Parameter Calibration using Meta-Algorithms

W.A. de Landgraaf A.E. Eiben V. Nannen

Vrije Universiteit Amsterdam
{wadlandg, gusz, volker}@cs.vu.nl

September 28, 2007

OUTLINE

OVERVIEW

ORIGINAL META-GA (GREFENSTETTE 1986)

- search space: 2^{18} parameter combinations of a GA
- improved known results by 3% with only 2000 evalutions

original Meta-GA (Grefenstette 1986)

- search space: 2^{18} parameter combinations of a GA
- improved known results by 3% with only 2000 evalutions

REVAC (Nannen, Eiben 2006)

- a smoothed **Univariate Marginal Distribution,** calibrates EA-parameters and measures the **Shannon entropy** of an **Estimation of Distribution Algorithm (EDA)**

Overview

original Meta-GA (Grefenstette 1986)

- search space: 2^{18} parameter combinations of a GA
- improved known results by 3% with only 2000 evalutions

REVAC (Nannen, Eiben 2006)

- a smoothed **Univariate Marginal Distribution,** calibrates EA-parameters and measures the **Shannon entropy** of an **Estimation of Distribution Algorithm (EDA)**

Evaluation

- Calibrate simple GA on many problems
- Compare performance on known problems
- Robustness: compare performance on unknown problems

Hierarchy of Experiments

Design Layer

- Meta Genetic Algorithm
- REVAC

Algorithm Layer

- Genetic Algorithm

Problem Layer

- Multimodal Landscapes

OUTLINE

Details of the Meta-GA

Grefenstette, 1986

Population	generational
Population size	100
Parent selection	fitness proportional
Mutation type	bit-flip
Mutation rate	0.001
Crossover	One-point
Crossover rate	0.5

REVAC

Variation of Univariate Marginal Distribution Algorithm*

searches for maximum entropy distribution by *continous smoothing*

+

Information Theory

using Shannon entropy of maximum entropy distribution to measure relevance

* Mühlenbein (1996)

REVAC

Variation of Univariate Marginal Distribution Algorithm*

searches for maximum entropy distribution by *continous smoothing*

+

Information Theory

using Shannon entropy of maximum entropy distribution to measure relevance

Relevance Estimation and Value Calibration

converges on regions with high expected performance
smoothing maximizes entropy, converges on *broad* peaks

* Mühlenbein (1996)

Shannon entropy

Shannon entropy (differential)

$$H(X) = -\int_{-\infty}^{\infty} P(x) \log_2 P(x)$$

Shannon entropy

Shannon entropy (differential)

$$H(X) = -\int_{-\infty}^{\infty} P(x) \log_2 P(x)$$

Robustness

maximum Shannon entropy
$\Rightarrow$ broad peak,
large confidence interval

Information

Shannon entropy is information
$\Rightarrow$ a measure of
parameter relevance!

DUAL VIEW: POPULATION

CONVENTIONAL POPULATION BASED VIEW:

- population of *m* parameter strings $\vec{\mathbf{x}} = \{x^1 \ldots x^k\}$
- evolution through *crossover* and *mutation*

	k parameters				
$\vec{\mathbf{x_1}}$	x_1^1	$\cdots$	x_1^i	$\cdots$	x_1^k
$\vdots$					
$\vec{\mathbf{x_j}}$	x_j^1	$\cdots$	x_j^i	$\cdots$	x_j^k
$\vdots$					
$\vec{\mathbf{x_m}}$	x_m^1	$\cdots$	x_m^i	$\cdots$	x_m^k

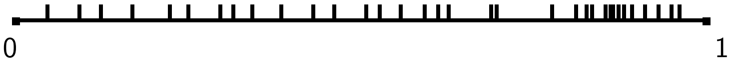
PROBABILISTIC VIEW:

- k marginal distributions $P(x)$
- evolution through *sampling* and *smoothing*

	$\mathbf{P^1(x)}$	$\cdots$	$\mathbf{P^i(x)}$	$\cdots$	$\mathbf{P^k(x)}$
$\vec{x_1}$	x_1^1	$\cdots$	x_1^i	$\cdots$	x_1^k
$\vdots$					
$\vec{x_j}$	x_j^1	$\cdots$	x_j^i	$\cdots$	x_j^k
$\vdots$					
$\vec{x_m}$	x_m^1	$\cdots$	x_m^i	$\cdots$	x_m^k

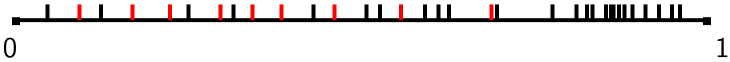
Drawing a new sample $P(x)$

- from population of size p select the q best



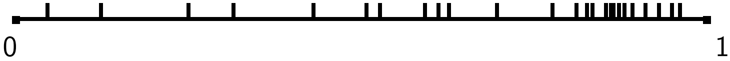
Drawing a new sample $P(x)$

- from population of size p select the q best



Drawing a new sample $P(x)$

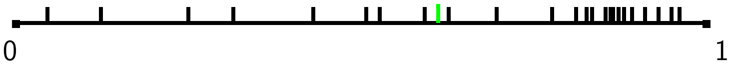
- from population of size p select the q best



How to construct the distribution

Drawing a new sample $P(x)$

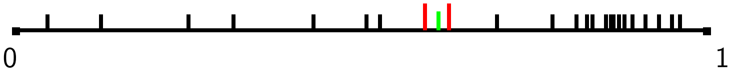
- from population of size p select the q best
- different parent for each parameter k



How to construct the distribution

Drawing a new sample $P(x)$

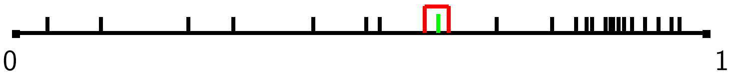
- from population of size p select the q best
- different parent for each parameter k
- take h^{th}upper neighbor a and h^{th}lower neighbor b along k



How to construct the distribution

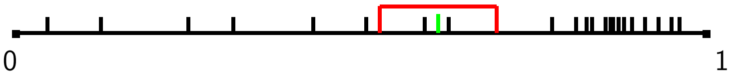
Drawing a new sample $P(x)$

- from population of size p select the q best
- different parent for each parameter k
- take h^{th}upper neighbor a and h^{th}lower neighbor b along k
- uniform from between the neigbors



Drawing a new sample $P(x)$

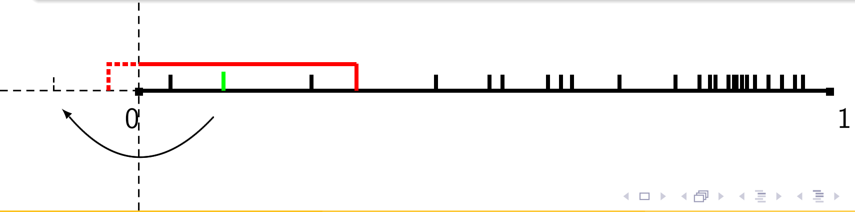
- from population of size p select the q best
- different parent for each parameter k
- take h^{th}upper neighbor a and h^{th}lower neighbor b along k
- uniform from between the neigbors
- h controls smoothing, quality, computational cost



How to construct the distribution

Drawing a new sample $P(x)$

- from population of size p select the q best
- different parent for each parameter k
- take h^{th}upper neighbor a and h^{th}lower neighbor b along k
- uniform from between the neigbors
- h controls smoothing, quality, computational cost
- mirror the sample at the edges



Outline

Design Layer

Meta Genetic Algorithm

REVAC

Algorithm Layer

Genetic Algorithm

Problem Layer

Multimodal Landscapes

Application layer: Simple GA

Simple GA

Population	Steady-state
Crossover	Uniform
Mutation	Bit-flip
Parent selec.	Tournament selection
Survival selec.	Delete-worst
Termination	Fitness of 1.0 or 10000 evaluations

4 Free Parameters

Parameters to be calibrated	Best known value
Crossover rate	0.5
Mutation rate	0.01
Population size	100
Tournament size	2

Problem layer: Multimodal problem generator

Spears, 2000

Strings of 100 bits,
{1, 2, 5, 10, 25, 50, 100, 250, 500, 1000} peaks.
Peaks have height between 0 and 1.
negative Hamming distance to closest peak + height of that peak.

Performance measure

Number of GA-evaluations to reach highest peak.
Termination after 3000 GA-evaluations.

Comparing GA performance

Peaks	hand-tuned	meta-GA	REVAC
1	1.0	1.0	1.0
2	1.0	1.0	1.0
5	1.0	0.988	1.0
10	0.9961	0.993	0.996
25	0.9885	0.994	0.991
50	0.9876	0.994	0.995
100	0.9853	0.983	0.989
250	0.9847	0.992	0.966
500	0.9865	0.989	0.970
1000	0.9891	0.987	0.985

Performance measure:

mean best fitness

And the winner is:

no winner;
no statistical significance

Comparing robustness

Measuring robustness

We test the robustness of the calibrated parameter vectors to changes in the problem specification by running the GA on landscape X with parameters found for landscape Y, for all numbers of peaks

Measured: best fitness at termination, averaged over 100 runs of each combination of X and Y.

hand-tuned	meta-GA	REVAC
0.915	0.911	0.927

And the winner is:

REVAC seems to win.

But again: no statistical significance.

OUTLINE

SUMMERY

- Two meta-algorithms: Meta-GA, REVAC

- Two meta-algorithms: Meta-GA, REVAC
- REVAC also estimates parameter relevance

- Two meta-algorithms: Meta-GA, REVAC
- REVAC also estimates parameter relevance
- no indication of improved GA performance

- Two meta-algorithms: Meta-GA, REVAC
- REVAC also estimates parameter relevance
- no indication of improved GA performance
- indication, but no statistical significant proof that REVAC is more robust

- Two meta-algorithms: Meta-GA, REVAC
- REVAC also estimates parameter relevance
- no indication of improved GA performance
- indication, but no statistical significant proof that REVAC is more robust
- REVAC calibrates as least as good as Meta-GA and ordinary methods.

- Two meta-algorithms: Meta-GA, REVAC
- REVAC also estimates parameter relevance
- no indication of improved GA performance
- indication, but no statistical significant proof that REVAC is more robust
- REVAC calibrates as least as good as Meta-GA and ordinary methods.

THANK YOU!

- www.few.vu.nl/∼wadlandg/pcma
- www.few.vu.nl/∼gusz/resources
- complexity-research.org/revac