Problem Understanding through Landscape Theory

Francisco Chicano, Gabriel Luque and Enrique Alba
Landscape Definition

- A landscape is a triple \((X, N, f)\) where
  - \(X\) is the solution space
  - \(N\) is the neighbourhood operator
  - \(f\) is the objective function

- The neighbourhood operator is a function
  \(N: X \to \mathcal{P}(X)\)

- Solution \(y\) is neighbour of \(x\) if \(y \in N(x)\)

- Regular and symmetric neighbourhoods
  - \(d = |N(x)| \quad \forall \ x \in X\)
  - \(y \in N(x) \iff x \in N(y)\)

- Objective function
  \(f: X \to \mathbb{R}\) (or \(\mathbb{N}, \mathbb{Z}, \mathbb{Q}\))
Elementary Landscapes: Formal Definition

- An elementary function is an eigenvector of the graph Laplacian (plus constant)

\[
A_{xy} = \begin{cases} 
1 & \text{if } y \in N(x) \\
0 & \text{otherwise}
\end{cases}
\]

\[
D_{xy} = \begin{cases} 
|N(x)| & \text{if } x = y \\
0 & \text{otherwise}
\end{cases}
\]

**Adjacency matrix**

**Degree matrix**

- Graph Laplacian:

\[
\Delta = A - D
\]

Depends on the configuration space

- Elementary function: eigenvector of \(\Delta\) (plus constant)

\[
(-\Delta) \times (\vec{f} - \vec{b}) = \lambda \cdot (\vec{f} - \vec{b})
\]

Eigenvalue
An elementary landscape is a landscape for which

$$\text{avg}\left\{ f(y) \right\} = \alpha f(x) + \beta \quad \forall x \in X$$

where

$$\text{avg}\left\{ f(y) \right\} \overset{\text{def}}{=} \frac{1}{d} \sum_{y \in N(x)} f(y)$$

Grover's wave equation

$$\text{avg}\left\{ f(y) \right\} = f(x) + \frac{\lambda}{d} \left( f - f(x) \right)$$

Eigenvalue

$$\alpha = 1 - \frac{\lambda}{d} \quad \beta = \frac{\lambda}{d} f$$
## Elementary Landscapes: Examples

<table>
<thead>
<tr>
<th>Problem</th>
<th>Neighbourhood</th>
<th>( d )</th>
<th>( k )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Symmetric TSP</td>
<td>2-opt</td>
<td>( n(n-3)/2 )</td>
<td>( n-1 )</td>
</tr>
<tr>
<td></td>
<td>swap two cities</td>
<td>( n(n-1)/2 )</td>
<td>( 2(n-1) )</td>
</tr>
<tr>
<td>Antisymmetric TSP</td>
<td>inversions</td>
<td>( n(n-1)/2 )</td>
<td>( n(n+1)/2 )</td>
</tr>
<tr>
<td></td>
<td>swap two cities</td>
<td>( n(n-1)/2 )</td>
<td>( 2n )</td>
</tr>
<tr>
<td>Graph ( \alpha )-Coloring</td>
<td>recolor 1 vertex</td>
<td>((\alpha-1)n)</td>
<td>( 2\alpha )</td>
</tr>
<tr>
<td>Graph Matching</td>
<td>swap two elements</td>
<td>( n(n-1)/2 )</td>
<td>( 2(n-1) )</td>
</tr>
<tr>
<td>Graph Bipartitioning</td>
<td>Johnson graph</td>
<td>( n^2/4 )</td>
<td>( 2(n-1) )</td>
</tr>
<tr>
<td>NAES</td>
<td>bit-flip</td>
<td>( n )</td>
<td>( 4 )</td>
</tr>
<tr>
<td>Max Cut</td>
<td>bit-flip</td>
<td>( n )</td>
<td>( 4 )</td>
</tr>
<tr>
<td>Weight Partition</td>
<td>bit-flip</td>
<td>( n )</td>
<td>( 4 )</td>
</tr>
</tbody>
</table>
Landscape Theory

• What if the landscape is not elementary?
• Any landscape can be written as the sum of elementary landscapes

Landscape Decomposition

• There exists a set of elementary functions that form a basis of the function space (Fourier basis)

Elementary functions (from the Fourier basis)

$e_1$

$e_2$

Non-elementary function

Elementary components of $f$

$< e_1, f >$

$< e_2, f >$

$< e_2, f >$

$< e_1, f >$
## Landscape Decomposition: Examples

<table>
<thead>
<tr>
<th>Problem</th>
<th>Neighbourhood</th>
<th>$d$</th>
<th>Components</th>
</tr>
</thead>
<tbody>
<tr>
<td>General TSP</td>
<td>inversions</td>
<td>$n(n-1)/2$</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>swap two cities</td>
<td>$n(n-1)/2$</td>
<td>2</td>
</tr>
<tr>
<td>Subset Sum Problem</td>
<td>bit-flip</td>
<td>$n$</td>
<td>2</td>
</tr>
<tr>
<td>MAX k-SAT</td>
<td>bit-flip</td>
<td>$n$</td>
<td>$k$</td>
</tr>
<tr>
<td>NK-landscapes</td>
<td>bit-flip</td>
<td>$n$</td>
<td>$k+1$</td>
</tr>
<tr>
<td>Radio Network Design</td>
<td>bit-flip</td>
<td>$n$</td>
<td>max. nb. of reachable antennae</td>
</tr>
<tr>
<td>Frequency Assignment</td>
<td>change 1 frequency</td>
<td>$(\alpha-1)n$</td>
<td>2</td>
</tr>
<tr>
<td>QAP</td>
<td>swap two elements</td>
<td>$n(n-1)/2$</td>
<td>3</td>
</tr>
</tbody>
</table>
Autocorrelation

• Let \( \{x_0, x_1, \ldots \} \) be a simple random walk on the configuration space where \( x_{i+1} \in N(x_i) \).

• The random walk induces a time series \( \{f(x_0), f(x_1), \ldots \} \) on a landscape.

• The autocorrelation function is defined as:

\[
    r(s) = \frac{\langle f(x_t) f(x_{t+s}) \rangle_t - \langle f \rangle^2}{\langle f^2 \rangle - \langle f \rangle^2}
\]

• The autocorrelation length and coefficient:

\[
    l = \sum_{s=0}^{\infty} r(s) \quad \xi = \frac{1}{1 - r(1)}
\]

• Autocorrelation length conjecture:

The number of local optima in a search space is roughly

\[
    M \approx |X|/|X(x_0, l)|
\]
Autocorrelation Length “Conjecture”

- The higher the value of $l$ and $\xi$ the smaller the number of local optima.
- $l$ and $\xi$ is a measure of ruggedness.

<table>
<thead>
<tr>
<th>Ruggedness</th>
<th>SA (configuration 1)</th>
<th>SA (configuration 2)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>% rel. error</td>
<td>nb. of steps</td>
</tr>
<tr>
<td>$10 \leq \zeta &lt; 20$</td>
<td>0.2</td>
<td>50,500</td>
</tr>
<tr>
<td>$20 \leq \zeta &lt; 30$</td>
<td>0.3</td>
<td>53,300</td>
</tr>
<tr>
<td>$30 \leq \zeta &lt; 40$</td>
<td>0.3</td>
<td>58,700</td>
</tr>
<tr>
<td>$40 \leq \zeta &lt; 50$</td>
<td>0.5</td>
<td>62,700</td>
</tr>
<tr>
<td>$50 \leq \zeta &lt; 60$</td>
<td>0.7</td>
<td>66,100</td>
</tr>
<tr>
<td>$60 \leq \zeta &lt; 70$</td>
<td>1.0</td>
<td>75,300</td>
</tr>
<tr>
<td>$70 \leq \zeta &lt; 80$</td>
<td>1.3</td>
<td>76,800</td>
</tr>
<tr>
<td>$80 \leq \zeta &lt; 90$</td>
<td>1.9</td>
<td>79,700</td>
</tr>
<tr>
<td>$90 \leq \zeta &lt; 100$</td>
<td>2.0</td>
<td>82,400</td>
</tr>
</tbody>
</table>

Angel, Zissimopoulos. Theoretical Computer Science 263:159-172 (2001)
Fitness-Distance Correlation: Definition

$$r = \frac{\text{Cov}(f, d)}{\sigma_f \sigma_d}$$

Difficult when $r < 0.15$ (Jones & Forrest)
Fitness-Distance Correlation Formulas

- Using the previous facts we get for the elementary landscapes...

\[
 r = \frac{-f_{[j]}(x^*)}{\sigma_f \sqrt{n}}
\]

If \( j = 1 \)

\[
 r = 0
\]

If \( j > 1 \)

- In general, for an arbitrary function... ... the only component contributing to \( r \) is \( f_{[1]}(x) \)

\[
 f(x) = f_{[0]}(x) + f_{[1]}(x) + f_{[2]}(x) + \ldots + f_{[n]}(x)
\]

\[
 r = \frac{-f_{[1]}(x^*)}{\sigma_f \sqrt{n}}
\]

Rugged components are not considered by FDC

---

EvoCOP 2012, LNCS 7245: 111-123
FDC: Implications for Linear Functions

- **Fitness-Distance Correlation** for order-1 (linear) elementary landscapes (assume max.)

\[ f(x) = \sum_{i=1}^{n} a_i x_i + b \]

\[ r = \frac{- \sum_{i=1}^{n} |a_i|}{\sqrt{n \sum_{i=1}^{n} a_i^2}} \]

\[-1 \leq r < 0\]

- We can define a linear elementary landscape with the desired FDC \( \rho \) (lower than 0)

\[ n > 1 / \rho^2 \]

\[ a_1 = \frac{(n - 1) + n|\rho|\sqrt{(1 - \rho^2)(n - 1)}}{n\rho^2 - 1} \]

\[ a_2 = a_3 = \ldots = a_n = 1 \]

- “Difficult” problems can be obtained starting in \( n=45 \) (\( |r| < 0.15 \))

\[
\begin{align*}
a_1 &= 7061.43 \\
a_2 &= a_3 = \ldots = a_{45} = 1
\end{align*}
\]
Expectation for Bit-flip Mutation

- Example (j=0,1,2,3):

\[ f = f_0 + f_1 + f_2 + f_3 \]

\[
E\{ f(M_p(x)) \} = \sum_{j=0}^{n} (1 - 2p)^j f_{[j]}(x)
\]

\( p \approx 0.23 \rightarrow \text{maximum expectation} \)

\( p = 1/2 \rightarrow \text{start from scratch} \)

The traditional \( p = 1/n \) could be around here
Expectation for the Uniform Crossover

\[ \mathbb{E}\{f(U_\rho(x, y))\} = \sum_{r=0}^{n} A_{x, y}^{(r)} (1 - 2\rho)^r \]

where

\[ A_{x, y}^{(r)} = \sum_{w \in \mathbb{B}^n, |(x \oplus y) \land w| = r} a_w \psi_w(y) \]

and

\[ \mathbb{E}\{f(U_{1/2}(x, y))\} = A_{x, y}^{(0)} = \sum_{w \in \mathbb{B}^n, |(x \oplus y) \land w| = 0} a_w \psi_w(y) \]
Only Expected Values?

• Can we compute the probability distribution?

\[
\pi(f(M_p(x))) = \left(V^T\right)^{-1} F(x) \Lambda(p)
\]

Polynomials in \( p \)

Current solution

fitness

probability

expectation

expectation
Runtime of (1+1) EA: Expected Hitting Time

- The expected hitting time is a fraction of polynomials in $p$

\[
E\{\tau\} = \begin{cases} 
\frac{1}{2p} & \text{for } n = 1, \\
\frac{7 - 5p}{4(p - 2)(p - 1)p} & \text{for } n = 2, \\
\frac{26p^4 - 115p^3 + 202p^2 - 163p + 56}{8(p - 1)^2p(p^2 - 3p + 3)(2p^2 - 3p + 2)} & \text{for } n = 3.
\end{cases}
\]

- Optimal probability of mutation for $n=2$

\[
p^*_2 = \frac{1}{5} \left(6 - \sqrt[3]{\frac{2}{23 - 5\sqrt{21}}} - \sqrt[3]{\frac{23 - 5\sqrt{21}}{2}}\right) \approx 0.561215,
\]
We can observe how the expressions grow very fast as \( n \) increases. The factor \( p(p-1) \) is always present in the denominator for \( n > 2 \), which means that when \( p \) takes extreme values, \( p = 0 \) or \( p = 1 \), it is not possible to reach the global optimum from any solution, since the algorithm will keep the same solution if \( p = 0 \) or will alternate between two solutions if \( p = 1 \). However, when \( n = 1 \) the probability \( p = 1 \) is valid, furthermore, is optimal, because if the global solution is not present at the beginning we can reach it by alternating the only bit we have. In Figure 1 we show the expected runtime as a function of the probability of flipping a bit for \( n = 1 \) to 7.

We can observe how the optimal probability (the one obtaining the minimum expected runtime) decreases as \( n \) increases.

Figure 1: Expected runtime of the \((1+1)\) EA for OneMax as a function of the probability of flipping a bit. Each line corresponds to a different value of \( n \) from 1 to 7.

Having the exact expressions we can compute the optimal mutation probability for each \( n \) by using classical optimization methods in one variable. In particular, for \( n = 1 \) the optimal value is \( p = 1 \) as we previously saw and for \( n = 2 \) we have to solve a cubic polynomial in order to obtain the exact expression. The result is:

\[
p^*_2 = \frac{1}{5} \left( \frac{1}{2} \right)^{\frac{1}{3}} - \frac{1}{5} \left( \frac{1}{2} \right)^{\frac{1}{3}}.
\]

which is slightly higher than the recommended value \( p = 1/n \). As we increase \( n \), analytical responses for the optimal probability are not possible and we have to apply numerical methods. In our case we used the Newton method in order to find a root of the equation \( d\mathbb{E}\{\tau\} / dp = 0 \). The results up to \( n = 100 \) can be found in Table 1. A fast observation of the results reveals that the optimal probability is always a little bit higher than the recommended \( p = 1/n \).

From previous work we know that the optimal probability is in the form \( c/n \) for a 2.
### Runtime of (1+1) EA: Optimal Probabilities

<table>
<thead>
<tr>
<th>$n$</th>
<th>$p^*_n$</th>
<th>$E{\tau}$</th>
<th>$n$</th>
<th>$p^*_n$</th>
<th>$E{\tau}$</th>
<th>$n$</th>
<th>$p^*_n$</th>
<th>$E{\tau}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.00000</td>
<td>0.500</td>
<td>35</td>
<td>0.03453</td>
<td>273.018</td>
<td>68</td>
<td>0.01741</td>
<td>648.972</td>
</tr>
<tr>
<td>2</td>
<td>0.56122</td>
<td>2.959</td>
<td>36</td>
<td>0.03354</td>
<td>283.448</td>
<td>69</td>
<td>0.01715</td>
<td>661.189</td>
</tr>
<tr>
<td>3</td>
<td>0.38585</td>
<td>6.488</td>
<td>37</td>
<td>0.03261</td>
<td>293.953</td>
<td>70</td>
<td>0.01690</td>
<td>673.445</td>
</tr>
<tr>
<td>4</td>
<td>0.29700</td>
<td>10.808</td>
<td>38</td>
<td>0.03172</td>
<td>304.531</td>
<td>71</td>
<td>0.01665</td>
<td>685.740</td>
</tr>
<tr>
<td>5</td>
<td>0.24147</td>
<td>15.758</td>
<td>39</td>
<td>0.03088</td>
<td>315.181</td>
<td>72</td>
<td>0.01642</td>
<td>698.073</td>
</tr>
<tr>
<td>6</td>
<td>0.20323</td>
<td>21.222</td>
<td>40</td>
<td>0.03009</td>
<td>325.900</td>
<td>73</td>
<td>0.01618</td>
<td>710.444</td>
</tr>
<tr>
<td>7</td>
<td>0.17526</td>
<td>27.120</td>
<td>41</td>
<td>0.02933</td>
<td>336.688</td>
<td>74</td>
<td>0.01596</td>
<td>722.852</td>
</tr>
<tr>
<td>8</td>
<td>0.15391</td>
<td>33.391</td>
<td>42</td>
<td>0.02861</td>
<td>347.541</td>
<td>75</td>
<td>0.01574</td>
<td>735.298</td>
</tr>
<tr>
<td>9</td>
<td>0.13710</td>
<td>39.990</td>
<td>43</td>
<td>0.02793</td>
<td>358.459</td>
<td>76</td>
<td>0.01553</td>
<td>747.779</td>
</tr>
<tr>
<td>10</td>
<td>0.12352</td>
<td>46.882</td>
<td>44</td>
<td>0.02727</td>
<td>369.441</td>
<td>77</td>
<td>0.01532</td>
<td>760.297</td>
</tr>
<tr>
<td>11</td>
<td>0.11233</td>
<td>54.039</td>
<td>45</td>
<td>0.02665</td>
<td>380.484</td>
<td>78</td>
<td>0.01512</td>
<td>772.849</td>
</tr>
<tr>
<td>12</td>
<td>0.10295</td>
<td>61.437</td>
<td>46</td>
<td>0.02605</td>
<td>391.587</td>
<td>79</td>
<td>0.01492</td>
<td>785.437</td>
</tr>
<tr>
<td>13</td>
<td>0.09499</td>
<td>69.057</td>
<td>47</td>
<td>0.02548</td>
<td>402.750</td>
<td>80</td>
<td>0.01473</td>
<td>798.059</td>
</tr>
<tr>
<td>14</td>
<td>0.08815</td>
<td>76.882</td>
<td>48</td>
<td>0.02493</td>
<td>413.970</td>
<td>81</td>
<td>0.01454</td>
<td>810.715</td>
</tr>
<tr>
<td>15</td>
<td>0.08220</td>
<td>84.898</td>
<td>49</td>
<td>0.02441</td>
<td>425.247</td>
<td>82</td>
<td>0.01436</td>
<td>823.405</td>
</tr>
<tr>
<td>16</td>
<td>0.07699</td>
<td>93.092</td>
<td>50</td>
<td>0.02391</td>
<td>436.580</td>
<td>83</td>
<td>0.01418</td>
<td>836.128</td>
</tr>
<tr>
<td>17</td>
<td>0.07239</td>
<td>101.454</td>
<td>51</td>
<td>0.02342</td>
<td>447.967</td>
<td>84</td>
<td>0.01400</td>
<td>848.884</td>
</tr>
<tr>
<td>18</td>
<td>0.06830</td>
<td>109.974</td>
<td>52</td>
<td>0.02296</td>
<td>459.407</td>
<td>85</td>
<td>0.01384</td>
<td>861.673</td>
</tr>
<tr>
<td>19</td>
<td>0.06463</td>
<td>118.642</td>
<td>53</td>
<td>0.02251</td>
<td>470.900</td>
<td>86</td>
<td>0.01367</td>
<td>874.493</td>
</tr>
<tr>
<td>20</td>
<td>0.06133</td>
<td>127.453</td>
<td>54</td>
<td>0.02208</td>
<td>482.444</td>
<td>87</td>
<td>0.01351</td>
<td>887.345</td>
</tr>
<tr>
<td>21</td>
<td>0.05835</td>
<td>136.398</td>
<td>55</td>
<td>0.02167</td>
<td>494.038</td>
<td>88</td>
<td>0.01335</td>
<td>900.229</td>
</tr>
<tr>
<td>22</td>
<td>0.05563</td>
<td>145.471</td>
<td>56</td>
<td>0.02127</td>
<td>505.682</td>
<td>89</td>
<td>0.01320</td>
<td>913.143</td>
</tr>
<tr>
<td>23</td>
<td>0.05316</td>
<td>154.667</td>
<td>57</td>
<td>0.02088</td>
<td>517.374</td>
<td>90</td>
<td>0.01304</td>
<td>926.088</td>
</tr>
</tbody>
</table>
Landscape Explorer

- **Main goals:**
  - Easy to extend
  - Multiplatform

- **RCP architecture**

RCP (Rich Client Platform)

Eclipse RCP specific plugins

- Workbench
- JFace
- SWT
- Runtime / Equinox (OSGi)
Rich Client Platform

• **Mechanisms for plugin interaction:**

• **Extensions, extension points and package exports**
• **Main plugins of the application**

![Diagram showing the main plugins of the application](image-url)

neo.landscapes.theory.tool

neo.landscapes.theory.tool.procedures

neo.landscapes.theory.tool.selectors

neo.landscapes.theory.kernel

TSP QAP SS UQO DFA FAP WF

Landscape Theory  Relevant Results  Software Tools  Conclusions & Future Work

RCP  Architecture  Proc. & Lands.  URLs

**Architecture**

Define un punto de extensión favorece la extensión del software. La descripción de las extensiones incluyen puntos de extensión, como definir los puntos de extensión de la aplicación así como de definir los responsables de mostrar la interfaz gráfica de usuario.

Ofrece varios ejemplos de ellos. (desde el punto de vista de la implementación) y interfaces y clases que definen lo que es un plug-in. Los puntos de extensión el comprobar que hay otros productos que deseen consultarla. Sin embargo, incluso este conjunto mínimo podría cambiar con el paso del tiempo y la modularización de los productos.
Included Procedures and Landscapes

• Procedures
  • Elementary Landscape Check
  • Mathematica program to get the Elementary Landscape Decomposition
  • Computation of the reduced adjacency matrix
  • Theoretical Autocorrelation Measures
  • Experimental Autocorrelation
  • Estimation of the number of elementary components

• Landscapes
  • QAP (Quadratic Assignment Problem)
  • UQO (Unconstrained Quadratic Optimization)
  • TSP (Traveling Salesman Problem)
  • Walsh Functions (linear combinations)
  • Subset Sum Problem
Available on the Internet

http://neo.lcc.uma.es/software/landexplorer

Introduction to Landscape Explorer

Landscape Explorer is an open-source and extensible software platform for the support of the research in the domain of Landscape Theory. It is based on the eclipse platform and it is composed of plug-ins which combine together to provide a large number of algorithms for analyzing the landscape of optimization.
On-line Computation of Autocorrelation for QAP

http://neo.lcc.uma.es/software/qap.php

Autocorrelation of the Quadratic Assignment Problem

Select the instance to analyze

From QAPLIB
bur26a.dat

Upload your own instance*

*The format of the uploaded file must be the same as the one of the QAPLIB instances.
Conclusions & Future Work

Conclusions

• Landscape Theory is very good for providing statistical information at a low cost
• FDC, expected fitness value after mutation and uniform crossover
• Runtime?
• Software Tools have been developed to help non-experts to use the knowledge

Future Work

\[ R(A, f) = \Phi(f) \otimes \Lambda(A) \]

\textit{problem} \hspace{1cm} \textit{algorithm}
Thanks for your attention !!!