3. Local optima network

Fitness landscape analysis for understanding and designing local search heuristics

SÉBASTIEN VEREL

LISIC - Université du Littoral Côte d’Opale, Calais, France
http://www-lisic.univ-littoral.fr/~verel/

The 51st CREST Open Workshop
Tutorial on Landscape Analysis
University College London

27th, February, 2017
Outline of this part

- **Basis of fitness landscape**:
  - introductory example *(Done)*
  - brief history and background of fitness landscape *(Done)*
  - fundamental definitions *(Done)*

- **Geometries of fitness landscapes**:
  - multimodality *(Done)*
  - ruggedness *(Done)*
  - neutrality *(Done)*
  - neutral networks *(Done)*

- **Local optima network**:
  - Definition inspired by complex systems science
  - Features of the network, design and performance
  - Performance prediction and portfolio
Join work

Gabriela Ochoa, University of Stirling, Scotland,
Marco Tomassini, University of Lausanne, Switzerland,
Fabio Daolio, University of Stirling, Scotland,
Key idea: Complex system tools

Principle of variables aggregation

A model for dynamical systems with two scales (time/space)

- Split the state space according to the different scales
- Study the system at the large scale
Key idea : Complex system tools

**Principle of variables aggregation**

A model for dynamical systems with two scales (time/space)

- Split the state space according to the different scales
- Study the system at the large scale

**Variables aggregation for fitness landscape**

- At solutions level (small scale):
  - Stochastic local search operator,
  - Exponential number of solutions,
  - Exponential size of the stochastic matrix of the process (Markov chain)

- Projection on a relevant space:
  - Reduce the size of state space
  - Potentially loose some information
  - Relevant information remains when:
    \[ p(op(x)) \approx op'(p(x)) \]
Key idea: Complex system tools

**Principle of variables aggregation**

A model for dynamical systems with two scales (time/space):

- Split the state space according to the different scales
- Study the system at the large scale

**Variables aggregation for fitness landscape**

- At solutions level (small scale):
  - Stochastic local search operator,
  - Exponential number of solutions,
  - Exponential size of the stochastic matrix of the process (Markov chain)

- Projection on a relevant space:
  - Reduce the size of state space
  - Potentially loose some information
  - Relevant information remains when:
    \[ p(op(x)) \approx op'(p(x)) \]
Key idea: Complex system tools

Complex network

Bring the tools of complex networks analysis to the study the structure of combinatorial fitness landscapes

Methodology

- **Design a network** that represents the landscape
  - Vertices: local optima
  - Edges: a notion of adjacency between local optima

- **Extract features**: “complex” network analysis

- **Use the network features**: search algorithm design, difficulty, etc.

Complex networks

**Scale free network**
(Watts and Strogatz, 1998 [WS98])

**Small world network**
(Barabasi and Albert, 1999 [BA99])
Energy surface and inherent networks

Inherent network

- **Nodes**: energy minima
- **Edges**: two nodes are connected if the energy barrier separating them is sufficiently low (transition state)

(a) Energy surface
(b) Contours plot: partition of states space into basins of attraction
(c) Landscape as a network

Basins of attraction in combinatorial optimization

Example of small $NK$ landscape with $N = 6$ and $K = 2$

- Bit strings of length $N = 6$
- $2^6 = 64$ solutions
- one point = one solution
Basins of attraction in combinatorial optimization
Example of small $NK$ landscape with $N = 6$ and $K = 2$

- Bit strings of length $N = 6$
- Neighborhood size = 6
- Line between points = solutions are neighbors
- Hamming distances between solutions are preserved (except for at the border of the cube)
Basins of attraction in combinatorial optimization

Example of small $NK$ landscape with $N = 6$ and $K = 2$

Color represent fitness value
- high fitness
- low fitness
Basins of attraction in combinatorial optimization

Example of small NK landscape with $N = 6$ and $K = 2$

- Color represent fitness value
  - red: high fitness
  - blue: low fitness
- Point towards the solution with highest fitness in the neighborhood

Exercise:
Why not make a Hill-Climbing walk on it?
Basins of attraction in combinatorial optimization

Example of small $NK$ landscape with $N = 6$ and $K = 2$

- Each color corresponds to one basin of attraction
- Basins of attraction are interlinked and overlapped
- Basins have no "interior"
Basins of attraction in combinatorial optimization
Example of small $NK$ landscape with $N = 6$ and $K = 2$

- Basins of attraction are interlinked and overlapped!
- Most neighbors of a given solution are outside its basin
Local optima network

Nodes:
- local optima

Edges:
- transition probabilities
Basin of attraction

Hill-Climbing algorithm (best-improvement)

Choose initial solution \( x \in X \)

repeat

choose \( x' \in \mathcal{N}(x) \) such that \( f(x') = \max_{y \in \mathcal{N}(x)} f(y) \)

if \( f(x) < f(x') \) then

\( x \leftarrow x' \)

end if

until \( x \) is a Local optimum

Basin of attraction of \( x^* \):

\[
b_{x^*} = \{ x \in X \mid \text{HillClimbing}(x) = x^* \}.
\]
local optima network

**Definition : Local Optima Network (LON)**

Orienter weighted graph \((V, E, w)\)

- **Notes** \(V\) : set of local optima \(\{LO_1, \ldots, LO_n\}\)
- **Edges** \(E\) : notion of connectivity between local optima
local optima network

**Definition : Local Optima Network (LON)**

Orienter weighted graph \((V, E, w)\)

- **Notes** \(V\) : set of local optima \(\{LO_1, \ldots, LO_n\}\)
- **Edges** \(E\) : notion of connectivity between local optima

### 2 possible definitions of edges

- **Basin-transition edges** :
  transition between random solutions from basin \(b_i\) to basin \(b_j\)
  ([OTVD08], [VOT08], [TVO08], [VOT10])

- **Escape edges** :
  transition from Local Optimum \(i\) to basin \(b_j\)
  (EA 2011, GECCO 2012, PPSN 2012, EA 2013 [DVOT13])
Basin-transition edges: random transition between basins

**Edges**

\[ e_{ij} \text{ between } LO_i \text{ and } LO_j \text{ if } \exists x_i \in b_i \text{ and } x_j \in b_j : x_j \in \mathcal{N}(x_i) \]

**Prob. from solution** \( x \) **to solution** \( x' \)

\[ p(x \to x') = \Pr(x' = \text{op}(x)) \]

**Prob. from solution** \( s \) **to basin** \( b_j \)

\[ p(x \to b_j) = \sum_{x' \in b_j} p(x \to x') \]

**Weights:** Transition prob. from basin \( b_i \) **to basin** \( b_j \)

\[ w_{ij} = p(b_i \to b_j) = \frac{1}{\#b_i} \sum_{x \in b_i} p(s \to b_j) \]
**Basin-transition edges**: random transition between basins

**Edges**

$e_{ij}$ between $LO_i$ and $LO_j$ if $\exists x_i \in b_i$ and $x_j \in b_j : x_j \in \mathcal{N}(x_i)$

**Prob. from solution $x$ to solution $x'$**

$$p(x \rightarrow x') = \Pr(x' = op(x))$$

For example, $X = \{0, 1\}^N$ and bit-flip operator

if $x' \in \mathcal{N}(x)$, $p(x \rightarrow x') = \frac{1}{N}$, otherwise $p(x \rightarrow x') = 0$

**Prob. from solution $s$ to basin $b_j$**

$$p(x \rightarrow b_j) = \sum_{x' \in b_j} p(x \rightarrow x')$$

**Weights**: Transition prob. from basin $b_i$ to basin $b_j$

$$w_{ij} = p(b_i \rightarrow b_j) = \frac{1}{\#b_i} \sum_{x \in b_i} p(s \rightarrow b_j)$$
LON with Escape edges

Definition: Local Optima Network (LON)

Orienter weighted graph \((V, E, w)\)

- Notes \(V\): set of local optima \(\{LO_1, \ldots, LO_n\}\)
- Edges \(E\): notion of connectivity between local optima

Escape edges

Edge \(e_{ij}\) between \(LO_i\) and \(LO_j\)

if \(\exists x: distance(LO_i, x) \leq D\) and \(x \in b_j\).

Weights

\[ w_{ij} = \# \{ x \in X \mid d(LO_i, x) \leq D, x \in b_j \} \]

can be normalized by the number of solutions at distance \(D\).
Definition: Local Optima Network (LON)

Orienter weighted graph \((V, E, w)\)

- **Notes** \(V\): set of local optima \(\{LO_1, \ldots, LO_n\}\)
- **Edges** \(E\): notion of connectivity between local optima

**Escape edges**

Edge \(e_{ij}\) between \(LO_i\) and \(LO_j\)

if \(\exists x: \text{distance}(LO_i, x) \leq D\) and \(x \in b_j\).

**Weights**

\(w_{ij} = \#\{x \in X \mid d(LO_i, x) \leq D, x \in b_j\}\)

- can be normalized by the number of solutions at distance \(D\)
Basins of attraction features

- **Basin of attraction:**
  - Size:
    - average, distribution, etc.
  - Fitness of local optima:
    - average, distribution, correlation, etc.
NK-landscapes
[Kauffman 1993] [Kau93]

\[ x \in \{0, 1\}^n \quad f(x) = \frac{1}{n} \sum_{i=1}^{n} f_i(x_j, x_{i_1}, \ldots, x_{i_k}) \]

Two parameters
- Problem size \( n \)
- Non-linearity \( k < n \)
  (multi-modality, epistatic interactions)
  - \( k = 0 \): linear problem, one single maxima
  - \( k = n - 1 \): random problem, number of local optima \( \frac{2^N}{N+1} \)

remarks: ”same” results with QAP, flow shop.
Global optimum basin size vs. non-linearity degree $k$

- Basin size of maximum decreases exponentially with non-linearity degree $k$
- $\Rightarrow$ Difficulty of (best-improvement) hill-climber from a random solution

Size of the global maximum basin as a function of non-linearity degree $k$
Distribution of basin sizes

Cumulative distribution of basin sizes for $n = 18$ and $k = 4$

- Log-normal cumulative distribution (not uniform!):
  - large number of small basins,
  - small number of large basins.
- Effect of non-linearity:
  the distribution becomes more uniform with non-linearity degree $k$
Fitness of local optima vs. basin size

The highest, the largest!

- On average, the global optimum easier to find than one given other local optimum
- But more difficult to find, as the number of local optima increases exponentially with increasing K

Correlation fitness of local optima vs. their corresponding basins sizes
Question:
Do basins look like a "mountain" with interior and border?

Solution: $\in$ interior if all neighbors are in the same basin
Basin: Interior and border sizes

**Question:**
Do basins look like a "mountain" with interior and border?

**Solution:**
(solution $\in$ interior if all neighbors are in the same basin)

**Answer:**
- Interior is very small
- Nearly all solutions $\in$ border
Features of local optima network

- \(nv\) : \#vertices
- \(lv\) : avg path length \(d_{ij} = 1/w_{ij}\)
- \(lo\) : path length to best
- \(fnn\) : fitness corr. 
  \((f(x), f(y)) \text{ with } (x, y) \in E\)
- \(wii\) : self loops
- \(wcc\) : weighted clust. coef.
- \(zout\) : out degree
- \(y2\) : disparity
- \(knn\) : degree corr.
  \((\text{deg}(x), \text{deg}(y)) \text{ with } (x, y) \in E\)
Some formal definitions

**Weighted clustering coefficient**

Local density of the network

\[
c^w(i) = \frac{1}{s_i(k_i - 1)} \sum_{j,h} \frac{w_{ij} + w_{ih}}{2} a_{ij} a_{jh} a_{hi}
\]

where \(s_i = \sum_{j \neq i} w_{ij}\), \(a_{nm} = 1\) if \(w_{nm} > 0\), \(a_{nm} = 0\) if \(w_{nm} = 0\) and

\(k_i = \sum_{j \neq i} a_{ij}\).

**Disparity**

Dishomogeneity of nodes with a given degree

\[
Y_2(i) = \sum_{j \neq i} \left( \frac{w_{ij}}{s_i} \right)^2
\]
A fitness landscape analysis approach

- Link between LON features and difficulty:
  small size instances of NK-landscapes
- Analysis of the LON structure:
  small size instances of NK-landscapes, QAP and FSSP
- Design of one local search component:
  small size instances of NK-landscapes and FSSP
- Explication de performance avec les propriétés du ROL:
  corrélation simple, petites instances, NK et QAP
  corrélation multi-linéaire, petites instances, FSSP
- Prédictiion de performance basée sur le ROL:
  grandes instances NK et QAP
- Portfolio d’algorithmes:
  grandes instances NK et QAP
Structure of Local Optima Network

- NK-landscapes (small instances): Most of the features are correlated with $K$ relevance of LON definition

- LON is **not a random** network (NK, QAP, FSSP): Highly clustered network, Distribution of weights and degrees have long tail, etc.
Example: clustering coefficient for NK-landscapes

- Network highly clustered
- Clustering coefficient decreases with the degree of non-linearity
LON to compare of problem difficulty
Local Optima Network for Quadratic Assignment Problem (QAP) [DTVO11]

→ Community detection in LON for
  Random instance
  Real-like instance

Structure of the LON related to problem difficulty
Comparaison of **operators** for Flow Shop Scheduling Problem
Comparaison of **pivot rule** in hill-climbing for NK-landscapes

<table>
<thead>
<tr>
<th>$K$</th>
<th>$\bar{n}_e / \bar{n}_v^2$</th>
<th>$Y$</th>
<th>$\bar{d}$</th>
<th>$\bar{d}_{\text{best}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>b-LON</td>
<td>f-LON</td>
<td>b-LON</td>
<td>f-LON</td>
</tr>
<tr>
<td>2</td>
<td>0.81</td>
<td>0.96</td>
<td>0.326</td>
<td>0.110</td>
</tr>
<tr>
<td>4</td>
<td>0.60</td>
<td>0.92</td>
<td>0.137</td>
<td>0.033</td>
</tr>
<tr>
<td>6</td>
<td>0.32</td>
<td>0.79</td>
<td>0.084</td>
<td>0.016</td>
</tr>
<tr>
<td>8</td>
<td>0.17</td>
<td>0.65</td>
<td>0.062</td>
<td>0.011</td>
</tr>
<tr>
<td>10</td>
<td>0.09</td>
<td>0.53</td>
<td>0.050</td>
<td>0.009</td>
</tr>
<tr>
<td>12</td>
<td>0.05</td>
<td>0.44</td>
<td>0.043</td>
<td>0.008</td>
</tr>
</tbody>
</table>
Information given by the local optima network

Advanced questions

- Can we explain the performance from the LON features?
- Can we predict the performance from the LON features?
- Can we select the relevant algorithm based on the LON features?
LON features vs. performance: simple correlation

Algorithm: Iterated Local Search on NK-landscapes with $N = 18$
Performance: $\text{ert} = \mathbb{E}(T_S) + \left(\frac{1-p_s}{p_s}\right) T_{\text{max}}$

<table>
<thead>
<tr>
<th>$n_v$</th>
<th>$\bar{d}_{\text{best}}$</th>
<th>$\bar{d}$</th>
<th>fnn</th>
<th>$w_{ij}$</th>
<th>$\bar{C}^w$</th>
<th>zout</th>
<th>$\bar{Y}$</th>
<th>knn</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.885</td>
<td>0.915</td>
<td>0.006</td>
<td>−0.830</td>
<td>−0.883</td>
<td>−0.875</td>
<td>0.885</td>
<td>−0.883</td>
<td>−0.850</td>
</tr>
</tbody>
</table>
ILS Performance vs LON Metrics

NK-landscapes [DVOT12]

Expected running times

vs.

Average shortest path to the global optimum.
Expected running times
vs.
Average shortest path to the global optimum.
LON features vs. performance: multi-linear regression

1. Multiple **linear** regression on all possible predictors:

\[
\log(ert) = \beta_0 + \beta_1 k + \beta_2 \log(nv) + \beta_2 lo + \cdots + \beta_{10} knn + \varepsilon
\]

2. Step-wise **backward elimination** of each predictor in turn.

<table>
<thead>
<tr>
<th>Predictor</th>
<th>$\hat{\beta}_i$</th>
<th>Std. Error</th>
<th>$p$-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Intercept)</td>
<td>10.3838</td>
<td>0.58512</td>
<td>$9.24 \cdot 10^{-47}$</td>
</tr>
<tr>
<td>lo</td>
<td>0.0439</td>
<td>0.00434</td>
<td>$1.67 \cdot 10^{-20}$</td>
</tr>
<tr>
<td>zout</td>
<td>$-0.0306$</td>
<td>0.00831</td>
<td>$2.81 \cdot 10^{-04}$</td>
</tr>
<tr>
<td>y2</td>
<td>$-7.2831$</td>
<td>1.63038</td>
<td>$1.18 \cdot 10^{-05}$</td>
</tr>
<tr>
<td>knn</td>
<td>$-0.7457$</td>
<td>0.40501</td>
<td>$6.67 \cdot 10^{-02}$</td>
</tr>
</tbody>
</table>

Multiple R-squared: 0.8494, Adjusted R-squared: 0.8471.
LON features vs. performance: multi-linear regression

For **Flow Shop Scheduling Problem** using exhaustive selection

<table>
<thead>
<tr>
<th>#P</th>
<th>$\log(N_V)$</th>
<th>$CC^w$</th>
<th>$F_{nn}$</th>
<th>$k_{nn}$</th>
<th>$r$</th>
<th>$\log(L_{opt})$</th>
<th>$\log(L_V)$</th>
<th>$w_{ii}$</th>
<th>$Y_2$</th>
<th>$k_{out}$</th>
<th>$C_p$</th>
<th>$adjR^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2.13</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>265.54</td>
<td>0.574</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1.43</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>64.06</td>
<td>0.675</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1.481</td>
<td>0.895</td>
<td></td>
<td></td>
<td></td>
<td>16.48</td>
<td>0.700</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1.473</td>
<td>0.540</td>
<td></td>
<td></td>
<td></td>
<td>8.75</td>
<td>0.704</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1.470</td>
<td>0.528</td>
<td></td>
<td></td>
<td></td>
<td>5.97</td>
<td>0.706</td>
</tr>
</tbody>
</table>
Sampling methodology for large size instances

From the sampling of large-size complex network:

- Random walk on the network
- Breadth-First-Search

**Procedure** LONSampling\((d, m, l)\)

\[
x_0 \leftarrow hc(x) \text{ with } x \text{ random solution}
\]

**for** \( t \leftarrow 0, \ldots, l - 1 \) **do**

- Snowball\((d, m, x_t)\)
- \( x_{t+1} \leftarrow \text{RandomWalkStep}(x_t) \)

**end for**
Set of estimated LON features for large size instances

<table>
<thead>
<tr>
<th>LON metrics</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>fit</strong> Average fitness of local optima in the network.</td>
</tr>
<tr>
<td><strong>wii</strong> Average weight of self-loops.</td>
</tr>
<tr>
<td><strong>zout</strong> Average outdegree.</td>
</tr>
<tr>
<td><strong>y_2</strong> Average disparity for outgoing edges.</td>
</tr>
<tr>
<td><strong>knn</strong> Weighted assortativity.</td>
</tr>
<tr>
<td><strong>wcc</strong> Weighted clustering coefficient.</td>
</tr>
<tr>
<td><strong>fnn</strong> Fitness-fitness correlation on the network.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Metrics from the sampling procedure</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>lhc</strong> Average length of hill-climbing to local optima.</td>
</tr>
<tr>
<td><strong>mlhc</strong> Maximum length of hill-climbing to local optima.</td>
</tr>
<tr>
<td><strong>nhc</strong> Number of hill-climbing paths to local optima.</td>
</tr>
</tbody>
</table>
Performance prediction based on estimated features

- Optimization scenario using off-the-shelf metaheuristics: TS, SA, EA, ILS on 450 instances for NK and QAP.
- Performance measures:
  - average fitness / average rank
- Model of regression:
  - linear model / random forest
- Set of features:
  - basic: 1st autocorr. coeff. of fitness (rw of length $10^3$), Avg. fitness of local optima ($10^3$ hc), Avg. length to reach local optima ($10^3$ hc).
  - lon: see previous,
  - all: basic and lon features
- Quality measure of regression:
  - $R^2$ on cross-validation (repeated random sub-sampling)
$R^2$ on cross-validation for NK-landscapes and QAP

Sampling parameters: length $l = 100$, sampled edge $m = 30$, and deep $d = 2$

<table>
<thead>
<tr>
<th>Mod.</th>
<th>Feat.</th>
<th>Perf.</th>
<th>NK</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th>QAP</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>l m</td>
<td>basic</td>
<td>fit</td>
<td>0.8573</td>
<td>0.8739</td>
<td>0.8763</td>
<td>0.8874</td>
<td>0.8737</td>
<td>-38.42</td>
<td>0.9995</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.9997</td>
<td>0.9998</td>
<td>0.9997</td>
</tr>
<tr>
<td>l m</td>
<td>lon</td>
<td>fit</td>
<td>0.8996</td>
<td>0.9015</td>
<td>0.9061</td>
<td>0.8954</td>
<td>0.9007</td>
<td>0.9996</td>
<td>0.9997</td>
<td>0.9999</td>
<td>0.9997</td>
<td>0.9997</td>
<td>0.9997</td>
<td>0.9997</td>
</tr>
<tr>
<td>l m</td>
<td>all</td>
<td>fit</td>
<td>0.9356</td>
<td>0.9455</td>
<td>0.9442</td>
<td>0.9501</td>
<td>0.9439</td>
<td>0.9996</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.9997</td>
<td>0.9997</td>
<td>0.9997</td>
<td>0.9997</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>0.8591</td>
<td>0.9147</td>
<td>0.6571</td>
<td>0.6401</td>
<td>0.7678</td>
<td>0.2123</td>
<td>0.9007</td>
<td>0.9029</td>
<td>0.9029</td>
<td>0.9029</td>
<td>0.9029</td>
<td>0.9029</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rank</td>
<td>0.9517</td>
<td>0.9332</td>
<td>0.7783</td>
<td>0.7166</td>
<td>0.8449</td>
<td>0.7893</td>
<td>0.9673</td>
<td>0.8794</td>
<td>0.9015</td>
<td>0.8844</td>
<td>0.9029</td>
<td>0.8826</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rank</td>
<td>0.9534</td>
<td>0.9355</td>
<td>0.7809</td>
<td>0.7177</td>
<td>0.8469</td>
<td>0.6199</td>
<td>0.9340</td>
<td>0.8577</td>
<td>0.9029</td>
<td>0.8826</td>
<td>0.9029</td>
<td>0.8826</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>0.9043</td>
<td>0.9104</td>
<td>0.9074</td>
<td>0.8871</td>
<td>0.9023</td>
<td>0.8811</td>
<td>0.8820</td>
<td>0.8806</td>
<td>0.8801</td>
<td>0.8809</td>
<td>0.8809</td>
<td>0.8809</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rank</td>
<td>0.9513</td>
<td>0.9433</td>
<td>0.7729</td>
<td>0.8075</td>
<td>0.8687</td>
<td>0.9375</td>
<td>0.9653</td>
<td>0.8710</td>
<td>0.9569</td>
<td>0.9327</td>
<td>0.9327</td>
<td>0.9327</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>0.9198</td>
<td>0.9291</td>
<td>0.7979</td>
<td>0.7798</td>
<td>0.8566</td>
<td>0.9308</td>
<td>0.9630</td>
<td>0.8820</td>
<td>0.9601</td>
<td>0.9340</td>
<td>0.9340</td>
<td>0.9340</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rank</td>
<td>0.9554</td>
<td>0.9465</td>
<td>0.8153</td>
<td>0.8151</td>
<td>0.8831</td>
<td>0.9381</td>
<td>0.9668</td>
<td>0.8779</td>
<td>0.9643</td>
<td>0.9368</td>
<td>0.9368</td>
<td>0.9368</td>
</tr>
</tbody>
</table>
Scatter plots of the observed-estimated performance

- On the 32 possibles cases (Mod. × Feat. × Algo.), the best set of features: all 27 times, lon 12 times, basic 6 times.

- With linear model: basic set is never the one of the best set, lon features are more linearity correlated with perf.

- Random forest model obtains higher regression quality: basic can be one of the best set (2 times), Nevertheless, 7/8 cases, all features are the best one.

\[ \text{basic, } R^2 = 0.9327 \quad \text{lon, } R^2 = 0.9601 \quad \text{all, } R^2 = 0.9643 \]
Portfolio scenario

- Portfolio of 4 metaheuristics: TS, SA, EA, ILS
- Classification task: selection of one of the best metaheuristic
- Models: logit, random forest, svm
- Quality of classification: error rate (algo. is not one of the best) on cross-validation.

<table>
<thead>
<tr>
<th>Probl.</th>
<th>Feat.</th>
<th>logit</th>
<th>rf</th>
<th>svm</th>
<th>cst</th>
<th>rnd</th>
</tr>
</thead>
<tbody>
<tr>
<td>NK</td>
<td>basic</td>
<td>0.0379</td>
<td>0.0278</td>
<td>0.0158</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>lon</td>
<td><strong>0.0203</strong></td>
<td><strong>0.0249</strong></td>
<td><strong>0.0168</strong></td>
<td>0.4711</td>
<td>0.6749</td>
</tr>
<tr>
<td></td>
<td>all</td>
<td>0.0244</td>
<td><strong>0.0269</strong></td>
<td><strong>0.0165</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>QAP</td>
<td>basic</td>
<td><strong>0.0142</strong></td>
<td>0.0107</td>
<td>0.0771</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>lon</td>
<td><strong>0.0156</strong></td>
<td><strong>0.0086</strong></td>
<td><strong>0.0456</strong></td>
<td>0.4222</td>
<td>0.6706</td>
</tr>
<tr>
<td></td>
<td>all</td>
<td><strong>0.0161</strong></td>
<td>0.0106</td>
<td><strong>0.0431</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Conclusions and perspectives

- Structure of the local optima network can explain problem difficulty
- Features of LON can be used for performance prediction
- The sampling methodology gives relevant estimation of LON features for performance prediction and portfolio design

Perspectives

- Reduce the cost and improve the efficiency of the sampling
- Test on others (real world black-box) problems with others metaheuristics
- Understand the link between problem definition and structure of LON
- Study LON as a landscape at large scale
References I

A-L. Barabási and R. Albert.
Emergence of scaling in random networks.

J. P. K. Doye.
The network topology of a potential energy landscape: a static scale-free network.

Fabio Daolio, Marco Tomassini, Sébastien Verel, and Gabriela Ochoa.
Communities of Minima in Local Optima Networks of Combinatorial Spaces.
References II

Fabio Daolio, Sébastien Verel, Gabriela Ochoa, and Marco Tomassini.
Local optima networks and the performance of iterated local search.
In *Proceedings of the fourteenth international conference on Genetic and evolutionary computation conference*, pages 369–376, Philadelphia, United States, July 2012. ACM.

Fabio Daolio, Sébastien Verel, Gabriela Ochoa, and Marco Tomassini.
Local Optima Networks of the Permutation Flow-Shop Problem.
References III


References IV


tea team.