Visualising the Landscape of Multi-Objective Problems using Local Optima Networks

Jonathan E. Fieldsend
University of Exeter
Department of Computer Science
J.E.Fieldsend@exeter.ac.uk

Khulood Alyahya
University of Exeter
Department of Computer Science
K.Alyahya@exeter.ac.uk

ABSTRACT

Local optima networks (LONs) represent the landscape of optimisation problems. In a LON, graph vertices represent local optima in the search domain, their radii the basin sizes, and directed edges between vertices the ability to transit from one basin to another (with the edge width denoting how easy this is). Recently, a network construction approach inspired by LONs has been proposed for multi-objective problems which uses an undirected graph, representing mutually non-dominating solutions and neighbouring links, but not basin sizes. In contrast, here we introduce two formulations for multi/many-objective problems which are analogous to the traditional LON, using dominance-based hill-climbing to characterise the search domain. Each vertex represents a set of locally optimal solutions, with basins and ease of transition between them shown. These LONs vary depending on whether a point-based (dominance neutral optima) or set-based (Pareto local optima) representation is used to define mode construction. We illustrate these alternative formulations on some illustrative problems. We discuss some of the underlying computational issues in constructing LONs in a multi-objective as opposed to uni-objective problem domain, along with the inherent issue of neutrality — as each a vertex in these graphs almost invariably represents a set in our proposed constructs.

CCS CONCEPTS

• **Mathematics of computing** → **Combinatorial optimization**; **Graph theory**; • **Applied computing** → **Multi-criterion optimization and decision-making**; • **Human-centered computing** → **Visualization**; **Visualization design and evaluation methods**;

KEYWORDS

multi-objective optimisation; fitness landscapes

ACM Reference Format:
Jonathan E. Fieldsend and Khulood Alyahya. 2019. Visualising the Landscape of Multi-Objective Problems using Local Optima Networks. In *Genetic and Evolutionary Computation Conference Companion (GECCO '19 Companion), July 13–17, 2019, Prague, Czech Republic.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3319619.3326838

1 INTRODUCTION

Local optima networks (LONs) have been developed and used to visualise the landscape of uni-objective problems [12, 17], using directed graphs with variable node size and colour to denote information about the search landscape. Recent work inspired by LONs has proposed a network representation for multi-objective problems [10]. These Pareto local optimal solutions networks (PLOS-nets) use undirected graphs. Each vertex representing a single local optima, with edges connecting neighbouring solutions. In contrast with PLOS-nets, we present here definitions of LONs for multi-objective problems which employ directed graphs, convey information regarding basin sizes and which can incorporate the standard edge definitions used in LONs. Due to the nature of multi-modality in multi-objective problems, the vertices represent sets of solutions rather than single solutions in our approach. As such, to the best of the authors' knowledge, this work presents the first full transference of the LON landscape visualisation methodology to the multi-objective domain.

We present two formulations: one relying on a point-based hill-climb and *dominance* to describe contiguous regions of neutrality under this measure to define vertices, the other, relying on a set-based search approach, which results in vertices composed of Pareto local optima.

The paper proceeds as follows: Section 2 briefly describes local optima networks, as developed in the uni-objective optimisation domain; Section 3 describes the multi/many-objective optimisation problem, along with the different types of optima that inhabit this landscape, and how they are identified; Section 4 describes the PLOS-net of [10] before introducing our two new LON-based approaches along with bi-objective examples; Section 5 provides illustrations of the proposed LONs constructs on some larger many-objective problems. The paper concludes with a discussion in Section 6.

2 LOCAL OPTIMA NETWORKS

A LON is a network-based model that compresses the information of the search space into a weighted oriented graph. The model is adapted from the inherent networks of energy landscapes in physical-chemistry and is used as a method to visualise and apply complex network analysis tools to combinatorial optimisation problems [12, 17]. The vertices of a given landscape graph are the local optima under its neighborhood operator. An edge between two vertices can be defined in different ways. It was originally defined in such a way to represent when two optima have adjacent

GECCO '19 Companion, July 13–17, 2019, Prague, Czech Republic

ACM ISBN 978-1-4503-6748-6/19/07…$15.00
https://doi.org/10.1145/3319619.3326838

Algorithm 1 Greedy (best) improvement hill-climbing

```
Require: $\mathbf{x}^{start}$                                  ▷ Initial sample
 1: $\mathbf{x}^{end} := \mathbf{x}^{start}$
 2: $converged$ := false
 3: while $converged$ = false do
 4:     $better\_found$ := false
 5:     for $\mathbf{x}' \in N(\mathbf{x}^{end})$ do
 6:         if $f(\mathbf{x}) < f(\mathbf{x}^{end})$ then             ▷ If neighbour is better
 7:             $\mathbf{x}^{end} = \mathbf{x}$                  ▷ Replace with better neighbour
 8:             $better\_found$ := true
 9:         end if
10:     end for
11:     if $better\_found$ = true then
12:         goto 4          ▷ Better neighbour found, so climb again
13:     else
14:         $converged$ := true              ▷ No better neighbour found
15:     end if
16: end while
17: return $\mathbf{x}^{end}$                 ▷ Return path start and end locations
```

Algorithm 2 First improvement hill-climbing

```
Require: $\mathbf{x}^{start}$                                  ▷ Initial sample
 1: $\mathbf{x}^{end} := \mathbf{x}^{start}$
 2: $converged$ := false
 3: while $converged$ = false do
 4:     $Z$ := random_order($N(\mathbf{x}^{end})$) ▷ Random order neighbours
 5:     for $\mathbf{z} \in Z$ do          ▷ For each randomly ordered neighbour
 6:         if $f(\mathbf{z}) < f(\mathbf{x}^{end})$ then             ▷ If neighbour is better
 7:             $\mathbf{x}^{end} = \mathbf{z}$                ▷ Replace with better neighbour
 8:             goto 4      ▷ Better neighbour found, so climb again
 9:         end if
10:     end for
11:     $converged$ := true                  ▷ No better neighbour found
12: end while
13: return $\mathbf{x}^{end}$                 ▷ Return path start and end locations
```

basins (i.e. the transition probability between the basins of the two optima). However, this definition was found to be computationally expensive and produces a densely connected graph. An alternative definition of an edge was therefore proposed to represent the probability of escaping from one local optimum to another after applying controlled mutation followed by hill-climbing [18].

Local optima networks have been applied to study the landscape of several combinatorial optimisation problems, e.g. [2, 6, 13]. It has also been applied to study the behaviour of different local search methods. For example, in [14] LONs are applied to *NK* problem examples with two different hill-climbing algorithms employed to generate paths, and hence define the subsequent LONs generated. These were (i) the greedy hill-climbing formulation used in earlier work on LONs, which always selects the best move out of all neighbours (detailed in Algorithm 1), and (ii) the next improvement hill-climbing formulation which selects the first evaluated of the neighbours (randomly ordered) that is better than the current design (detailed in Algorithm 2). LONs generated using these two different algorithms were designated b-LONs and f-LONs respectively. The edges were calculated using the basin transition definition. Calculating the f-LONs is obviously cheaper compared to b-LONs where each move requires the evaluation of the entire neighbourhood. The f-LONs were found to be more dense than b-LONs with lower weighted self-loops indicating that it is easier to escape an optimum using first-improvement hill-climber. The shortest paths in f-LONs were found to be on average slightly longer than the ones in b-LONs.

Most of the existing work on LONs were limited to relatively small problem sizes due to the requirement of complete enumeration of the search space to find all local optima and calculate the edges between them. However, recently sampling methods have been proposed to sample LONs efficiently [3] and in statistical sound way as not to lose accuracy of the network metrics [19]. Other methods have been proposed to sample LONs from the actual runs of local search methods [7, 11].

The LON construction for uni-objective landscapes exhibiting neutrality was investigated in [21] on neutral variants of *NK* landscapes. The study generated LONs using exact calculation and thus was limited to enumerable problem sizes. The main issue they found with neutrality is the definition of basins of attraction and the transition between them. LONs were defined such that each vertex represents a network of neutral neighbours, i.e. a plateau. An edge between two vertices was defined as the transition probability between their basins of attraction. In [19], they mention as future work extending their sampling method to sample LONs of fitness landscapes with significant neutrality.

3 MULTI-OBJECTIVE OPTIMISATION

In the general multi-objective optimisation case (two or three objectives) and the many-objective optimisation case (four or more objectives) we seek to find the Pareto set of solutions — or, more realistically, an estimate of it. Without loss of generality, we seek to simultaneously minimise K objectives: $f_k(\mathbf{x})$, $k = 1, \ldots, K$, where each objective depends upon a vector $\mathbf{x} = (x_1, \ldots, x_N)$ of N design or decision variables. These variables may be continuous, discrete, or a combination of both. The variables may also be subject to equality and inequality constraints. Such constraints define X: the feasible design space.

When there is more than one objective to be minimised, solutions may exist for which performance on one objective cannot be improved without reducing performance on at least one other. Such solutions are said to be *Pareto optimal*. The set of all Pareto optimal solutions is said to form the *Pareto set*, whose image in the objective space is known as the *Pareto front*. Identifying such solutions relies on Pareto *dominance*. A decision vector $\mathbf{x}$ is said to dominate another $\mathbf{x}'$ iff

$$f_k(\mathbf{x}) \leq f_k(\mathbf{x}') \quad \text{for all} \quad k = 1, \ldots, K \quad \text{and} \quad \mathbf{f}(\mathbf{x}) \neq \mathbf{f}(\mathbf{x}'). \quad (1)$$

This is often simply denoted as $\mathbf{x} \prec \mathbf{x}'$ rather than $\mathbf{f}(\mathbf{x}) \prec \mathbf{f}(\mathbf{x}')$. Formally, the Pareto set (the global Pareto optima, or GPO) can be extracted from X using the nondom function:

$$\text{nondom}(X) = \{\mathbf{x} \in X | \nexists \mathbf{x}' \in X, \mathbf{x}' \prec \mathbf{x}\}, \quad (2)$$

Algorithm 3 Pareto Local Search (PLS) of [15].

```
Require: $\mathbf{x}^{start}$                                   ▷ Initial sample
$F := \{\mathbf{x}^{start}\}$                                   ▷ Initial set to process
$X_{PLO} := \emptyset$                                          ▷ Set holding estimated PLO
while $F \neq \emptyset$ do                                     ▷ While still locations to process
    $\mathbf{x} := random\_draw(F)$
    $W := \emptyset$
    for all $\mathbf{x}' \in \mathcal{N}(\mathbf{x})$ do
        if $\nexists \mathbf{x}'' \in W \cup F \cup X_{PLO} \,|\, \mathbf{x}'' \preceq \mathbf{x}'$ then
            $W := W \cup \mathbf{x}'$                           ▷ Not dominated by current estimate
        end if
    end for
    $X_{PLO} := \{\mathbf{x}' \,|\, \mathbf{x}' \in X_{PLO} : \nexists \mathbf{x}'' \in W \,|\, \mathbf{x}'' \prec \mathbf{x}'\}$
    $F := F \setminus \{\mathbf{x}\} \cup W$
    $F := \{\mathbf{x}' \,|\, \mathbf{x}' \in F : \nexists \mathbf{x}'' \in W \,|\, \mathbf{x}'' \prec \mathbf{x}'\}$
end while
return $X_{PLO}$                                                ▷ Return path start location and end set
```

however in practice $|X|$ is often too large to be exhaustively searched, and optimisers seek to return an approximation to X_{GPO} (the set returned by nondom(X)), in terms of quality under $\mathbf{f}$.

What does a fitness landscape mean in the context of multi- or many-objective optimisation? Multi-objective optimisation is most typically a set-based procedure. Even single parent single child (1+1) methods like the popular PAES algorithm [9] rely on maintaining an archive of the approximated Pareto set, to reference for movement decisions. When X is in the plane we may use dominance landscapes [4] to visualise the local search landscape, however in higher design dimensions it is natural to explore whether there is an appropriate analogue of the uni-objective LON visualisation. To do this we need to consider what the multi-objective equivalent to the point-based hill-climbing used in LONs is.

3.1 Pareto local optima

We first consider the *set-based* local hill-climb described in Pareto Local Search (PLS) [15]. This is detailed in Algorithm 3. Although it starts with a single point in design space, it maintains a dynamic set to identify the Pareto local set of solutions (the Pareto local optima) identified from hill-climbing from an initial point. This set is comprised entirely of mutually non-dominating designs. More formally, the Pareto local optimum (PLO) set $X_{PLO} \in X$ with respect to the neighbour function $\mathcal{N}$ is composed such that $\forall \mathbf{x} \in X_{PLO} \nexists \mathbf{x}' \in \mathcal{N}(X_{PLO})$ where $\mathbf{x}' \prec \mathbf{x}$ (see e.g. [15, 20] for further details).

Consider the simple problem illustrated in Figure 1, using the von Neumann neighbourhood function in a box constrained plane. Here there are two criteria evaluated on designs from the domain X, which has 25 members in total. There are two distinct spatial groupings of X_{PLO} individuals, which are shown with red vertices. The Pareto set (the global Pareto optima) are highlighted with blue circles, and are a subset of the union of all PLO.

Figure 2 illustrates the process from an initial sample to convergence using PLS on the problem from Figure 1. Outwardly using PLS to characterise the landscape is an attractive approach, however, from a practical point of view (and as identified in [15]) it can be prohibitively expensive in large domains. This is because, where a landscape is approximated with a fixed budget rather than completely enumerated, one may spend the entire budget on a single hill-climb (and resultant local Pareto set enumeration), rather than approximating the landscape over the broader domain. Effectively, each X_{PLO} behaves similarly to a plateau in the uni-objective fitness landscape. However, this landscape feature is considerably more expensive to identify and explore. In the uni-objective case, a solution equal in fitness to one plateau member is equal in fitness to all other members, but in the multi-objective case being

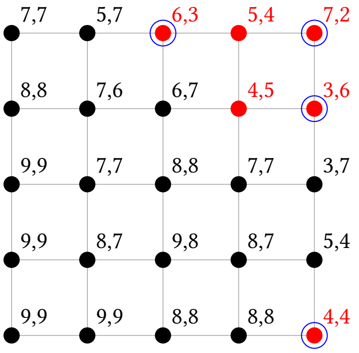
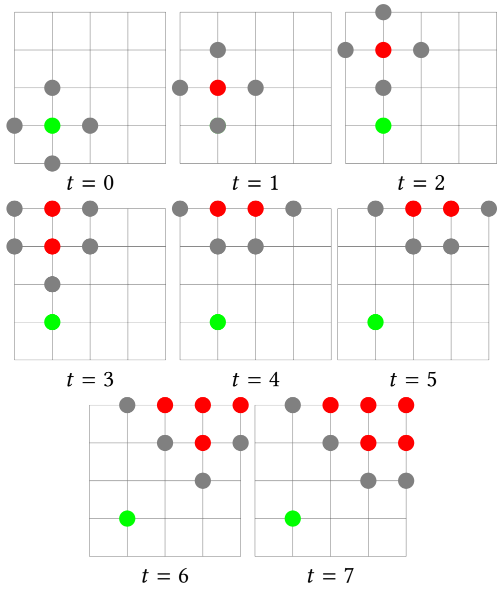


Figure 1: Landscape illustration. Each vertex represents a solution, with edges connecting neighbours under $\mathcal{N}$. Cost under two criteria denoted to the top right of a vertex. X_{PLO} members are coloured red. The global Pareto optima (GPO) are circled in blue. Note in this example $X_{GPO} \neq \bigcup_i X^i_{PLO}$.


$t = 0$ $t = 1$ $t = 2$

$t = 3$ $t = 4$ $t = 5$

$t = 6$ $t = 7$

Figure 2: Landscape hill-climb illustration using PLS on the problem shown in Figure 1, with repetitions t of the while loop. The start point is denoted with a green node, the estimate of X_{PLO} with red nodes, and $\mathcal{N}(X_{PLO}) \setminus X_{PLO}$ with grey nodes.

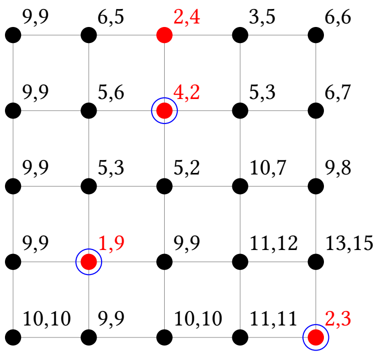


Figure 3: Landscape illustration. Each vertex represents a solution, with edges connecting neighbours under $\mathcal{N}$. Cost under two criteria denoted to the top right of a vertex. There are three X_{PLO}, whose members are coloured red. However, depending on the start location two of the X^i_{PLO} will be returned as a *single* set from a hill-climb. The global Pareto optima (GPO) are circled in blue.

mutually non-dominated with one local Pareto set member *does not* ensure mutual non-domination with all other members of an X^i_{PLO}. Although the use of advanced data structures means the computational cost of comparison does not need to grow linearly with the set size [5, 8], nevertheless, the computational cost of verifying membership still grows with $|X^i_{PLO}|$, and can become a significant computational burden.

3.1.1 Inconsistency of landscape under Pareto local search, and implications. Until now we have denoted the distinct PLO sets returned by an initial location in PLS as X^i_{PLO}. However, a property not generally discussed in the literature regarding the PLO returned from PLS is that in some cases both $X^i_{PLO} \cap X^j_{PLO} \neq \emptyset$ and $X^i_{PLO} \setminus X^j_{PLO} \neq \emptyset$ are true. The root cause of this that the PLO returned by PLS can be *disjoint*. That is, one or more elements of the returned set are not reachable by walking from other members of the returned set, via neighbours in the same set. Also, depending on the initial starting point for the PLS, two sets of PLOs returned by two different start points can be intersecting *but not equal*.

The possibility of returning intersecting but not equal sets effectively means the landscape is not consistent when viewed from different starting points in the design space, and interpreting the resultant visualisation can become problematic. We cannot easily make the cognitive transition from modes in uni-objective space (single solutions) to modes in multi-objective space (sets of solutions), as the PLS from a location may return a single mode, or multiple disjoint modes, which may (partially) overlap with single or multiple modes returned from the PLS from a *different* starting point. A simple illustration of such a situation is provided in Figures 3 and 4, which show a problem and the basins associated with each (disjoint) PLO set induced by it using the von Neumann neighbourhood.

A solution to this may initially appear to be splitting (via e.g. path finding algorithms) each PLO set discovered by a PLS run into its maximally sized disjoint subsets — that is those subsets where all elements can be reached by neighbourhood walks, whose union covers the PLO of the domain, and who do not intersect. However, this will not in general be sufficient. Figure 5 shows the basins of attraction for the problem illustrated in Figure 1, and the corresponding X^i_{PLO} returned using PLS. The PLO which are in the same returned set from one starting point can have elements rejected due to domination from PLS members from another starting point, due to discovery of a disjoint PLO members during the search. In this particular example the same pair of solutions appear in three different X^i_{PLO}. At the same time, only a subset of the X_{PLO} members appearing in the top right panel, also appear in the two bottom panels.

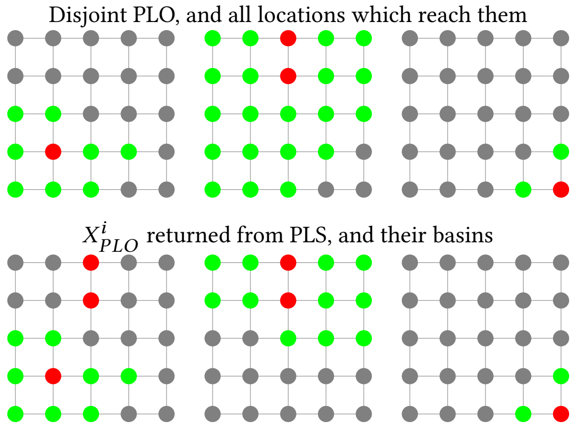

Disjoint PLO, and all locations which reach them
X^i_{PLO} returned from PLS, and their basins

Figure 4: *Top:* Basins for each disjoint PLO set, illustration from Figure 3 used. PLO sets highlighted in red, basin members in green. Note that from some starting locations disconnected X_{PLO} appear in the same set returned from PLS, but from other starting locations only one disjoint subset is returned. *Bottom:* The X^i_{PLO} returned from PLS, with starting locations indicated in green showing respective basins.

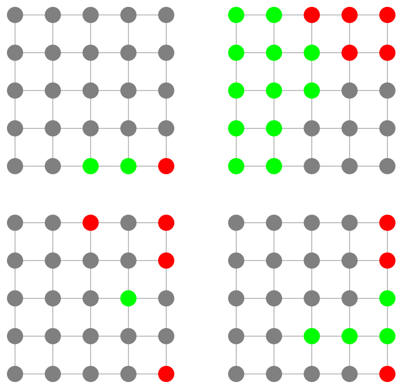


Figure 5: Exact basins for each X^i_{PLO} set returned by PLS, using problem illustrated in Figure 1. PLO sets highlighted in red, respective basin in green. Note that from some starting locations the PLO returned may include a subset of that from another starting location, due to members being dominated by a disjoint elements also discovered during search.

Algorithm 4 Pareto Hill-Climbing (PHC) approach of [20].

```
Require: $\mathbf{x}^{start}$                              ▷ Initial sample
$\mathbf{x}^{end} := \mathbf{x}^{start}$
$steps := 0$
$converged :=$ false
while $converged =$ false do
    $Z :=$ dominators($\mathcal{N}(\mathbf{x}^{end}), \mathbf{x}$)          ▷ Get dominating
    if $Z = \emptyset$ then
        $converged :=$ true      ▷ No dominating neighbour found
    else
        $\mathbf{x}^{end} =$ random_select($Z$)        ▷ Move to dominator
        $steps := steps + 1$
    end if
end while
return ($\mathbf{x}^{end}, steps$)                   ▷ Return path details
```

As outlined above, generating meaningful LONs using PLO is non-trivial. The set returned by PLS can intersect with that returned from a PLS started from a different solution, and therefore a solution may appear in multiple modes (in standard uni-objective LONs, a solution may contribute weight to multiple basins but a modal solution defines only one mode).

3.2 Dominance neutral optima

An alternative approach to define the landscape is to employ a *point-based* local hill-climb to define a dominance-neutral neighbourhood [4]. Previous work to take such an approach to multi-objective landscape analysis is the Pareto Hill-Climbing (PHC) approach of [20], which randomly moves a design in the hill-climb to a dominating neighbouring solution until no dominating neighbour was found. This approach is detailed in Algorithm 4.

It was asserted in [20] that the solution returned by PHC would be a member of an X_{PLO}. However, this is not guaranteed, and indeed we observe that regularly this is not the case when $\mathbf{x}^{start} \notin \bigcup_i X^i_{PLO}$. Figure 6 illustrates a simple counter example to the assertion that PHC results in a member of X_{PLO}, using the same data as illustrated in Figure 1. Nevertheless, such a point-based rather than set-based hill-climb is attractive from a practical point of view, as it does not require exhaustive enumeration of a X_{PLO} for a hill-climb to complete, and is therefore practical in a landscape approximation context (via sampling).

Using PHC results in dominance-neutral neighbourhoods (X_{DNO}) comprised of solutions where each member is mutually non-dominated with immediate neighbouring members of the set, which we call dominance neutral optima (DNO). However, it does not ensure the set as a whole is comprised of mutually non-dominating solutions. In theory, this may lead to the pathological situation where $X_{DNO} = X$ (or a substantial portion of it). This may arise when there are *many* objectives, where the dominance measure is known to be less discriminating [1]. To be clear, in terms of set relationships: $\bigcup_i X^i_{GPO} \subseteq \bigcup_j X^j_{PLO} \subseteq \bigcup_m X^m_{DNO} \subseteq X$.

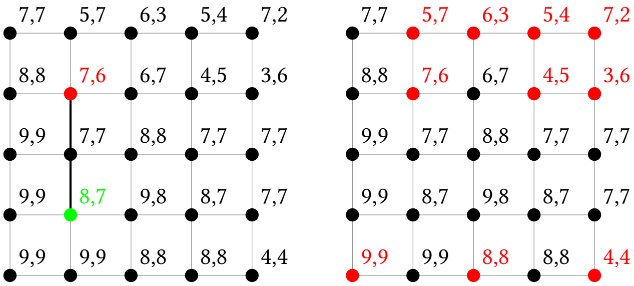


Figure 6: Example where PHC does not return a member of an X_{PLO}. Left panel shows $\mathbf{x}^{start}$ in green, and $\mathbf{x}^{end}$ in red with the path in black (note how none of the neighbours of $\mathbf{x}^{end}$ dominate it). The right panel shows all dominance-neutral optima under PHC coloured red. $\bigcup_j X^j_{DNO}$ is a superset of $\bigcup_i X^i_{PLO}$.

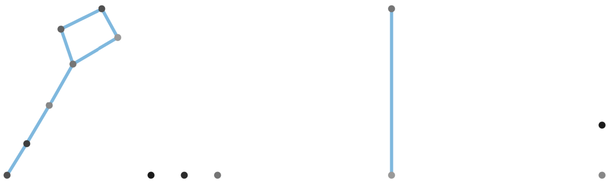


Figure 7: PLOS-nets corresponding to the problems illustrated in Figure 1 (*left*) and Figure 3 (*right*).

4 MULTI-OBJECTIVE OPTIMISATION LANDSCAPES

We now describe existing work on representing multi-objective optimisation landscapes as graph, before presenting two new approaches which embody many of the properties of LONs.

4.1 PLOS-nets

As mentioned in Section 1, PLOS-nets have recently been proposed as the first attempt to formulate a LON-like graph visualisation for multi-objective problems [10]. Although the term 'Pareto local optima' is used to describe the solutions defining the vertices in [10], they are actually defined in the work in terms of immediate neighbourhood non-dominance, and thus are not PLO (as defined in [15]). To avoid confusion, we will refer to the vertex solution properties in PLOS-nets as dominance neutral optima rather than Pareto local optima.

In PLOS-net construction, after the DNO are identified via enumeration of the domain, the graph vertices are used to represent each DNO solution. Edges between vertices are formed if one solution is a mutually non-dominating neighbour of another, resulting in an undirected graph. Figure 7 shows the PLOS-net graphs for the problems illustrated in Figure 1 and Figure 3. In keeping with [10], vertices are coloured according to their corresponding solution quality under the Kth objective.

Algorithm 5 PLON generation.

```
$\mathbf{X}_{PLO} = \emptyset$    ▷ Initial set of vertex sets
$\mathbf{B} = \emptyset$    ▷ Initial map to basin size
$\mathbf{C} = \emptyset$    ▷ Initial map to vertex colours
$E^{subset} = \emptyset$    ▷ Initial map of vertices to subsets
for $\forall \mathbf{x} \in X$ do
    $P := \texttt{PLS}(\mathbf{x})$    ▷ Get resulting PLO set, see Algorithm 3
    if $P \neq X^i_{PLO} \; \forall X^i_{PLO} \in \mathbf{X}_{PLO}$ then    ▷ If new $X^i_{PLO}$
        $\mathbf{X}_{PLO} := \texttt{append}(\mathbf{X}_{PLO}, P)$    ▷ Add to list of $X^i_{PLO}$
        $B_P := 1$    ▷ Initialise basin count
    else
        $B_P := B_P + 1$    ▷ Increment basin count through map
    end if
end for
for $\forall X^i_{PLO} \in \mathbf{X}_{PLO}$ do
    $C_{X^i_{PLO}} := \texttt{count\_GPO}(X^i_{PLO}, \mathbf{X}_{PLO})$ ▷ Get number of GPO
    for $\forall X^j_{PLO} \in \mathbf{X}_{PLO}$ do
        if $X^j_{PLO} \subset X^i_{PLO}$ then
            $E^{subset}_{X^i_{PLO}} := \texttt{append}(E^{subset}_{X^i_{PLO}}, X^j_{PLO})$ ▷ Get covering
        end if
    end for
end for
$E = \texttt{get\_edge\_weights}(\mathbf{X}_{PLO}, \mathcal{N})$    ▷ Generate edges
return $(\mathbf{X}_{PLO}, B, C, E, E^{subset})$
```

4.2 Pareto local optima networks

Our first approach to transfer the LON to multi/many-objective problems is to initially identify the unique X^i_{PLO} returned by PLS, which may include intersecting X^i_{PLO}. We can then identify all solutions which lead to each of these X^i_{PLO}, and therefore identify the basin sizes. There remains the issue that some X^i_{PLO} may be proper subsets of others, and therefore effectively their basin is larger than merely those members whose PLS lead to them exactly. We address this here by additionally highlighting directed edges where a mode completely contains the solutions defining another.

We now describe the construction of LONs using PLS and PHC, which we call Pareto LONs (or PLONS). In PLONs the vertices represent the X^i_{PLO} returned from PLS. Their other properties are as follows:

1. vertices are coloured proportional to the number of GPO they contain;
2. the vertex radii are proportional to basin size;
3. there is a directed edge between vertex A and vertex B if it is possible to move from A to B using a transition to the other (illustrated using escape edges [18] here), with the width of the edge denoting the number of neighbours of the source X^i_{PLO} which lead to this transition;
4. highlighted edges denote if the destination is a subset of the source.

An algorithm outlining the generation of a PLON is provided in Algorithm 5. Example PLONs are provided in Figure 8 for the problems illustrated in Figure 1 and Figure 3.

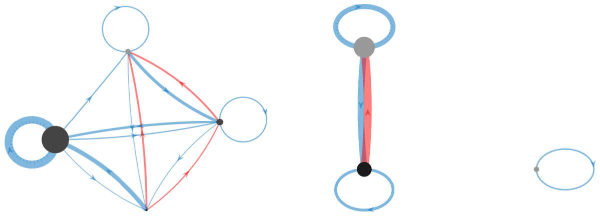


Figure 8: PLONs corresponding to the problems illustrated in Figure 1 (*left*) and Figure 3 (*right*). Edges in red denote where the X_{PLO} of the destination node is a subset of the X_{PLO} of the source node.

Algorithm 6 DNON generation.

```
$D = \emptyset$    ▷ Initial set of dominance neutral locations
$\mathbf{X}_{DNO} = \emptyset$    ▷ Initial set of vertex sets
$\mathbf{B} = \emptyset$    ▷ Initial map to basin size
$\mathbf{C} = \emptyset$    ▷ Initial map to vertex colours
$E^{subset} = \emptyset$    ▷ Initial map of vertices to subsets
for $\forall \mathbf{x} \in X$ do
    $Y := \texttt{PHC}(\mathbf{x})$  ▷ Get list of all locations resulting from PHC
    for each $\mathbf{y} \in Y$ do
        if $\mathbf{y} \in D$ then
            $B_\mathbf{y} := B_\mathbf{y} + 1$
        else
            $D = D \cup \{\mathbf{y}\}$
            $B_\mathbf{y} := 1$
        end if
    end for
end for
$(\mathbf{X}_{DNO}, B) := \texttt{agglomerate}(D, B)$ ▷ Get walkable components
$C := \texttt{get\_number\_of\_GPO\_per\_vertex}(\mathbf{X}_{DNO})$
$E = \texttt{get\_edge\_weights}(\mathbf{X}_{DNO}, \mathcal{N})$    ▷ Generate edges
return $(\mathbf{X}_{DNO}, B, C, E)$
```

4.3 Dominance-neutral optima networks

Using PHC, and X_{DNO}, we may construct Dominance-Neutral Optima Networks (DNONs) with the following properties:

1. vertices are coloured by the number of GPO contained within the corresponding X^i_{DNO};
2. the vertices represent the unique *disjoint* subsets of the X^i_{DNO}, constructed by walking across solutions returned by PHC;
3. the vertex radii are proportional to basin size (the proportion of walks from each solution in X where PHC leads to a member of the X^i_{DNO} represented by the vertex);
4. there is an edge between two vertices A and B if it is possible to move from one vertex to the other, with the width of the edge denoting how easy this is.

An algorithm outlining the generation of a DNON is provided in Algorithm 6. Example DNONs are provided in Figure 9 for the problems illustrated in Figure 1 and Figure 3.

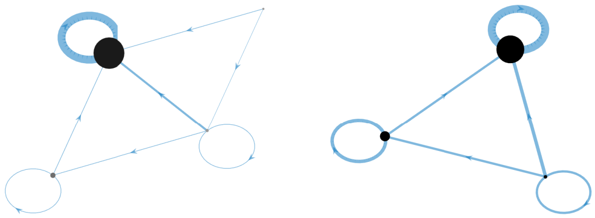


Figure 9: DNONs corresponding to the problems illustrated in Figure 1 (*left*) and Figure 3 (*right*).

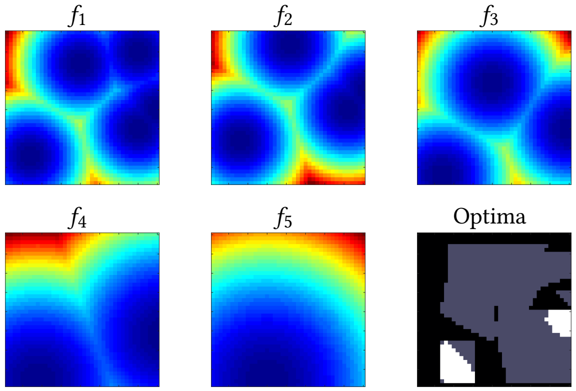


Figure 10: Fitness landscapes under the five different relatively smooth objective functions. Bottom right panel highlights the GPO, PLO and DNO sets.

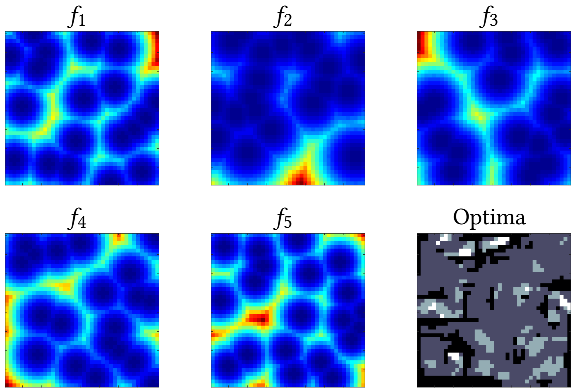


Figure 11: Fitness landscapes under the five different relatively rugged objective functions. Bottom right panel highlights the GPO, PLO and DNO sets.

5 MANY-OBJECTIVE ILLUSTRATION

Our previous illustrations have been on small ($|X| = 25$) bi-objective problems. We now illustrate the PLOS-net, PLON, and DNON on two larger ($|X| = 1600$) five-objective (i.e. many-objective) problems, one comprised of relatively smooth functions, and the other composed of relatively rugged functions. To aid visualisation and understanding, we again use a problem where the search domain is in the plane. The images of these domains under the five different objective functions of the two problems are shown in Figures 10 and 11. The bottom right panel of each of these figures colours each cell (solution) by type. White are GPO, light grey are PLO (as are all black cells) and dark grey are DNO (as are all light grey and white cells). None of the black cells act as attractors under any optima category.

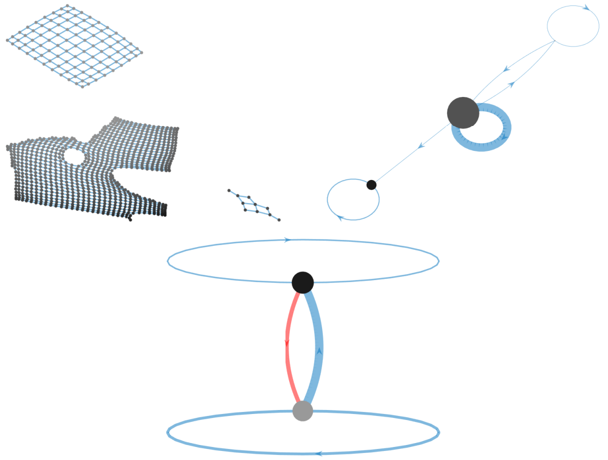


Figure 12: PLOS-net (*top-left*), DNON (*top-right*) and PLON (*bottom*) for the problem illustrated in Figure 10.

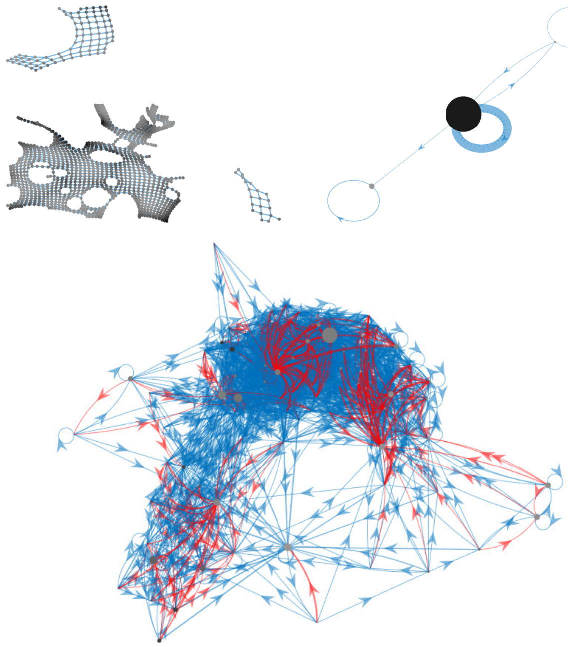


Figure 13: PLOS-net (*top-left*), DNON (*top-right*) and PLON (*bottom*) for the problem illustrated in Figure 11.

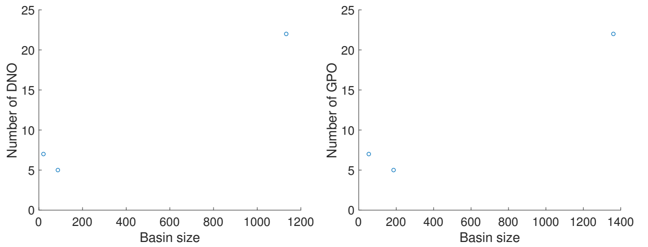


Figure 14: DNON basin size versus number of DNO (*left*) and DNON basin size versus number of GPO (*right*) using the problem illustrated in Figure 11.

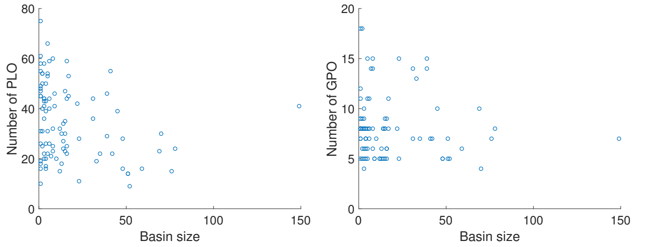


Figure 15: PLON basin size versus number of PLO (*left*) and PLON basin size versus number of GPO (*right*) using the problem illustrated in Figure 11.

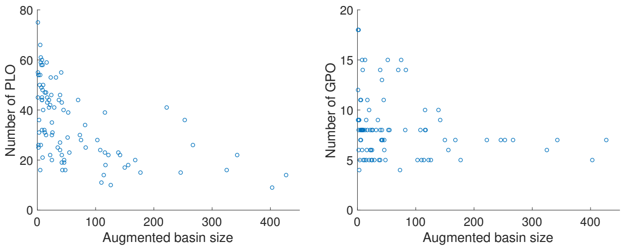


Figure 16: Augmented PLON basin size versus number of PLO (*left*) and augmented PLON basin size versus number of GPO (*right*) using the problem illustrated in Figure 11.

In the smooth landscape with only a few minima under each individual objective, there are two X^i_{PLO} in the PLON, one of which is a subset of the other, and three X^j_{DNO} in the DNON, with the global Pareto optima split amongst two of them (see Figure 12). There is a marked contrast in the LONs generated for the rugged landscape. Again there are only three X^j_{DNO} using the DNON, although the basin sizes are different than for the first problem (the PLOS-net has correspondingly three maximal subgraph components). In contrast, there are 93 X^i_{PLO} under the PLON representation — although nearly three-quarters of these are subsets of others (see Figure 13). The GPO are distributed amongst all vertices in both the DNON and PLON for this problem.

Figure 14 shows the distribution of basin size versus number of DNO in a vertex, and basin size versus the number of GPO in a vertex for the DNON on the rugged problem (correspondingly the number of vertices in each component of the PLOS-net). As can be seen, the distributions between the two subplots are fairly consistent, and the positive trend obvious (albeit on a small number of points). Figure 15 shows the distribution of basin size versus number of PLO in a vertex, and basin size versus the number of GPO in a vertex for the PLON on the rugged problem. In this particular instance there is a poor correlation between basin size and number of PLO or number of GPO (with a Spearman's ρ of -0.31 and -0.17 respectively) — however, once basins which lead to vertices which are supersets of other vertices are taken into consideration we find the scatter plots in Figure 16, with a Spearman's ρ of -0.65 and -0.25 for the correlation between the augmented basin size and PLO and GPO respectively, highlighting the difficulty of this particular problem.

As mentioned in Section 4.3, it is well-known that dominance becomes much less discriminatory as the number of objectives increases, making the landscape look largely neutral under local dominance comparisons. This is likely the root cause of the small number of vertices in the DNON representation, and PLOS-net components, though we have yet to undertake a rigourous examination of DNON properties as the number of objectives increases.

6 DISCUSSION

We have presented two new approaches for visualising multi/many-objective landscapes using the LON framework.

The first, the PLON exploits Pareto local search to build the LON and define the set memberships of each mode, mimicking the *set-based* nature of typical multi-objective search. Its construction is however relatively costly, and the inherent nature of the X^i_{PLO} means there are two different forms of edges between vertices which require representation in the single graph. We note that alternatives could be to employ a graph *pair* representation, or employing a partitioned graph visualisation, akin to that used in e.g. [16] for funnels to denote this.

The second, the DNON uses point-based local search, with solutions requiring agglomeration after the individual Pareto hillclimbs have all completed to obtain the vertices. Nevertheless, its computational cost is much lower than the PLON, and the sets of solutions determining the vertices do not intersect. As such, it is more amenable to use in a estimation (sampling-based) context. It may also be effectively paired with the PLOS-net visualisation, to gain additional information as to how the separate DNO components are formed.

Although our illustrations are on problems with neighbourhoods on a grid, this was merely to aid visualisation and explanation. Both PLONs and DNONs may be applied generally to any multi-objective problem where a neighbourhood function is available for solutions, and we look forward to developing generic and efficient generators for this (building on our previous work developing efficient packages in the uni-objective domain [3]).

Matlab code to regenerate the examples and visualisations in this paper is available at https://github.com/fieldsend.

ACKNOWLEDGMENTS

This work was supported by the Engineering and Physical Sciences Research Council [grant number EP/N017846/1].

REFERENCES

[1] David W. Corne and Joshua D. Knowles. 2007. Techniques for Highly Multiobjective Optimisation: Some Nondominated Points Are Better Than Others. In *Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation (GECCO ’07)*. ACM, New York, NY, USA, 773–780.

[2] Fabio Daolio, Sébastien Verel, Gabriela Ochoa, and Marco Tomassini. 2010. Local Optima Networks of the Quadratic Assignment Problem. In *IEEE Congress on Evolutionary Computation*. 1–8.

[3] Jonathan E. Fieldsend. 2018. Computationally Efficient Local Optima Network Construction. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion (GECCO ’18)*. ACM, New York, NY, USA, 1481–1488.

[4] Jonathan E. Fieldsend, Tinkle Chugh, Richard Allmendinger, and Kaisa Miettinen. 2019. A Feature Rich Distance-Based Many-Objective Visualisable Test Problem Generator. In *Genetic and Evolutionary Computation Conference, GECCO2019, Proceedings*. ACM. https://doi.org/10.1145/3321707.3321727

[5] Tobias Glasmachers. 2017. A Fast Incremental BSP Tree Archive for Nondominated Points. In *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 252–266.

[6] Leticia Hernando, Fabio Daolio, Nadarajen Veerapen, and Gabriela Ochoa. 2017. Local Optima Networks of the Permutation Flowshop Scheduling Problem: Makespan vs. total flow time. In *2017 IEEE Congress on Evolutionary Computation (CEC)*. 1964–1971.

[7] David Iclanzan, Fabio Daolio, and Marco Tomassini. 2014. Data-driven Local Optima Network Characterization of QAPLIB Instances. In *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation (GECCO ’14)*. ACM, New York, NY, USA, 453–460.

[8] Andrzej Jaszkiewicz and Thibaut Lust. 2018. ND-Tree-Based Update: A Fast Algorithm for the Dynamic Nondominance Problem. *IEEE Transactions on Evolutionary Computation* 22, 5 (2018), 778–791.

[9] Joshua Knowles and David Corne. 1999. The Pareto archived evolution strategy: A new baseline algorithm for Pareto multiobjective optimisation. In *Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on*, Vol. 1. IEEE, 98–105.

[10] Arnaud Liefooghe, Bilel Derbel, Sébastien Verel, Manuel López-Ibáñez, Hernán Aguirre, and Kiyoshi Tanaka. 2018. On Pareto Local Optimal Solutions Networks. In *Parallel Problem Solving from Nature – PPSN XV*, Anne Auger, Carlos M. Fonseca, Nuno Lourenço, Penousal Machado, Luís Paquete, and Darrell Whitley (Eds.). Springer International Publishing, Cham, 232–244.

[11] Gabriela Ochoa and Sebastian Herrmann. 2018. Perturbation Strength and the Global Structure of QAP Fitness Landscapes. In *Parallel Problem Solving from Nature – PPSN XV*, Anne Auger, Carlos M. Fonseca, Nuno Lourenço, Penousal Machado, Luís Paquete, and Darrell Whitley (Eds.). Springer International Publishing, Cham, 245–256.

[12] Gabriela Ochoa, Marco Tomassini, Sebástien Vérel, and Christian Darabos. 2008. A Study of NK Landscapes’ Basins and Local Optima Networks. In *Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation (GECCO ’08)*. ACM, New York, NY, USA, 555–562.

[13] Gabriela Ochoa, Nadarajen Veerapen, Darrell Whitley, and Edmund K. Burke. 2016. The Multi-Funnel Structure of TSP Fitness Landscapes: A Visual Exploration. In *Artificial Evolution*, Stéphane Bonnevay, Pierrick Legrand, Nicolas Monmarché, Evelyne Lutton, and Marc Schoenauer (Eds.). Springer International Publishing, Cham, 1–13.

[14] Gabriela Ochoa, Sébastien Verel, and Marco Tomassini. 2010. First-Improvement vs. Best-Improvement Local Optima Networks of NK Landscapes. In *International Conference on Parallel Problem Solving from Nature*. Springer, 104–113.

[15] Luis Paquete, Tommaso Schiavinotto, and Thomas Stützle. 2007. On local optima in multiobjective combinatorial optimization problems. *Annals of Operations Research* 156, 1 (2007), 83.

[16] Sarah L Thomson, Fabio Daolio, and Gabriela Ochoa. 2017. Comparing communities of optima with funnels in combinatorial fitness landscapes. In *Proceedings of the Genetic and Evolutionary Computation Conference*. ACM, 377–384.

[17] Marco Tomassini, Sébastien Vérel, and Gabriela Ochoa. 2008. Complex-network analysis of combinatorial spaces: The NK landscape case. *Physical Review E* 78 (Dec 2008), 066114. Issue 6.

[18] Sébastien Vérel, Fabio Daolio, Gabriela Ochoa, and Marco Tomassini. 2012. Local Optima Networks with Escape Edges. In *Artificial Evolution*, Jin-Kao Hao, Pierrick Legrand, Pierre Collet, Nicolas Monmarché, Evelyne Lutton, and Marc Schoenauer (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 49–60.

[19] Sébastien Verel, Fabio Daolio, Gabriela Ochoa, and Marco Tomassini. 2018. Sampling Local Optima Networks of Large Combinatorial Search Spaces: The QAP Case. In *Parallel Problem Solving from Nature – PPSN XV*, Anne Auger, Carlos M. Fonseca, Nuno Lourenço, Penousal Machado, Luís Paquete, and Darrell Whitley (Eds.). Springer International Publishing, Cham, 257–268.

[20] Sébastien Verel, Arnaud Liefooghe, Laetitia Jourdan, and Clarisse Dhaenens. 2011. Pareto local optima of multiobjective NK-landscapes with correlated objectives. In *European Conference on Evolutionary Computation in Combinatorial Optimization*. Springer, 226–237.

[21] Sébastien Verel, Gabriela Ochoa, and Marco Tomassini. 2011. Local optima networks of NK landscapes with neutrality. *IEEE Transactions on Evolutionary Computation* 15, 6 (2011), 783–797.