A Theoretical & Empirical Analysis of Evolutionary Testing and Hill Climbing for Structural Test Data Generation
ISSTA July 2007

Mark Harman
King’s College London

Phil McMinn
Sheffield University

Mark Harman
ISSTA: Empirical and Theoretical Search Based Testing
Thursday, 10 December 2009
Which Search Technique: Global or Local?

Mark Harman
King’s College London

Phil McMinn
Sheffield University
Which Search Technique: Global or Local?

Mark Harman and Phil McMinn.
A Theoretical and Empirical Analysis of Evolutionary Testing and Hill Climbing for Structural Test Data Generation
ISSTA 2007.

Mark Harman and Phil McMinn,
A Theoretical and Empirical Study of Search Based Testing: Local, Global and Hybrid Search
TSE. To appear.
Which Search Technique: Global or Local?

Mark Harman and Phil McMinn.  
A Theoretical and Empirical Analysis of Evolutionary Testing and Hill Climbing for Structural Test Data Generation  
ISSTA 2007.

Mark Harman and Phil McMinn,  
A Theoretical and Empirical Study of Search Based Testing: Local, Global and Hybrid Search  
TSE. To appear.
Which Search Technique: Global or Local?

Mark Harman and Phil McMinn.
A Theoretical and Empirical Analysis of Evolutionary Testing and Hill Climbing for Structural Test Data Generation
ISSTA 2007.

Mark Harman and Phil McMinn,
A Theoretical and Empirical Study of Search Based Testing: Local, Global and Hybrid Search
TSE. To appear.

Author order is alphabetical
Where is King’s College London?

Mark Harman
ISSTA: Empirical and Theoretical Search Based Testing

Thursday, 10 December 2009
Where is Sheffield University?

The British Isles

Sheffield

London

Mark Harman
ISSTA: Empirical and Theoretical Search Based Testing

Thursday, 10 December 2009
No Full Monty Joke
No Full Monty Joke

Sorry
Overview

Search Based Testing

Local: Hill Climbing using Alternating variable method

Global: Genetic Algorithms

Theoretical foundations

Schemas

Royal Roads

Empirical study

Implications
What is SBT

In Search based testing we apply search techniques to search large input spaces, guided by a fitness function.
What is SBT

In Search based testing we apply search techniques to search large input spaces, guided by a fitness function.

Genetic Algorithms, Hill climbing, Simulated Annealing, Random, Tabu Search, Estimation of Distribution Algorithms, Particle Swarm Optimization
What is SBT

In Search based testing we apply search techniques to search large input spaces, guided by a fitness function.

Genetic Algorithms, Hill climbing, Simulated Annealing, Random, Tabu Search, Estimation of Distribution Algorithms, Particle Swarm Optimization
What is SBT

In Search based testing we apply search techniques to search large input spaces, guided by a fitness function.

Genetic Algorithms, Hill climbing, Simulated Annealing, Random, Tabu Search, Estimation of Distribution Algorithms, Particle Swarm Optimization
Structural Testing

Focus on branch testing
Most widely studied
So ready for some more in depth analysis
Other Search Based Testing Applications

Temporal  Wegener et al.
Coverage  Pargass & Harrold, Xanthakis et al., McMinn, Harman, Michael et al, Sthamer, Jones …
Functional Wegener et al.
Regression Rothermel et al., Woolcott et al., Yoo and Harman,…
Interaction Cohen et al. Bryce, Colbourn
Exception Tracey and Clark
Stress Briand et al., Antoniol, Di Penta
Robustness Shultz et al.
Structural Testing

Focus on branch testing

Most widely studied

So ready for some more in depth analysis

Two algorithms:

Hill Climbing, using Korel’s alternating variable method

Genetic Algorithms, using DaimlerChrysler approach
Structural Testing

Focus on branch testing

Most widely studied
So ready for some more in depth analysis

Two algorithms:

Hill Climbing, using Korel’s alternating variable method
Genetic Algorithms, using DaimlerChrysler approach
... and Random Search
Fitness Computation

1. **Approximation level**

Identify relevant branching statements using control dependence

Evaluation of predicate in a branching condition

\[
\text{if } A = B \quad \text{Local Distance} = |A - B|
\]

Fitness = \text{Approximation Level} + \text{Local Distance}
Alternating Variable Method

The alternating variable method is hill climbing plus accelerated moves

Near Neighbour?
One small increase
One small decrease

Method:
Cycle through input variables one at a time:
probe moves move to near neighbour:
If probing works, make accelerated pattern moves
Until no improvement on any variable
Goal-Oriented Approach:

Alternating Variable Method

Fitness

Accelerated hill climb

Input variable value
Hill Climbing ↔ Steepest Descent
Hill Climbing ↔ Steepest Descent
void example(int a, int b, int c) {
    if (a == 0) {
        ...
    }
    if (b == 0) {
        if (c == 0) {
            // target
        }
    }
}

Random start: a=10  b=20  c=30

Case  a :-
    Probe move has no effect

Case  b :-
    Decrease probe improves
    So accelerate until b=0

Case  c :-
Evolutionary Algorithms

- Mutation
- Recombination
- Selection
- Insertion
- Fitness evaluation
- End?

Test execution
Evolutionary Testing

- Mutation
- Selection
- Recombination
- Fitness evaluation
- Insertion
- Test cases
- Test execution
- End?
Evolutionary Testing
How mating makes life easier
How mating makes life easier

Mating is really very much an analogy
The important property is *crossover*
How mating makes \textit{life} easier
How mating makes life easier

Mark Harman
ISSTA: Empirical and Theoretical Search Based Testing

Thursday, 10 December 2009
How mating makes *life* easier
void MrNiceGuy(double a, double b) {
    if (a == b)
    {
        // target 1
        double c = b + 1;
        if (c == 0)
        {
            // target 2
            ...
        }
    }
}
void alfred(double a, double b)
{
    if (a == b)
    {
        // target 1
        double c = b + 1;
        if (c == 0)
        {
            // target 2
        }...

Hitchcock Fitness Landscape
Hitchcock Fitness Landscape
Hitchcock Fitness Landscape
But …

When does it work
Why does it work (when it does)?
How does it compare to local search?
Schemas

\[
P_{1} = 010111001010100110010010001 \\
P_{2} = 1010010010101010111110000000 \\
P_{3} = 010100101010101000001010110 \\
P_{4} = 010111101010111110101001001 \\
... \
\]
Schemas

\[\text{Pop}[1] = 010111001010100110010010001\]
\[\text{Pop}[2] = 101001001010101011111000000000\]
\[\text{Pop}[3] = 01010010101010101000000101010110\]
\[\text{Pop}[4] = 0101111010101010111111010101001\]

...
Pop[1] = 010111001010100110010010001
Pop[2] = 101001001010101011110000000
Pop[3] = 010100101010101000001010110
Pop[4] = 01011110101010111110101001
...

Schemas

\[
\begin{align*}
\text{Pop[1]} &= 01011100101010011001001001001001 \\
\text{Pop[2]} &= 1010010010101010101111110000000 \\
\text{Pop[3]} &= 01010010101010101000000101010110 \\
\text{Pop[4]} &= 01011110101010111111010101001 \\
\end{align*}
\]

...
Schemas

\[ \overline{f}(h, K) = \frac{1}{| \{ x \mid x \in h \land x \in K \} |} \sum_{x \in h \land x \in K} f(x) \]

\[ N(h, g + 1) \geq N(h, g) \frac{\overline{f}(h, P(g))}{\frac{1}{M} \sum_{x \in P(g)} f(x)} \]

\[ N(h, g+1) \geq N(h, g) \frac{\overline{f}(h, P(g))}{\frac{1}{M} \sum_{x \in P(g)} f(x)} \left( 1 - p_e \frac{\delta(h)}{\lambda - 1} - p_m \nu(h) \right) \]
Schemas

\[ \overline{f}(h, K) = \frac{1}{| \{ x \mid x \in h \wedge x \in K \} |} \sum_{x \in h \wedge x \in K} f(x) \]

\[ N(h, g + 1) \geq N(h, g) \cdot \frac{\overline{f}(h, P(g))}{\frac{1}{M} \sum_{x \in P(g)} f(x)} \]

\[ N(h, g + 1) \geq N(h, g) \cdot \frac{\overline{f}(h, P(g))}{\frac{1}{M} \sum_{x \in P(g)} f(x)} \left( 1 - p_e \frac{\delta(h)}{\lambda - 1} - p_m o(h) \right) \]
The Royal Road
The Royal Road
The Royal Road
The Royal Road
The Royal Road
The Royal Road
The Royal Road

S1: 1111******************************
S2: ****1111******************************
S3: ********1111******************************
S4: ************1111******************************
S5: ****************1111************
S6: ****************1111************
S7: 1111111111111111************
S8: 111111111111111111111111111111111
S9: 111111111111111111111111111111111
S10: 111111111111111111111111111111111
S11: 111111111111111111111111111111111
S12: 111111111111111111111111111111111
S13: 111111111111111111111111111111111
S14: 111111111111111111111111111111111
S15: 111111111111111111111111111111111111111
The Royal Road

S1: 1111111111111111
S2: 1111111111111111
S3: 1111111111111111
S4: 1111111111111111
S5: 1111111111111111
S6: 1111111111111111
S7: 1111111111111111
S8: 1111111111111111
S9: 1111111111111111
S10: 1111111111111111
S11: 1111111111111111
S12: 1111111111111111
S13: 1111111111111111
S14: 1111111111111111
S15: 1111111111111111

Mark Harman
ISSTA: Empirical and Theoretical Search Based Testing

Thursday, 10 December 2009
The Royal Road
The Royal Road
The Royal Road

S1: 1111************************************************************************
S2: ****1111************************************************************************
S3: ********1111************************************************************************
S4: ************1111************************************************************************
S5: ****************1111************
S6: ************************1111****
S7: ****************************1111
S8: ................................1111
S9: 11111111************************************************************************
S10: ****************11111111************************************************************************
S11: ******************11111111************
S12: *************************11111111
S13: 11111111111111111111111111111111
S14: 1111111111111111111111111111111111111111111
S15: 1111111111111111111111111111111111111111111
The Royal Road

S1: 1111******************************
S2: ****1111***************************
S3: ********1111********************
S4: ************1111****************
S5: ****************1111************
S6: ********************1111********
S7: ************************1111****
S8: ****************************1111
S9: 11111111************************
S10: ********11111111*************
S11: ******************11111111*****
S12: *********************11111111
S13: 1111111111111111**********
S14: ******************111111111111
S15: 111111111111111111111111
## Subjects

<table>
<thead>
<tr>
<th>Subject</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bibclean</td>
<td>open source BibTeX pretty printer</td>
</tr>
<tr>
<td>Eurocheck</td>
<td>open source € serial number validation</td>
</tr>
<tr>
<td>Gimp</td>
<td>open source image manipulation</td>
</tr>
<tr>
<td>Spice</td>
<td>analogue circuit simulator</td>
</tr>
<tr>
<td>Tiff</td>
<td>TIFF library for image manipulation</td>
</tr>
<tr>
<td>Space</td>
<td>ever heard of this one?</td>
</tr>
</tbody>
</table>
## Experimental set up

<table>
<thead>
<tr>
<th>Fitness evaluations</th>
<th>100,000</th>
</tr>
</thead>
<tbody>
<tr>
<td>Executions</td>
<td>30</td>
</tr>
<tr>
<td>Same seeds</td>
<td>for statistical testing</td>
</tr>
</tbody>
</table>
Overall Results

- Covered by all
- Uncovered or infeasible
- Covered by random only
- Covered by GA only
- Covered by hill climbing only
- Covered by GA and hill climbing only
Results

From 640 branches in the six subjects
10 branches
  for which Evolutionary Testing was successful but a simple Hill Climb search was not
5 branches
  for which Hill Climbing was successful but Evolutionary Testing was not
26 branches
Comparability
Royal Roads?

8 branches

Evolutionary Testing succeeds where Hill Climbing fails:

Branches of the *bibclean* test object
String check for a valid ISBN/ISSN
At least 10 digits
Headless chicken test: success rate

![Graph showing success rates for various checks]

-_headless chicken test: success rate

**Thursday, 10 December 2009**
Headless chicken test: test effort

Average No. of Fitness Evaluations

- check_ISBN (23F)
- check_ISBN (27T)
- check_ISBN (29F)
- check_ISBN (29T)
- check_ISSN (23F)
- check_ISSN (27T)
- check_ISSN (29F)
- check_ISSN (29T)

Legend:
- Normal Parents
- Rnd 2nd Parent
Comparison Local vs Global

Av. No. of Fitness Evaluations

Evolutionary Testing
Hill Climbing

Thursday, 10 December 2009
Comparison Local vs Global

Av. No. of Fitness Evaluations

Evolutionary Testing
Hill Climbing

Thursday, 10 December 2009
HC outperforms GA

HC is fast, easy and effective
In 24 of the 26 comparable cases it beats GA
Average speed up is approximately a factor of 20
The results were statistically significant (paired t test)
Conclusions

GA does perform well for Royal Road Functions
... and this is because of the cross over operator

But how many real programs have royal roads?

For those which don’t HC is comfortably faster
... by an order of magnitude
... evolution strategies may outperform GA for RR

Of course random covers most branches
... but only the easy ones
Future work

Memetic algorithms
Evolution strategies
Multi objective test data generation
Study of SBT and DART
Other GA theories