NAIS 2025

June 18, 2025

Visualizing Multimodality in Combinatorial Search Landscapes

by

Xavier F. C. Sánchez-Díaz

Ole Jakob Mengshoel

Abstract: This work walks through different visualization techniques for combinatorial search landscapes, focusing on multimodality. We discuss different techniques from the landscape analysis literature, and how they can be combined to provide a more comprehensive view of the search landscape. We also include examples and discuss relevant work to show how others have used these techniques in practice, based on the geometric and aesthetic elements of the Grammar of Graphics. We conclude that there is no free lunch in visualization, and provide recommendations for future work as there are several paths to continue the work in this field.

Keywords: Multimodality, Visualization, Search Landscape, Combinatorial Optimization, Local Optima

Poster presented at the 2025 Symposium of the Norwegian Artificial Intelligence Society (NAIS 2025). June 18th & 19th, Tromsø, Norway.

NORWEGIAN UNIVERSITY OF SCIENCE AND TECHNOLOGY - 2025

1 Introduction

Context. Heuristic optimization algorithms are widely used to solve complex problems in engineering, industry, and science. Regardless of their representation, we aim to find solutions that are optimal inside the set of all possible solutions (or what we call the *search space*). The *search landscape* metaphor is a way to "visualize" the search space of an optimization problem [49], and how we exactly "see" this landscape is determined by the plotting techniques we use. Combinatorial problems in AI are abundant. From pseudo-Boolean problems like model finding [51], sentence summarization [41] and feature selection [20, 37], to other classical problems like the Traveling Salesman [2], routing [52], packing problems [36, 39], scheduling [10] and assignment [43]. Furthermore, quite often we are interested in multiple solutions—multimodality in these problems is common, but its existence is not always acknowledged. Recent conferences on evolutionary computation had just a few papers on multimodality (or problems with multiple optima), while the majority of the works focused on multi-objective optimization, benchmarking, and genetic programming.¹

Challenges. How do we get the most information from static visualizations of a combinatorial search landscape? Combinatorial problems are discrete, and the notions of order and continuity might not be well defined [35]. Thus, defining a neighborhood is crucial for calculating locality [38]. An interesting idea is to create composite visualizations. However, it is not trivial when an optimization problem contains multiple solutions of interest and the problem has high dimensionality [19, 22]. Moreover, oftentimes the number of optimal solutions is large, and we might be interested in finding as many as possible [11, 38].

Contributions. In this work, we get an overview of several visualization techniques for combinatorial search landscapes from an information design perspective. We focus on multimodality, and identify their strengths and limitations. Additionally, we propose a simple framework for combining these visualizations on the basis of their aesthetic attributes using the Grammar of Graphics [47, 48].

2 Background

2.1 Fitness Landscapes

This paper is aimed at researchers with a moderate understanding of optimization but who are not necessarily experts in search landscape analysis. We now introduce relevant concepts and notation.

We define a search (or fitness) landscape as a tuple $\mathfrak{L} = (\mathcal{X}, f, \mathcal{N})$, where \mathcal{X} is the search space, f is a generic fitness² function and \mathcal{N} is the neighborhood or any notion of accessibility [27]. By generic function we mean that f could be either a real-valued function, or a pseudo-Boolean function of any arity that we aim to optimize. For the sake of simplicity, we assume, without loss of generality, that we want to minimize f. The fitness function is *navigated* using a given optimization algorithm, and thus \mathfrak{L} is algorithm-dependent, which in turn determines how the neighborhood \mathcal{N}

¹See https://dl.acm.org/doi/proceedings/10.1145/3638529 and https://link.springer.com/book/10.1007/ 978-3-031-70085-9

²We use the term fitness as it is compared to the ability to survive of an individual in the context of biological evolution.

is constructed.

The search space \mathcal{X} is a set of solutions, which in our case is either a set of binary strings, permutations, or any other combinatorial object [35]. We use *b* to denote a solution in \mathcal{X} , as a solution is not necessarily a binary string.

We define a global optimum, denoted as b^* , as the optimal solution, i.e., $f(b^*) \leq f(b)$ for all $b \in \mathcal{X}$. A local optimum is a solution $b^+ \in \mathcal{X}$ such that $f(b^+) \leq f(b)$ for all $b \in \mathcal{N}(b^+)$, where $\mathcal{N}(b^+)$ is the neighborhood of b^+ . The set of all local optima in a given landscape \mathfrak{L} is represented with \mathcal{L} , and its cardinality (or size) as $|\mathcal{L}|$.

The neighborhood \mathcal{N} of a solution \boldsymbol{b} is the set of solutions that are *close* to \boldsymbol{b} . We define this *closeness* using a distance metric d. For example, in the case of binary strings, the Hamming distance d_H is a common choice for d, and the neighborhood of \boldsymbol{b} is the set of all solutions that differ from \boldsymbol{b} in one bit: $\mathcal{N}(\boldsymbol{b}) = \{\boldsymbol{b}' \in \mathcal{X} \mid d_H(\boldsymbol{b}, \boldsymbol{b}') = 1\}$. This neighborhood is usually employed in local search, genetic algorithms, and many other solvers that operate with bitstrings. In this work we do not focus on any specific neighborhood nor distance metric, but use these concepts to understand a search landscape.

Finally, throughout this work, we use several terms to refer to landscape features that resemble the topography of a physical landscape: *peaks*, *valleys*, *plateaus*, *funnels*, and *basins*. We do not define these terms formally, as their physical analogies should suffice to understand their meaning.

2.2 Visualization and the Grammar of Graphics

The Grammar of Graphics is a framework for building graphics, based on the idea of *layering* the different semantic elements of a plot, and then mapping these elements to data. Originally proposed by Wilkinson [48], the Grammar of Graphics has been implemented in software libraries and several programming languages, like R [47] and Julia [15].

From the Grammar of Graphics, we highlight two important concepts: *aesthetics* and *geometries*. Geometries are the "physical" elements of a plot: points, lines, bars, etc. These geometrical objects (which we refer to as *geoms*) are then mapped to the data using aesthetics elements. Aesthetics, on the other hand, are the visual properties of the geometries, like color, size, shape and position. See Table 1 for a list of common aesthetics on different mediums.

The notion of aesthetic elements and geometric objects is useful when designing visualizations for search landscapes, as we can use these elements to highlight different aspects of the landscape [7, 9]. Most importantly, the main advantage is the natural *association* between aesthetics elements and the characteristics of the landscape that can be visualized using them. For example, some visualizations employ color to represent the fitness of a solution, or use the size of a node to communicate the size of a basin of attraction. Using this framework, one can design and analyze visualizations that are both informative and aesthetically pleasing.

Table 1: Aesthetic attributes in the Grammar of Graphics. Adapted from Table 10.1 in [48]



Figure 1: The landscapes of some 2D test functions in the continuous domain [46]. The *z*-axis is used to plot the fitness, while the *x*- and *y*-axes are function parameters (shown here as x_1 and x_2).

2.3 Landscapes and Visualization

From Wright's conception of the fitness landscape [49] we have inherited the notion of *peaks* and *valleys* in the search space. Since then, the *visual features* of a search landscape have been discussed in detail [31, 34]. We now have the notion of *ridges*, *plateaus* and *basins*, and have identified more global structures like *funnels* and *canals* [16, 26, 27].

Visualizing search landscapes of real-valued functions of low dimensionality ($\mathbb{R}^1, \mathbb{R}^2$ or \mathbb{R}^3) is straightforward. For higher dimensionalities, one must find a mapping from \mathbb{R}^n to \mathbb{R}^2 or \mathbb{R}^3 , and then plot the function to get a sense of the search space.³ This mapping forms a continuous surface that can be analyzed with calculus tools, and our brains can wrap around it due to this continuity in the *space*. In this way, the features of the landscape (like peaks, valleys and funnels) are easily identified. See for example Figure 1 where different landscapes of 2D continuous test functions are plotted.

The situation is different when we deal with combinatorial landscapes since \mathcal{X} is a discrete set. What before was a smooth *dune* in the continuous case, becomes a set of *platforms* floating in mid-air, and the search points might not be able to be ordered—distance and continuity might

³Finding a mapping from \mathbb{R}^n to \mathbb{R}^2 or \mathbb{R}^3 is not trivial [19].

⁴ Visualizing Multimodality in Combinatorial Search Landscapes

not be well defined in the combinatorial case. We discuss methods for visualizing combinatorial search landscapes in Section 3, paying special attention to those highlighting multimodality and their use of geoms and aesthetics. For a more in-depth discussion of the mathematical implications of combinatorial landscapes, we refer the reader to the work of Reidys and Stadler [35].

2.4 Multimodality in Landscape Analysis

Multimodality, in the context of optimization, refers to the presence of multiple *modes* in \mathcal{X} , i.e., the presence of multiple optima in the landscape. When doing multimodal optimization, the goal is to *extract the full set of optima and optimizers the problem possesses* [32]. Realistically, in practice we look for multiple local optima for different reasons—to find different alternatives to a solution (to foster diversity [1] or for solution robustness [3]), to understand the structure of the problem and improve our search algorithms [22, 30, 43], as well as a stepping stone to avoiding premature convergence to find the global optimum [11, 38].

Several *everyday* activities of an AI practitioner involve multimodality. Consider the problem of feature selection in Machine Learning. It is a combinatorial optimization problem that is inherently multimodal (as several subsets of features can lead to the same performance of an ML model [13, 37]). Another multimodal combinatorial problem in the field of AI is hyper-parameter optimization: different combinations of hyper-parameter values can achieve the same classification results [21, 40]. Yet another example is the problem of neural network architecture search, where different architectures can achieve the same performance [14, 42]. These are only a handful of examples of multimodal optimization problems in AI, and they are all combinatorial in nature. Therefore, a push for more studies on multimodality in optimization would be beneficial for both the evolutionary computation and the landscape analysis community.

3 Related Work

3.1 Distance–Fitness Correlation

One of the key concepts in landscape analysis is the presence of a hypothetical *big valley*: a region in the search space where multiple local optima occur at around the same distance from a global optimum. The big valley is a common feature in multimodal landscapes, and it is often associated with the presence of a big central *funnel* [4]. A useful visualization technique to identify funnel-like structures in search landscapes is to plot the distance between local optima and the global optimum (or the closest, in case there are multiple global optima). See for example Figure 2, which highlights many local optima b^+ at the same distance from the closest global optimum b^* .

Another approach is to aggregate the local optima in bins, and then reporting the size of each bin to get a sense of the distribution of local optima in the landscape. Figure 3 shows this approach, illustrating the landscape of a feature selection problem under different values of regularization.

3.2 Local Optima Networks (LONs)

LONs are the de-facto standard for highlighting the multimodal structure of a search landscape [13, 27, 29, 37]. LONs can be described as graphs where nodes represent the local optima, and the



Figure 2: Analysis of 2511 local optima, on the feature selection problem of the Credit Approval [33] dataset using a decision tree classifier. The *y*-axis shows the quality of a solution (accuracy of classification) while the *x*-axis shows average Hamming distance between local optima (cf. left panel), and Hamming distance from each local optima to its closest global optimum (cf. right panel).

edges represent paths between them. An additional aesthetic element used in LONs is size, where bigger nodes represent bigger *basins of attraction*. Color can be used to further convey the size of the basin (as shown in Figure 7), but can also be used to communicate the fitness of each optimum [37].

Different versions of LONs exist, like the Monotonic LON (MLON) where escape edges that end up in a worse local optima are removed, and the Compressed MLONs where plateaus are compressed into a single node [26]. Additionally, LONs have been extended to multi-objective landscapes. The Pareto Local Optimal Solutions Network (PLOS-net), as these LONs are called, are plotted along a vertical axis to represent the different ranks of the optima [12]. As with LONs, PLOS-nets can also be compressed to remove plateaus.

Both LONs and PLOS-nets give a visual overview of the multimodality of a landscape as well as its neighborhood or connectedness. The main focus on these visualizations is to highlight the size of the basins of attraction, but the number and the distribution of the optima cannot be easily inferred from these plots.

3.3 Hinged Bitstring Maps (HBMs)

HBMs were introduced for visualizing pseudo-Boolean landscapes in PPSN 2024 [37]. HBMs plot the entire search space \mathcal{X} by slicing the bitstring *b* into two halves, and using the first half (converted to its decimal representation) as the *x*-axis. The *y*-axis is assigned to the decimal representation of the second half. Each solution is then plotted as a dot at the (x, y) coordinate, colored by its fitness value. Optima can additionally be highlighted by using a colored outline. We show an example of an HBM in Figure 8.



Hamming distance to closest global minimum

Figure 3: Hex-bin plot of distance correlation for the Heart Disease (Cleveland) dataset [8], using a decision tree classifier under four different levels of regularization. Each bin aggregates different number of local optima, b^+ . A darker shade means a higher concentration of b^+ .

HBMs are designed to look at \mathcal{X} in its entirety to highlight multimodality. However, depicting the neighborhood is difficult, as HBMs can become cluttered for large values of n since $|\mathcal{X}| = |\mathbb{B}^n| = 2^n$. Nevertheless, one of the advantages of looking at the whole search space is that local optima can be 'counted' and patterns about the distribution of optima can be analyzed.

3.4 Sequence Index Plots

As opposed to bitstrings, sequences do not have an inherent order. As such, visualizing sequences usually requires full enumeration. Combinations can be represented as a path in a fully connected graph, and sequences can be represented as a path in a directed graph in a similar manner, but this approach becomes infeasible for large cardinalities of \mathcal{X} , and so a sequence index plot might be more suitable [5, 7]. A sequence index plot uses the *x*-axis to represent the order of the sequence, and the *y*-axis to stack the different sampled sequences (or the *states*, in optimization). This type of visualization is useful when the number of sequences is small. However, it can also become cluttered when the domain for each position in the sequence is large enough.

Alternative visualizations for sequences exist, but these often include fully-built graphical user interfaces in specialized software, where additional information can be provided via *tooltips* or other interactive elements [7, 45]. Among static graphics, LONs tend to be more suitable to represent search landscapes of sequences. HBMs could also be used if the number of sequences to be visualized is small, provided that the sequences can be enumerated.

3.5 Search Trajectory Networks (STNs)

A search trajectory network (STN) is a special case of LON. It is a directed graph that represents the search process of multiple optimization algorithms [28]. As the name implies, the network is made up of different *trajectories*, which are paths in \mathcal{X} that represent the sequence of states *b* visited by each algorithm. Several algorithms can be represented in the same STN, and the edges can be weighted by the number of times a transition between two states has been made.

STNs can be used to compare different optimization algorithms in a single visualization, using



Figure 4: An STN comparing two algorithms: Biased Random-Key Genetic Algorithm and Ant Colony Optimization using the discrete example problem from Ochoa et al. [28]. The plot was generated using the STNs online tool.

different *geoms* and aesthetics to represent the different heuristics. Additionally, notable nodes are usually highlighted—like the best solution found by each algorithm, its start and end states, as well as "shared" states that were visited in different runs. As is the case with LONs, STNs can be compressed depending on the granularity used to represent the states, and additional metrics can be calculated due to their graph structure. An example of an STN is shown in Figure 4, generated by the STN Analytics online tool using the discrete example problem.⁴

Multimodality can be inferred from STNS by looking at the shared nodes. However, the notions of distance and neighborhoods between the optima are not as clear as with LONs, since these graphics were not specifically designed to highlight such features.

3.6 Violation Landscapes (VL)

An important feature of real-world optimization problems is the presence of constraints, and they occur in both continuous and combinatorial landscapes. When \mathcal{X} contains solutions that are not feasible, we say that the solution is in *violation*. In a constrained search space, optimization algorithms must navigate the landscape to find feasible solutions, and it can be challenging when the constraints are complex or when the feasible region is small. It is, therefore, important to visualize the violation landscape (VL), which is the subset of \mathcal{X} that is feasible.

One way to visualize the VL is to plot the fitness of the solutions in \mathcal{X} , but only showing the fea-

⁴See https://www.stn-analytics.com/.

⁸ Visualizing Multimodality in Combinatorial Search Landscapes



Figure 5: Fitness and violation landscapes of a 2D Constrained Ackley Function. The constraint is handled as a penalty for all solutions b with f(b) > 15, creating two regions in \mathcal{X} .

sible solutions. Another (and perhaps the most common) approach is to use color in the plot to differentiate between the feasible and infeasible solutions. This seems more intuitive when the \mathcal{X} is continuous [17, 18]. Figure 5 shows an example of a 2D continuous functions with a constraint, the Ackley function.

As shown in Figure 5, VL visualizations use color to represent the feasibility of a solution which has a location determined by its x and y coordinates in the plot. To have a better overview of the problem instance, both fitness and violation landscapes are usually presented next to one another [17, 18]. However, using color to show infeasibility is not the only way to plot a VL. In fact, the VL itself depends on the definition of *constraint violation*, and in turn, different visualizations can be generated depending on the definitions used [50].

Combinatorial landscapes with constraints are more difficult to visualize, since the same issue as with unconstrained landscapes persists, but now with the additional requirement of adding solution feasibility to the plot. However, the same principles can be applied: color can be used to represent the feasibility of a solution, and the x and y axes can be used to represent the solution itself (as in Figure 5).

4 Combining Visualization Techniques

Different visualization techniques are not mutually exclusive. In fact, combining them can provide a more comprehensive view of a combinatorial landscape, since different visualizations highlight different aspects of the search space. Several ways of merging visualizations exist [7, 25], but the most common are juxtaposition, superimposition, overloading and nesting [9]. In this work we focus on the first two as a basis for combining visualization techniques, and present this process in Figure 6.



Figure 6: A simple process for combining visualizations. Superimposition may require additional data transformations, while juxtaposition demands more space.

4.1 Juxtaposition

Placing different visualizations next to one another is the simplest way to have different views of the same landscape. However, it requires an efficient relational linking and sufficient space to have a comfortable layout [9]. As an example of juxtaposition, consider aligning two binary strings (one on top of another). This view allows for immediate recognition of bits that are different at a given position, and it can be useful to identify, for example, redundant features in feature selection [23, 37].

Most of the figures in this work (Figures 2, 3, 5, 7 and 8) are examples of juxtaposition. We now discuss two case studies.

4.1.1 Juxtaposition of LONs: Zooming in on \mathcal{L}

A first obvious case of study is the juxtaposition of two different views of the same object. In Figure 7, we present two LON-based views of the same landscape: a feature selection problem sampled with different levels of detail. On the left panel of Figure 7, a LON with all basin transitions is presented, while the right panel shows a LON with only escape edges of size 3 or less. The former presents an overview of the whole landscape (and the connectedness between all the 17 local optima), while the latter presents a detailed view of only those local optima that can be reached in three or fewer bit flips from the other optima. In this case, juxtaposition allows us to switch the view and *zoom-in* on the details of the important elements of the landscape.



Figure 7: LON-LON: Juxtaposition of two LONs, representing the feature selection problem on the E-coli dataset [24], using a decision tree classifier. In 7b, the escape edges are only kept if their Hamming distance $d_H \leq D$.

4.1.2 Juxtaposition of HBMs: Transformations on *f*

Since the HBM was designed to look at the whole search space, transformations of the fitness function can be easily visualized by placing two HBMs side by side as long as the transformations on *f* have the same domain (\mathcal{X}). In Figure 8 we show the landscape of a feature selection problem under two different degrees of regularization. Here we used a colored outline to highlight local and global optima, and can compare how the number (and distribution) of these optima changes when the landscape is transformed. In this way, juxtaposition helps us identify that the $\epsilon = 1/8$ regularization (cf. right panel in Figure 8) reduces the number of optima that originally existed when $\epsilon = 0$ (left panel in Figure 8).

4.2 Superimposition

A superimposed view can be obtained when a visualization is overlaid on top of another, resulting in a new view that combines elements of both. These views are usually employed to highlight *spatial* relations between the original visualizations since there is a one-to-one correspondence between their spatial linking [9]. For instance, showing several routes in a Vehicle Routing or a Traveling Salesman Problem on top of a map allows for an easy comparison between them.

Superimposition of different kinds of landscape visualizations can be done in a similar fashion if there are *free* or *unused* aesthetics on at least one of them. Table 2 shows a summary of the different aesthetics and geoms used by the three main visualization techniques studied in this work—LONs, HBMs, and STNs. The aesthetic attributes that are not in use in one visualization can be used to represent information from the other, as long as there exists a mapping between them. A transformation might be needed to *fit* one of the aesthetics on top of another, but it can lead to more informative designs, even when there is no isometric projection between their axes [19].



Figure 8: HBM-HBM: Juxtaposition of two HBMs for the Glass Identification dataset [6] with different values of regularization. The local and global optima are highlighted with blue and red outlines, respectively.

We now discuss a case study of superimposition, describing how to combine a LON and an HBM.

4.2.1 Superimposition of LON+HBM: Identification of \mathcal{L}

A LON+HBM is a good example of superimposition as it gives a glimpse into the multimodality of the landscape and the distribution of the optima. As an example, see Figure 9, where a LON is plotted on top of an HBM depicting the landscape of toy problem: $f(b) = \sin(2 \operatorname{Dec}(b))$, $\forall b \in \mathbb{B}^6$. The used aesthetics are the same as in the standalone plots, but we use the unused attributes of one to merge into the other: LONs do not have a natural mapping to the *x*- and *y*-axes, but using the HBM coordinate system, we can place the nodes in space using the same hinged-mapping. Thus, identification of the location of a local optimum is possible, as well as working out which optimizer b^+ corresponds to a specific basin of attraction.

4.3 No Free Lunch in Landscape Visualization

Even when there are *free* attributes and a compromise can be made to merge two visualizations, some combinations can become difficult to read. For example, using the unused aesthetic attribute of STNs, we can plot the sequence of visited solutions on top of an HBM. However, the mapping between the sequence of visited states and the HBM would be difficult to establish, as the x- and y-axes of the HBM are not directly related to the sequence of the search. A possible workaround is to use text labels inside each node to represent the sequence of a specific algorithm, and use color to show different algorithms. The fitness would need to be taken out of the plot, with the opportunity to compensate by plotting another HBM next to it. It should not take long to realize that this visualization would quickly become difficult to read, as following different number sequences using different colors for the foreground text and the background shape requires a lot of focus. Nevertheless, it is a good example to understand the limitations of the different visualization techniques.

On the other hand, combining an HBM and a VL is trivial: using color to plot the feasibility of a solution instead of the fitness. This works well because the *position* aesthetic in both visualization

Table 2: Different aesthetic and geoms used by the three main visualization techniques studied. Attributes marked with N/A, also called unused aesthetic elements, are not used in the visualization and are thus free to be used for other purposes, as is the case with the LON+HBM (superimposition).

Plot type	Geoms		Aesthetics			
	Primary	Secondary	Color	Size	Position	Visibility
LON	Circle	Lines	Basin of attraction	Basin of attraction	N/A	$\mathcal{L} \subset \mathcal{X}$
НВМ	Circle	Rings	Fitness	N/A	Ь	X
STN	Circle	Arrows	Algorithm	Frequency	N/A	Explored space
	Triangle					
	Square					
			Combined visua	lizations		
LON-LON	Circle	Lines	Basin of attraction	Basin of attraction	N/A	$\mathcal{L} \subset \mathcal{X}$
HBM-HBM	Circle	Rings	Fitness	N/A	b	$f \circ \mathcal{X}, f' \circ \mathcal{X}$
LON+HBM	Circle	Lines	Basin of attraction	Basin of attraction	b	\mathcal{X}

techniques is used to represent the same feature of the landscape, which is the location of the solution itself. Then, to convey more information, the feasibility HBM can be *juxtaposed* next to an HBM with fitness, and the distribution of both feasible solutions and the optima over the landscape can be observed.

5 Conclusions and Recommendations

In this work we discussed the importance of visualizing multimodality in combinatorial search landscapes. We have shown that different visualization techniques have their own strengths and limitations, and provided a simple way to combine them as their composition can generate more comprehensive views of the landscape. With the Grammar of Graphics, we have shown that the different aesthetics and geoms can be employed to represent different features of the landscape, and that the unused aesthetic attributes of one plot can extend the expressive power of another.

For future work, the use of another aesthetics might be worth visiting. In particular, the time aesthetic—animated visualizations can provide a detailed view of the landscape in dynamic or noisy scenarios. Considering other landscape visualization methods (e.g. attractor networks [44]) is another interesting path for future work.

Acknowledgments

We would like to acknowledge the financial support from Project no. 311284 of the Research Council of Norway, as well as access to computing resources from the Department of Computer Science of the Norwegian University of Science and Technology.



Figure 9: LON+HBM: a LON using the HBM coordinate system to visualize the landscape of $f(\mathbf{b}) = \sin(2 \operatorname{Dec}(\mathbf{b}))$, $\forall \mathbf{b} \in \mathbb{B}^6$. The color is used to represent the size of the basin of attraction. Figure reproduced with permission from Masson et al. [19].

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

- [1] D. Antipov, A. Neumann, and F. Neumann. Local Optima in Diversity Optimization: Non-trivial Offspring Population is Essential. en. In *Parallel Problem Solving from Nature – PPSN XVIII*, pages 181–196. Springer, Cham, 2024. ISBN: 978-3-031-70071-2. DOI: 10.1007/978-3-031-70071-2_12. URL: https://link.springer.com/chapter/10.1007/978-3-031-70071-2_12 (visited on 09/17/2024). ISSN: 1611-3349.
- [2] K. D. Boese, A. B. Kahng, and S. Muddu. A new adaptive multi-start technique for combinatorial global optimizations. *Operations Research Letters*, 16(2):101–113, Sept. 1994. ISSN: 0167-6377. DOI: 10.1016/0167-6377(94)90065-5. (Visited on 11/25/2024).
- [3] D.-C. Dang, T. Jansen, and P. K. Lehre. Populations Can Be Essential in Tracking Dynamic Optima. en. Algorithmica, 78(2):660–680, June 2017. ISSN: 1432-0541. DOI: 10.1007/ s00453-016-0187-y. URL: https://doi.org/10.1007/s00453-016-0187-y (visited on 07/13/2022).
- [4] J. P. K. Doye, M. A. Miller, and D. J. Wales. The double-funnel energy landscape of the 38-atom Lennard-Jones cluster. *The Journal of Chemical Physics*, 110(14):6896–6906, Apr. 1999. ISSN: 0021-9606. DOI: 10.1063/1.478595. (Visited on 11/28/2024).

- [5] A. E. Fasang and T. F. Liao. Visualizing Sequences in the Social Sciences: Relative Frequency Sequence Plots. *Sociological Methods & Research*, 43(4):643–676, 2014. ISSN: 1552-8294. DOI: 10.1177/0049124113506563. (Visited on 11/23/2024).
- [6] B. German. Glass Identification. UCI Machine Learning Repository, 1987. DOI: https://doi.org/10.24432/C5WW2P.
- M. Gleicher. Considerations for Visualizing Comparison. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):413–423, Jan. 2018. ISSN: 1941-0506. DOI: 10.1109/TVCG. 2017.2744199. (Visited on 04/07/2025).
- [8] A. Janosi, W. Steinbrunn, M. Pfisterer, and R. Detrano. Heart Disease. UCI Machine Learning Repository, 1989. DOI: https://doi.org/10.24432/C52P4X.
- W. Javed and N. Elmqvist. Exploring the design space of composite visualization. In 2012 IEEE Pacific Visualization Symposium, pages 1–8, Feb. 2012. DOI: 10.1109/PacificVis. 2012.6183556. (Visited on 04/07/2025).
- [10] E. Lara-Cárdenas, X. Sánchez-Díaz, I. Amaya, and J. C. Ortiz-Bayliss. Improving Hyperheuristic Performance for Job Shop Scheduling Problems Using Neural Networks. In L. Martínez-Villaseñor, I. Batyrshin, and A. Marín-Hernández, editors, *Advances in Soft Computing*, pages 150–161, Cham. Springer International Publishing, 2019. ISBN: 978-3-030-33749-0. DOI: 10.1007/978-3-030-33749-0_13.
- X. Li, M. G. Epitropakis, K. Deb, and A. Engelbrecht. Seeking Multiple Solutions: An Updated Survey on Niching Methods and Their Applications. *IEEE Transactions on Evolutionary Computation*, 21(4):518–538, Aug. 2017. ISSN: 1941-0026. DOI: 10.1109/TEVC.2016.2638437. (Visited on 09/25/2023).
- [12] A. Liefooghe, B. Derbel, S. Verel, M. López-Ibáñez, H. Aguirre, and K. Tanaka. On Pareto Local Optimal Solutions Networks. In A. Auger, C. M. Fonseca, N. Lourenço, P. Machado, L. Paquete, and D. Whitley, editors, *Parallel Problem Solving from Nature PPSN XV*, pages 232–244, Cham. Springer International Publishing, 2018. ISBN: 978-3-319-99259-4. DOI: 10.1007/978-3-319-99259-4_19.
- [13] A. Liefooghe, R. Tanabe, and S. Verel. Contrasting the Landscapes of Feature Selection Under Different Machine Learning Models. In M. Affenzeller, S. M. Winkler, A. V. Kononova, H. Trautmann, T. Tušar, P. Machado, and T. Bäck, editors, *Parallel Problem Solving from Nature – PPSN XVIII*, pages 360–376, Cham. Springer Nature Switzerland, 2024. ISBN: 978-3-031-70055-2. DOI: 10.1007/978-3-031-70055-2_22.
- Z. Lu, I. Whalen, V. Boddeti, Y. Dhebar, K. Deb, E. Goodman, and W. Banzhaf. NSGA-Net: neural architecture search using multi-objective genetic algorithm. In *Proceedings of the Genetic and Evolutionary Computation Conference*, GECCO '19, pages 419–427, New York, NY, USA. Association for Computing Machinery, July 2019. ISBN: 978-1-4503-6111-8. DOI: 10.1145/3321707.3321729. URL: https://doi.org/10.1145/3321707.3321729 (visited on 09/01/2023).
- [15] MakieOrg/AlgebraOfGraphics.jl. MakieOrg, Nov. 2024. (Visited on 11/29/2024).
- [16] K. Malan and G. Ochoa. Recent advances in landscape analysis for optimisation and learning. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, GECCO '21, pages 899–917, New York, NY, USA. Association for Computing Machinery, July 2021. ISBN: 978-1-4503-8351-6. DOI: 10.1145/3449726.3461396. (Visited on 01/27/2024).
- [17] K. M. Malan, J. F. Oberholzer, and A. P. Engelbrecht. Characterising constrained continuous optimisation problems. In 2015 IEEE Congress on Evolutionary Computation (CEC), pages 1351–1358, May 2015. DOI: 10.1109/CEC.2015.7257045. (Visited on 11/29/2024).

- [18] R. Mallipeddi and P. Suganthan. Problem Definitions and Evaluation Criteria for the CEC 2010 Competition on Constrained Real- Parameter Optimization. In 2010. (Visited on 12/01/2024).
- [19] C. Masson, X. F. C. Sánchez-Díaz, and O. J. Mengshoel. Visualizing Pseudo-Boolean Functions: Feature Selection and Regularization for Machine Learning. In M. S. Krejca and M. Wagner, editors, *Evolutionary Computation in Combinatorial Optimization*, pages 150–166, Cham. Springer Nature Switzerland, 2025. ISBN: 978-3-031-86849-8. DOI: 10.1007/978-3-031-86849-8_10.
- [20] O. J. Mengshoel, E. Flogard, J. Riege, and T. Yu. Stochastic local search heuristics for efficient feature selection: an experimental study. In *Proc. NIKT*, pages 58–71, 2021.
- [21] O. J. Mengshoel, E. L. Flogard, T. Yu, and J. Riege. Understanding the cost of fitness evaluation for subset selection: Markov chain analysis of stochastic local search. In *Proceedings of the Genetic and Evolutionary Computation Conference*, GECCO '22, pages 251–259, New York, NY, USA. Association for Computing Machinery, July 2022. ISBN: 978-1-4503-9237-2. DOI: 10.1145/3512290.3528689. URL: https://doi.org/10.1145/3512290.3528689 (visited on 07/14/2022).
- [22] O. Mersmann, B. Bischl, H. Trautmann, M. Preuss, C. Weihs, and G. Rudolph. Exploratory landscape analysis. In *Proceedings of the 13th Annual Conference on Genetic and Evolutionary Computation*, GECCO '11, pages 829–836, New York, NY, USA. Association for Computing Machinery, July 2011. ISBN: 978-1-4503-0557-0. DOI: 10.1145/2001576.2001690. (Visited on 01/22/2024).
- W. Mostert, K. M. Malan, and A. P. Engelbrecht. A Feature Selection Algorithm Performance Metric for Comparative Analysis. en. *Algorithms*, 14(3):100, Mar. 2021. ISSN: 1999-4893.
 DOI: 10.3390/a14030100. URL: https://www.mdpi.com/1999-4893/14/3/100 (visited on 01/28/2024). Number: 3 Publisher: Multidisciplinary Digital Publishing Institute.
- [24] K. Nakai. Ecoli. UCI Machine Learning Repository, 1996. DOI: https://doi.org/10.24432/C5388M.
- [25] C. Nobre, M. Meyer, M. Streit, and A. Lex. The State of the Art in Visualizing Multivariate Networks. *Computer Graphics Forum*, 38(3):807–832, 2019. ISSN: 1467-8659. DOI: 10.1111/ cgf.13728. (Visited on 04/07/2025).
- [26] G. Ochoa and K. Malan. Recent advances in fitness landscape analysis. In *Proceed-ings of the Genetic and Evolutionary Computation Conference Companion*, GECCO '19, pages 1077–1094, New York, NY, USA. Association for Computing Machinery, July 2019. ISBN: 978-1-4503-6748-6. DOI: 10.1145/3319619.3323383. (Visited on 01/29/2024).
- [27] G. Ochoa and K. Malan. Landscape Analysis of Optimisation Problems and Algorithms. In Proceedings of the Genetic and Evolutionary Computation Conference Companion, GECCO '24 Companion, pages 984–1004, New York, NY, USA. Association for Computing Machinery, Aug. 2024. ISBN: 979-8-4007-0495-6. DOI: 10.1145/3638530.3648421. (Visited on 11/24/2024).
- [28] G. Ochoa, K. M. Malan, and C. Blum. Search trajectory networks: A tool for analysing and visualising the behaviour of metaheuristics. *Applied Soft Computing*, 109:107492, Sept. 2021. ISSN: 1568-4946. DOI: 10.1016/j.asoc.2021.107492. (Visited on 11/15/2024).
- [29] G. Ochoa, M. Tomassini, S. Vérel, and C. Darabos. A study of NK landscapes' basins and local optima networks. In *Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation*, GECCO '08, pages 555–562, New York, NY, USA. Association for Computing Machinery, July 2008. ISBN: 978-1-60558-130-9. DOI: 10.1145/1389095.1389204. (Visited on 11/28/2024).

- [30] G. Ochoa, S. Verel, F. Daolio, and M. Tomassini. Local Optima Networks: A New Model of Combinatorial Fitness Landscapes. In H. Richter and A. Engelbrecht, editors, *Recent Advances in the Theory and Application of Fitness Landscapes*, pages 233–262. Springer, Berlin, Heidelberg, 2014. ISBN: 978-3-642-41888-4. DOI: 10.1007/978-3-642-41888-4_9. (Visited on 11/29/2024).
- [31] E. Pitzer and M. Affenzeller. A Comprehensive Survey on Fitness Landscape Analysis. In J. Fodor, R. Klempous, and C. P. Suárez Araujo, editors, *Recent Advances in Intelligent Engineering Systems*, Studies in Computational Intelligence, pages 161–191. Springer, Berlin, Heidelberg, 2012. ISBN: 978-3-642-23229-9. DOI: 10.1007/978-3-642-23229-9_8. (Visited on 01/28/2024).
- [32] M. Preuss. Multimodal Optimization by Means of Evolutionary Algorithms. Natural Computing Series. Springer International Publishing, Cham, 2015. ISBN: 978-3-319-07406-1 978-3-319-07407-8. DOI: 10.1007/978-3-319-07407-8. (Visited on 07/02/2024).
- [33] J. R. Quinlan. Credit Approval. UCI Machine Learning Repository, 1987. DOI: https://doi.org/10.24432/C5FS30.
- [34] C. R. Reeves. Fitness Landscapes. In E. K. Burke and G. Kendall, editors, Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques, pages 681– 705. Springer US, Boston, MA, 2014. ISBN: 978-1-4614-6940-7. DOI: 10.1007/978-1-4614-6940-7_22. (Visited on 01/28/2024).
- [35] C. M. Reidys and P. F. Stadler. Combinatorial Landscapes. *SIAM Review*, 44(1):3–54, 2002. ISSN: 0036-1445. JSTOR: 4148412. (Visited on 04/08/2024).
- [36] X. Sánchez-Díaz, J. C. Ortiz-Bayliss, I. Amaya, J. M. Cruz-Duarte, S. E. Conant-Pablos, and H. Terashima-Marín. A Feature-Independent Hyper-Heuristic Approach for Solving the Knapsack Problem. *Applied Sciences*, 11(21):10209, Jan. 2021. ISSN: 2076-3417. DOI: 10. 3390/app112110209. (Visited on 11/18/2022).
- [37] X. F. C. Sánchez-Díaz, C. Masson, and O. J. Mengshoel. Regularized Feature Selection Landscapes: An Empirical Study of Multimodality. In M. Affenzeller, S. M. Winkler, A. V. Kononova, H. Trautmann, T. Tušar, P. Machado, and T. Bäck, editors, *Parallel Problem Solving from Nature – PPSN XVIII*, pages 409–426, Cham. Springer Nature Switzerland, Sept. 2024. ISBN: 978-3-031-70055-2. DOI: 10.1007/978-3-031-70055-2_25.
- [38] X. F. C. Sánchez-Díaz and O. J. Mengshoel. Estimating the Number of Local Optima in Multimodal Pseudo-Boolean Functions: Validation via Landscapes of Triangles. In *Proceedings* of the Genetic and Evolutionary Computation Conference Companion, GECCO '24 Companion, pages 211–214, New York, NY, USA. Association for Computing Machinery, Aug. 2024. ISBN: 9798400704956. DOI: 10.1145/3638530.3654156. URL: https://dl.acm.org/doi/ 10.1145/3638530.3654156 (visited on 08/07/2024).
- [39] X. F. C. Sánchez-Díaz, J. C. Ortiz-Bayliss, I. Amaya, J. M. Cruz-Duarte, S. E. Conant-Pablos, and H. Terashima-Marín. A Preliminary Study on Feature-independent Hyper-heuristics for the 0/1 Knapsack Problem. In 2020 IEEE Congress on Evolutionary Computation (CEC), pages 1–8, July 2020. DOI: 10.1109/CEC48606.2020.9185671. (Visited on 11/30/2023).
- [40] L. Schneider, L. Schäpermeier, R. P. Prager, B. Bischl, H. Trautmann, and P. Kerschke. HPO × ELA: Investigating Hyperparameter Optimization Landscapes by Means of Exploratory Landscape Analysis. en. In G. Rudolph, A. V. Kononova, H. Aguirre, P. Kerschke, G. Ochoa, and T. Tušar, editors, *Parallel Problem Solving from Nature PPSN XVII*, Lecture Notes in Computer Science, pages 575–589, Cham. Springer International Publishing, 2022. ISBN: 978-3-031-14714-2. DOI: 10.1007/978-3-031-14714-2_40.

- [41] R. Schumann, L. Mou, Y. Lu, O. Vechtomova, and K. Markert. Discrete optimization for unsupervised sentence summarization with word-level extraction. In *Proc. ACL*, pages 5032– 5042, 2020.
- K. O. Stanley, J. Clune, J. Lehman, and R. Miikkulainen. Designing neural networks through neuroevolution. en. *Nature Machine Intelligence*, 1(1):24–35, Jan. 2019. ISSN: 2522-5839.
 DOI: 10.1038/s42256-018-0006-z. URL: https://www.nature.com/articles/s42256-018-0006-z (visited on 09/27/2022). Number: 1 Publisher: Nature Publishing Group.
- S. L. Thomson, G. Ochoa, D. van den Berg, T. Liang, and T. Weise. Entropy, Search Trajectories, and Explainability for Frequency Fitness Assignment. In M. Affenzeller, S. M. Winkler, A. V. Kononova, H. Trautmann, T. Tušar, P. Machado, and T. Bäck, editors, *Parallel Problem Solving from Nature PPSN XVIII*, pages 377–392, Cham. Springer Nature Switzerland, 2024. ISBN: 978-3-031-70055-2. DOI: 10.1007/978-3-031-70055-2_23.
- [44] S. L. Thomson, Q. Renau, D. Vermetten, E. Hart, N. v. Stein, and A. V. Kononova. Stalling in Space: Attractor Analysis for any Algorithm, Dec. 2024. DOI: 10.48550/arXiv.2412.15848. URL: http://arxiv.org/abs/2412.15848 (visited on 04/03/2025). arXiv:2412.15848 [cs].
- [45] S. van der Linden, E. de Fouw, S. van den Elzen, and A. Vilanova. A survey of visualization techniques for comparing event sequences. *Computers & Graphics*, 115:522–542, Oct. 2023. ISSN: 0097-8493. DOI: 10.1016/j.cag.2023.05.016. (Visited on 11/23/2024).
- [46] C. Vanaret, J.-B. Gotteland, N. Durand, and J.-M. Alliot. Certified Global Minima for a Benchmark of Difficult Optimization Problems, Mar. 2020. DOI: 10.48550/arXiv.2003.09867. arXiv: 2003.09867 [cs, math]. (Visited on 05/25/2023).
- [47] H. Wickham. *Practical Tools for Exploring Data and Models*. PhD thesis, Iowa State University, 2008.
- [48] L. Wilkinson. The Grammar of Graphics. Statistics and Computing. Springer-Verlag, New York, 2nd edition, 2005. ISBN: 978-0-387-24544-7. DOI: 10.1007/0-387-28695-0. (Visited on 11/13/2024).
- [49] S. Wright. The roles of mutation, inbreeding, crossbreeding and selection in evolution, Proceedings of the Sixth International Congress of Genetics. In *Proc Sixth Int Congr Genet*, volume 1, page 356, 1932. (Visited on 11/28/2024).
- [50] Y. Yasuda, K. Tamura, and K. Yasuda. Analyzing Violation Landscapes Using Different Definitions of Constraint Violation. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, GECCO '24 Companion, pages 1815–1823, New York, NY, USA. Association for Computing Machinery, Aug. 2024. ISBN: 979-8-4007-0495-6. DOI: 10.1145/3638530.3664118. (Visited on 11/24/2024).
- [51] E. Yolcu and B. Poczos. Learning local search heuristics for boolean satisfiability. In *Proc. NeurIPS*, pages 7992–8003. 2019.
- [52] D. Zhang, S. Cai, F. Ye, Y.-W. Si, and T. T. Nguyen. A hybrid algorithm for a vehicle routing problem with realistic constraints. *Information Sciences*, 394–395:167–182, July 2017. ISSN: 0020-0255. DOI: 10.1016/j.ins.2017.02.028. (Visited on 11/09/2022).