Visualising the Search Landscape of the Triangle Program

EuroGP 2017, Mauro Castelli et al., LNCS 10196, pages 96-113, Amsterdam

W. B. Langdon
Department of Computer Science

Humies
$10,000 Human-Competitive Results
Visualising the Search Landscape of the Triangle Program

W. B. Langdon
Department of Computer Science

Humies
$10,000 Human-Competitive Results
Fitness Landscape of the Triangle Program

- **Background, what is Genetic Improvement**
  - Fitness landscapes of genetic improvement
  - the Triangle program.
  - Constructing Triangle’s fitness landscape
    - Two versions: binary and all comparison changes

- **Results**
  - searchspaces
  - schema analysis
  - Hill climbing and local optima visualisation

- **Insight into Genetic Improvement?**
Genetic Improvement

• Genetic Improvement is the application of search (often genetic programming) to improve existing software, e.g.
  – Fix bugs
  – Faster (CPU or on parallel hardware: GPU)
  – Less energy used
  – Less memory

• Real programs (10⁴ to 10⁶ of lines of code)

W. B. Langdon, UCL
89% mutations which compile make no change to test case **CS-DC'15**
GI Fitness Landscapes

• Real software is resilient to mutations.
• Schema (crossover) analysis.
• The Triangle program is a software engineering benchmark
Triangle Program

- Given length of three sides what type is triangle? (Software Engineering benchmark)
- Test suite (14) covers all paths
- Mutate conditionals
- Fitness is number of tests that fail (minimize)
- Code and datasets online
  http://www.cs.ucl.ac.uk/staff/W.Langdon/egp2017/triangle/
- whole landscape
int gettri(int side1, int side2, int side3)
{
    int triang;
    if( side1 <= 0 || side2 <= 0 || side3 <= 0){
        return 4;
    }
    triang = 0;
    if(side1 == side2){
        triang = triang + 1;
    }
    if(side1 == side3){
        triang = triang + 2;
    }
    if(side2 == side3){
        triang = triang + 3;
    }
    if(triang == 0){
        if(side1 + side2 <= side3 ||
            side2 + side3 <= side1 ||
            side1 + side3 <= side2){
            return 4;
        } else {
            return 1;
        }
    }
    if(triang > 3){
        return 3;
    } else if ( triang == 1 && side1 + side2 > side3) {
        return 2;
    } else if (triang == 2 && side1 + side3 > side2){
        return 2;
    } else if (triang == 3 && side2 + side3 > side1){
        return 2;
    }
    return 4;
}
## testcases_oracle.txt  14 tests

<table>
<thead>
<tr>
<th>Three inputs</th>
<th>expected output</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 0</td>
<td>4</td>
</tr>
<tr>
<td>1 0 0</td>
<td>4</td>
</tr>
<tr>
<td>1 1 0</td>
<td>4</td>
</tr>
<tr>
<td>1 1 1</td>
<td>3</td>
</tr>
<tr>
<td>2 2 1</td>
<td>2</td>
</tr>
<tr>
<td>1 1 2</td>
<td>4</td>
</tr>
<tr>
<td>2 1 2</td>
<td>2</td>
</tr>
<tr>
<td>1 2 1</td>
<td>4</td>
</tr>
<tr>
<td>2 1 1</td>
<td>4</td>
</tr>
<tr>
<td>3 2 2</td>
<td>2</td>
</tr>
<tr>
<td>3 2 1</td>
<td>4</td>
</tr>
<tr>
<td>4 3 2</td>
<td>1</td>
</tr>
<tr>
<td>2 3 1</td>
<td>4</td>
</tr>
<tr>
<td>2 1 3</td>
<td>4</td>
</tr>
</tbody>
</table>

Inputs are the three sides of the triangle. Output is correct classification of the triangle. Test suite covers all paths but is not strong enough to detect all mutations. **Dataset** gives whole test equivalent fitness landscape for 2-way comparisons.

W. B. Langdon, UCL
Triangle Fitness Landscape

- Only mutate C comparison operators (17)
- Run all tests (14). For how many does new code give the wrong answer? (0-14)
Triangle Program search space

- Fitness distribution of all 16,926,659,444,735 mutations
- 78% of mutants fail five tests (mean 5.42, sd 1.05)
- 9215 global optima
- fitness distance correlation -0.070837
Genetic Improvement Schema

Binary schema

**011*0001*0001** * don’t care

Triangle Program 17 locations

***== == == **** > ** == * == *

Example 1st order triangle schema

>= * ***** ***** *****

i.e. replace if( side1 <= 0 with if( side1 >= 0

Deceptive schema has no solutions but its mean fitness is better than competing schema which contains solutions, vice-versa
102 First Order Schema

- $17 \times 6 = 102$ schema. 56 deceptive (×)
- 3 “best” get five tests wrong. This makes them better than average but contain no solutions
- (2-way, two alternatives only, schema are not deceptive)
Deception in 6-way Schema

- 1st order cover the search space, i.e. far from start point. (two alternatives schema are closer to start point.)
- Strong epistasis so hard for naive GA and Hill Climbing.
- 1st order only have one defined position, often low signal to noise. $\sigma \approx 1$ so $\Delta/\sigma \ll 0.01$
- 20 schema where signal/noise $\Delta/\sigma > 0.1$, 18 are deceptive
Summary 6-way 1\textsuperscript{st} order Schema

- 17×6=102 schema. 56 deceptive(×)
- None with strong signal towards any solution
- Strong epistasis, low fitness distance correlation
- Hard for global search GA and Hill Climbing but GI starts search near solution
- 1\textsuperscript{st} order only have one defined position, often low signal to noise. $\sigma \approx 1$ so $\Delta/\sigma << 0.01$
- 20 schema where signal/noise $\Delta/\sigma > 0.1$, 18 are deceptive
- 2-way case not deceptive. (Closer to solution?)
Hill Climbing can always Improve

- Start from **7 mutations** or below, there probably is a route to **solution**.
- Start between **7 and 14** route to fail just one test
- Start between **11-17** fail at most **two tests**
Iterated Local Search

87% of Iterated Local Search (ILS) find a program which passes all the tests.
Local Optima

• A solution is a local optimum if none of its neighbours have a better fitness value.
• In Triangle Program each mutant has $5 \times 17 = 85$ neighbours.
• E.g. a mutants which fails 5 tests is a local optima if all of its 85 neighbours fail 5 or more tests.
Local Optima

1000 Iterated Local Searches (ILS) find 2,372,805 unique local optima

"Big valley" local optima leading towards global solution

Sunflower plot: number of petals gives density of local optima

Best, fails 0 tests
Path of 100 Iterated Local Searches

- First 1000 ILS iterations
- Only the edges shown.
- Black if fitness improves.
- Edges are coloured no fitness change.
Conclusions

- Whole fitness landscape $16,926,659,444,735$
  - 9215 pass all tests. Neutral moves & plateaus
- Global schema analysis
  - Low signal to noise, deceptive, epistasis.
  - 2-way closer to solution, no deception.
- “big valley” with many local optima, particularly close to global optima
- Hill climbing can always improve
- Iterated Local Search often finds solution.
- Start near human code

W. B. Langdon, UCL
In Lewis Carroll's “Through the Looking-Glass…” the White Queen says she could believe "six impossible things before breakfast”.

Do something impossible
Evolve something never attempted before
Humies
$10000
Human-Competitive Results

END

http://www.cs.ucl.ac.uk/staff/W.Langdon/
http://www.epsrc.ac.uk/
Genetic Improvement

W. B. Langdon
CREST
Department of Computer Science
Discussion: Lessons for GI?

• With simple scalar fitness, almost everyone fails 5 tests.
• Need more fitness levels?
• Is “fitness” function too simple?
• Would a novelty reward help?
• Would multi-objective approach help?
Discussion: Lessons for GI?

• Global search dominated by mutants which fail five of fourteen tests.
• 1st order, schema says global search hard
  – Epistasis, deception.
• ILS more practical than simple hillclimbing (HC)
• ILS shows big valley of local optima near global optima (human code)
• Even simple HC may work close to solution
• START NEAR SOLUTION
Insight into Genetic Improvement?

- Triangle program is software engineering benchmark
  - small but akin to unit testing

- **Whole** fitness landscape
  - Much of search space passes most tests
  - Very few pass all tests
  - Large areas where neighbours pass 5 tests. **Neutral networks, plateaus.**

- **Global** schema analysis.
  - Low signal to noise, **deceptive**, epistasis
  - But (2-way mutation) closer to solution no deception.

- Many local optima, particularly close to global optima.

- Hill climbing can **always improve** (if allow neutral moves) but if start far from solution HC sometimes gets stuck.

- Iterated Local Search **often finds solution.**

- For global and hill climbing search **starting near human code** helps

[W.Langdon egp2017_triangle.tar.gz](W.Langdon%20egp2017%20triangle.tar.gz)
int gettri(int side1, int side2, int side3)  
{
    int triang;
    if (side1 <= 0 || side2 <= 0 || side3 <= 0){
        return 4;
    }
    triang = 0;
    if(side1 == side2){
        triang = triang + 1;
    }
    if(side1 == side3){
        triang = triang + 2;
    }
    if(side2 == side3){
        triang = triang + 3;
    }
    if(triang == 0){
        if(side1 + side2 <= side3 || side2 + side3 <= side1 || side1 + side3 <= side2){
            return 4;
        } else {
            return 1;
        }
    }
    if(triang > 3){
        return 3;
    } else if (triang == 1 && side1 + side2 > side3) {
        return 2;
    } else if (triang == 2 && side1 + side3 > side2){
        return 2;
    } else if (triang == 3 && side2 + side3 > side1){
        return 2;
    }
    return 4;
}
## First Order Schema

<table>
<thead>
<tr>
<th>Schema id</th>
<th>mean</th>
<th>sd</th>
<th>pop size</th>
</tr>
</thead>
<tbody>
<tr>
<td>-4</td>
<td>3.719 ±1.328</td>
<td>1.9</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>4.969 ±1.075</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-5</td>
<td>4.062 ±1.478</td>
<td>4.7</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>4.625 ±1.166</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-6</td>
<td>3.812 ±1.509</td>
<td>2.4</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>4.875 ±0.927</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-11</td>
<td>3.438 ±1.273</td>
<td>1.1</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>5.250 ±0.661</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-14</td>
<td>4.312 ±1.424</td>
<td>43.5</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>4.375 ±1.293</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-16</td>
<td>4.188 ±1.550</td>
<td>8.6</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>4.500 ±1.118</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2: Mean and standard deviation of number of tests failed for first order schema (excluding 22 with average means).
Fitness in bit order \((2^8 \times 2^9)\)

- 2048 global optima in white.
- Regular patterns indicate small building blocks.
- Vertical strips 8 pixels wide says first three bits do not impact fitness.
- Last but one bit gives four horizontal stripes:
  - two contain 50176 mutants fail \(\geq 4\) tests (dark)
  - others hold all the solutions (white)
Fitness in bit order \((2^8 \times 2^9)\)

- Vertical strips 8 pixels wide says first three bits do not impact fitness.
- Last but one bit gives four horizontal stripes:
  - two contain 50176 mutants fail \(\geq 4\) tests (dark)
  - others hold all the solutions (white)
Genetic Improvement evolves code patches

W. B. Langdon, UCL
Recent Successes of Genetic Improvement

• Automatic bug repair
  – GenProg, e.g. 105 bugs fix most (multiple best papers, IFIP TC2 Manfred Paul Award, 2 Humies)

• Better programs
  – 70x Bowtie2, BarraCUDA, pknots 10000x
  – Less energy, less memory
  – MOGA speed v. quality, e.g. [SIGGRAPH]

• Code transplant [Marginean, e.g. best paper ISSTA 2015]
  – E.g. C++, code indent, call graph layout into Kate editor (we can evolve an editor) Humie
Genetic Improvement Benchmarks

• Bugs to be fixed

• Software Engineering
  Many, e.g. SIR [http://sir.unl.edu](http://sir.unl.edu)

• Fitness landscape
  Mutation testing/GA fitness landscape for the Triangle Program, UCL CS [RN/16/05](http://www.cs.ucl.ac.uk/staff/W.Langdon/ppsnn2016/triangle/)

Code and datasets online
The Genetic Programming Bibliography

http://www.cs.bham.ac.uk/~wbl/biblio/

11484 references

Make sure it has all of your papers!
E.g. email W.Langdon@cs.ucl.ac.uk or use | Add to It | web link

RSS Support available through the
Collection of CS Bibliographies.

Co-authorship community.
Downloads

A personalised list of every author’s
GP publications.

blog

Search the GP Bibliography at
http://liinwww.ira.uka.de/bibliography/Ai/genetic.programming.html