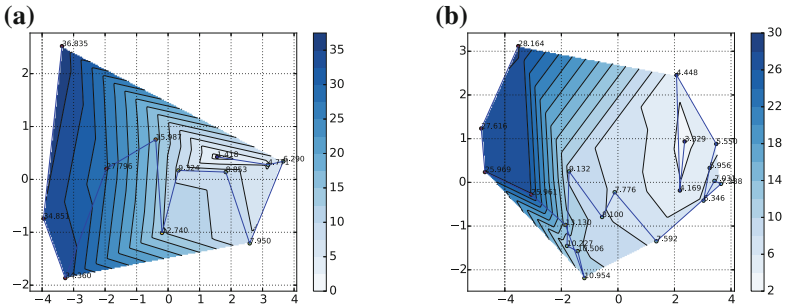

Fig. 9.4 Visualizing of (1+1)-ES run on the Sphere function, $N = 10$ **a** with constant step size, **b** with Rechenberg step size adaptation

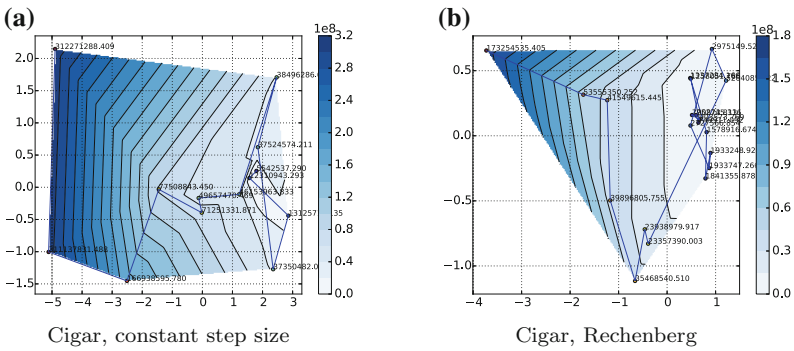

Fig. 9.5 Visualizing of (1+1)-ES run on the Cigar function, $N = 10$ **a** with constant step size, **b** with Rechenberg step size adaptation

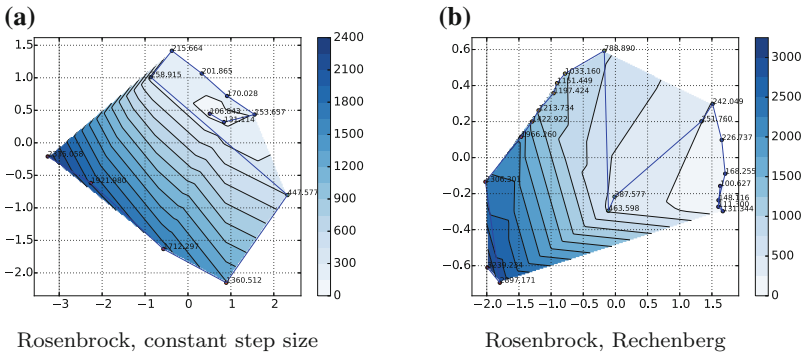

Fig. 9.6 Visualizing of (1+1)-ES run on Rosenbrock with $N = 10$ and **a** with constant step size, **b** with Rechenberg step size adaptation

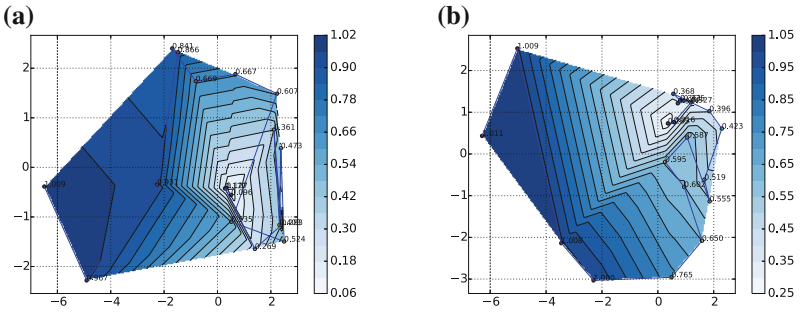

Fig. 9.7 Visualizing of (1+1)-ES run on the Griewank, $N = 10$ **a** with constant step size, **b** with Rechenberg step size adaptation

Table 9.1 Co-ranking matrix measure of the VIS-ES based on a (1+1)-ES and the COV-ES while embedding 100 solutions of a 20-dimensional evolutionary runs on five benchmark functions

Problem	Sphere	Cigar	Rosenbrock	Rastrigin	Griewank
(1+1)-ES	0.739	0.766	0.811	0.763	0.806
COV-ES	0.806	0.792	0.760	0.829	0.726

the VIS-ES on the Sphere, Cigar, Rosenbrock, Rastrigin, and Griewank with $d = 20$, see Table 9.1. The first line of the table shows the results of the VIS-ES based on ISOMAP with neighborhood size $k = 10$ and the (1+1)-ES, see Algorithm 7. The optimization part of the second line is based on the COV-ES, see Chap. 3. For E_{NX}, we also use a neighborhood size of $k = 10$. The results show that ISOMAP achieves comparatively high values when embedding the evolutionary runs reflecting a high neighborhood maintenance. This result is consistent with our previous analysis of the co-ranking measure for the embedding of evolutionary runs in [5].

9.6 Conclusions

Visualization has an important part to play for human understanding and decision making. The complex interplay between evolutionary runs and multimodal optimization problems let sophisticated visualization techniques for high-dimensional solution spaces become more and more important. In this chapter, we demonstrate how ISOMAP allows the visualization of high-dimensional evolutionary optimization runs. ISOMAP turns out to be an excellent method for maintaining important properties like neighborhoods, i.e., candidate solutions neighboring in the high-dimensional solution space are neighboring in latent space. It is based on MDS and graph-based distance computations. The success of the dimensionality reduction process of the search is demonstrated with the co-ranking matrix measure that indicates the ratio of coinciding neighborhoods in high- and low-dimensional space.

Further dimensionality reduction methods can easily be integrated into this framework. For example, PCA and LLE showed promising results in [5]. The interpolation step for the colorized fitness visualization in the low-dimensional space can be replaced, e.g., by regression approaches like kNN or SVR. Incremental dimensionality reduction methods allow an update of the plot after each generation of the (1+1)-ES. In practice, the visualization can be used to support the evolutionary search in an interactive manner. After a certain number of generations, the search can be visualized, which offers the practitioner the necessary means to evaluate the process and to interact with the search via parameter adaptations.

References

1. Tenenbaum, J.B., Silva, V.D., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. Science **290**, 2319–2323 (2000)
2. Roweis, S.T., Saul, L.K.: Nonlinear dimensionality reduction by locally linear embedding. Science **290**, 2323–2326 (2000)
3. Kruskal, J.: Nonmetric multidimensional scaling: a numerical method. Psychometrika **29**, (1964)
4. Law, M.H.C., Jain, A.K.: Incremental nonlinear dimensionality reduction by manifold learning. IEEE Trans. Pattern Anal. Mach. Intell. **28**(3), 377–391 (2006)
5. Kramer, O., Lückehe, D.: Visualization of evolutionary runs with isometric mapping. In: Proceedings of the IEEE Congress on Evolutionary Computation, CEC 2015, pp. 1359–1363. Sendai, Japan, 25–28 May 2015
6. Hunter, J.D.: Matplotlib: a 2d graphics environment. Comput. Sci. Eng. **9**(3), 90–95 (2007)
7. Pohlheim, H.: Multidimensional scaling for evolutionary algorithms—visualization of the path through search space and solution space using sammon mapping. Artif. Life **12**(2), 203–209 (2006)
8. Romero, G., Guervos, J.J.M., Valdivieso, P.A.C., Castellano, F.J.G., Arenas, M.G.: Genetic algorithm visualization using self-organizing maps. In: Proceedings of the Parallel Problem Solving from Nature, PPSN 2002, pp. 442–451 (2002)
9. Lotif, M.: Visualizing the population of meta-heuristics during the optimization process using self-organizing maps. In: Proceedings of the IEEE Congress on Evolutionary Computation, CEC 2014, pp. 313–319 (2014)
10. Volke, S., Zeckzer, D., Scheuermann, G., Middendorf, M.: A visual method for analysis and comparison of search landscapes. In: Proceedings of the Genetic and Evolutionary Computation Conference, GECCO 2015, pp. 497–504. Madrid, Spain, 11–15 July 2015
11. Collier, R., Wineberg, M.: Approaches to multidimensional scaling for adaptive landscape visualization. In: Pelikan, M., Branke, J. (eds.) Proceedings of the Genetic and Evolutionary Computation Conference, GECCO 2010, pp. 649–656. ACM (2010)
12. Masuda, H., Nojima, Y., Ishibuchi, H.: Visual examination of the behavior of emo algorithms for many-objective optimization with many decision variables. In: Proceedings of the IEEE Congress on Evolutionary Computation, CEC 2014, pp. 2633–2640 (2014)
13. Jornod, G., Mario, E.D., Navarro, I., Martinoli, A.: Swarmviz: An open-source visualization tool for particle swarm optimization. In: Proceedings of the IEEE Congress on Evolutionary Computation, CEC 2015, pp. 179–186. Sendai, Japan, 25–28 May 2015
14. Grond, F., Hermann, T., Kramer, O.: Interactive sonification monitoring in evolutionary optimization. In: 17th Annual Conference on Audio Display, Budapest (2011)

15. Zhang, Y., Dai, G., Peng, L., Wang, M.: Hmoeda_lle: A hybrid multi-objective estimation of distribution algorithm combining locally linear embedding. In: Proceedings of the IEEE Congress on Evolutionary Computation, CEC 2014, pp. 707–714 (2014)
16. Lee, J.A., Verleysen, M.: Quality assessment of dimensionality reduction: rank-based criteria. Neurocomputing **72**(7–9), 1431–1443 (2009)