Genetic Improvement of Data gives double precision invsqr \( \frac{1}{\sqrt{x}} \)

GI @ GECCO 2019 workshop, Saturday 13 July, Prague. doi:10.1145/3319619.3326800

W. B. Langdon
Department of Computer Science
Genetic Improvement of Data gives double precision invsqrtr $\frac{1}{\sqrt{x}}$

W. B. Langdon
Department of Computer Science
Fluid Genetic Improvement Programming

- $\frac{1}{\sqrt{x}}$ inverted square root
- New type of Genetic Improvement
- Update fluid embedded literals i.e. data
  1. New functionality
  2. Better non-functionality (e.g. faster)?
- Why
  1. FGIP is a new way to do GI, tackle data driven code
  2. Minimal code changes may be more acceptable?
Why invsqrtsqrt x

- Reciprocal of square root for normalising vectors. E.g. in games, image processing and neural networks
- Before hardware support in GPU invsqrtsqrt was computationally heavy. Hence Quake used own approximation, Quare root.
FGIP New type of Genetic Improvement

• New functionality via data changes
  – RNAfold better predictions
  – New maths functions, e.g. convert $\sqrt{x}$ into $\frac{1}{\sqrt{x}}$

• Can we do non-functionality (e.g. faster)?

• How is this different from:
  – Deep parameter tuning? Change functionality
    – No mutation testing
    – Emphasis on data rather than values in code
    – Scale: hundreds even thousands of values
    – Multiple examples
Maintaining Embedded Constants

• **EuroGP 2018**
  – RNAfold 7000 lines of code 50000 numbers
  – On average better predictions of RNA folding.
  – Shipped since 2.4.7

• CMA-ES evolves data in a GNU C library sqrt to give new functionality with double precision accuracy. sqrt converted to
  – cube root, cbrt
  – square root converted to log₂ (poster monday)
  – \( \text{invsqrt} \frac{1}{\sqrt{x}} \)
  – division less division
Use CMA-ES to convert \( \sqrt{x} \) into \( \frac{1}{\sqrt{x}} \)

By updating table of 512 floats

IEEE 754 Double Precision
GNU C library sqrt converted to $\frac{1}{\sqrt{x}}$

- Chosen implementation of sqrt divides normalised input into 512 bins.
- Each bin holds start point for Newton-Raphson and initial derivative (two floats)
- Run CMA-ES per table element:
  - Seed with square root value, ignore 2\textsuperscript{nd} float
  - Run code with CMA-ES generated test value
  - Fitness = $\log |1/(c^2) - x|$ where $c=\text{invsqrt'}(x)$
    - $x$ takes three test values: smallest, mid, max in bin
  - Random restart if search fails to find value for which 3 diffs are smallest possible
CMA-ES

$sqrt$ seed values close to $invsqrt$

Evolved value very close to theory.

W. B. Langdon, UCL
Evolved $\frac{1}{\sqrt{x}}$

Evolved cbrt tested many thousands of times

- Always within DBL_EPSILON
- Almost always gives best possible double

Compared to Quake (single precision approximation)

- Quake seldom gives exact answer
- Quake can be 0.17% wrong (0.43/256)
- Quake does not trap negative numbers, sometimes fails, sometimes just wrong
- Quake odd behaviour $<1.5 \times 10^{-37}$ or $>3.3 \times 10^{38}$
Automatic Software Maintenance

• In a world addicted to software, maintenance is the dominant cost of computing.

• Need to keep parameters up to date. E.g.
  – New science, new laws or regulations, new users, new user expectations
  – Change of load, new hardware (e.g. bigger RAM), automatic porting
  – Search can be fast (total CMA-ES runtime 6 secs)

• Little SBSE research

• Great scope for automation
Summary: FGIP

- Problem of maintaining data in code ignored
- GI can optimize data in programs
- Rapidly generated maths (\(\text{cbrt}, \log_2, \frac{1}{\sqrt{x}}, \frac{1}{x}\))
  - Frame work open to evolve new functions
    http://www.cs.ucl.ac.uk/staff/W.Langdon/ftp/gp-code/gi_cbrt.tar.gz
- Software is not fragile
END

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

W. B. Langdon

CREST

Department of Computer Science
Improving RNAfold parameters

- RNAfold 7100 lines of C source code, 51521 parameters.
- Fitness correlation between prediction and true structure (MCC).
- Post evolution tidy
- 14732 (29%) parameters changed
- Holdout set significant increase in MCC
- Also better than constrained optimisation
- GI parameters [RNA Langdon2018.par](#) shipped with ViennaRNA since 13 Jun 2018

EuroGP-2018
Manual Changes I

• Most implementations of square root use hardware support.
• GNU C library glibc 2.29 also includes Newton-Raphson iterative solution
• Trap bad values, e.g. negative
• Normalise double input to 0.5 .. 2.0
• Guaranteed convergence in three steps:
  – Update both estimate of $\sqrt{x}$ and derivative
• Apply square root to exponent, i.e. divide by 2
Code Changes II

• Normalise double precision input to 1.0..2
  – Update estimate of $x^{-1/2}$
  – Use reciprocal of derivative, i.e. $x^{3/2}$, directly

• Apply invsqrt to exponent, i.e. divide by 2 and make negative.

• Could we use code GI to further improve?

W. B. Langdon, UCL
Square root to binary log

Frame work as sqrt to cbrt but

– Derivative known
– CMA-ES one dimension at a time (512 times) very easy

W. B. Langdon, UCL
The Genetic Programming Bibliography

http://gpbib.cs.ucl.ac.uk/

13217 references, 12000 authors

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