An Analysis of a Model-based Evolutionary Algorithm: Learnable Evolution Model

Coletti, Mark

http://hdl.handle.net/1920/8922

a service of Mason Publishing
An Analysis of a Model-based Evolutionary Algorithm: Learnable Evolution Model

Dissertation Defense

M. Coletti

Dr. Kenneth De Jong, Chair
Dr. Sean Luke
Dr. Carlotta Domeniconi
Dr. Thomasz Arciszewski

April 28, 2014
Outline

1. Introduction
   - Evolutionary Algorithms
   - Model-based Evolutionary Algorithms
   - Canonical Model-based Evolutionary Algorithms

2. Learnable Evolution Model

3. Motivation

4. Methodology

5. Results from Rule Interval Sampling

6. Results from Training Set Configuration

7. General Results

8. Conclusion
Evolutionary Algorithms (EA) are biologically inspired problem solvers.

\[ P_0 = \text{initialize}(n) \]
\[ P_{t+1} = \text{survive}(\text{reproduce}(\text{select}(P_t))) \]
Model-based Evolutionary Algorithms (MBEA) derive a model from the population, and then use that model to create offspring.

\[ P_0 = initialize(n) \]

\[ P_{t+1} = survive(reproduceWithModel(buildModel(select(P_t)))) \]
Canonical Model-based Evolutionary Algorithms

**Cultural Algorithms (CA)**  Update “belief space” from best individuals and use that to influence operators.

**Estimation of Distribution Algorithms (EDA)**  Derive a statistical model from population then create offspring directly from that.

**Learnable Evolution Model (LEM)**  Use a Machine Learner (ML) to learn a model representing regions of higher fitness from best and least fit individuals, then exploit that knowledge to create offspring in those regions.
How LEM Works
How LEM Works
How LEM Works
How LEM Works
How LEM Works
How LEM Works
How LEM Works
Rules Define Regions of Higher Fitness

- “Genomes” for individuals are fixed-length real-valued vectors
- Each “gene” is a vector element
- Rules are conjunctions of $[l, u]_i$
- $l$ is a lower bound, $u$ an upper
- applied to the $i$th gene when creating offspring

$[-6.128321, 5.608972]_2 \land [-4.885725, 3.176545]_4$
Creating Offspring by Sampling Rule Intervals

1. when an individual selected to be a parent, it is copied to make start of new offspring
2. then a rule is selected to be applied to that copy
3. for each gene an interval references, assign new gene value from a sample from a distribution
Outline

1. Introduction
2. Learnable Evolution Model
3. Motivation
   - State of the Art
   - Expanding Our Knowledge of LEM
   - Focus
4. Methodology
5. Results from Rule Interval Sampling
6. Results from Training Set Configuration
7. General Results
8. Conclusion
LEM has been applied to a number of real-valued optimization problems.

- Digital signal filter problems (Coletti, et al. 1999)
- Test functions such as Rosenbrock, Rastrigin, and Griewangk (Wojtusiak, Michalski, 2006)
- Atmospheric source detection problem (Cervone, 2011)


We Do Not Know Much About LEM

- Design is very complex with a lot of hand optimization with unknown usefulness
- LEM treated like a black box with dogmatic design parameters
  - Training set configurations used top- and bottom-most third as positive and negative training examples
  - Rule intervals sampled using a uniform distribution

Practitioners have little in the way of design guidelines.
EA with no selection pressure is a random walk.

Same EA except that offspring created using ML inferred rules.

Focus

Research focus

- simplified version of LEM, SLEM
- rule interval sampling
- training set configuration

Research outcome

Guidelines for practitioners wanting to assemble these kinds of systems.
Outline

1. Introduction
2. Learnable Evolution Model
3. Motivation
4. Methodology
   - Experimental Set-up
5. Results from Rule Interval Sampling
6. Results from Training Set Configuration
7. General Results
8. Conclusion
Experimental Set-up

- Deterministic parent selection – every parent produces exactly one offspring
- Non-overlapping generations
- No mutation (except from using rules)
- No crossover
Spheroid and Griewangk Test Functions

**Spheroid**

\[ f(\vec{x}) = \sum_{i=1}^{n} x_i^2 \]

(Image courtesy of Sean Luke <sean@cs.gmu.edu>)

**Griewangk**

\[ f(\vec{x}) = 1 + \sum_{i=1}^{n} \frac{x_i^2}{4000} - \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}}\right) \]

(Image courtesy of Sean Luke <sean@cs.gmu.edu>)
Outline

1. Introduction
2. Learnable Evolution Model
3. Motivation
4. Methodology
5. Results from Rule Interval Sampling
   - Missing Rule Interval Bounds
   - Sampling Rule Intervals
   - General Rule Interval Guidelines
6. Results from Training Set Configuration
7. General Results
8. Conclusion
Missing Rule Interval Bounds

Example PART rule
Gene3 > -5.132166 AND
Gene3 <= 6.423435 AND
Gene0 <= 8.38355: best (25.0/2.0)

Corresponding SLEM Rule
[?, 8.38355]_0 \land [-5.132166, 6.423435]_3

Problem
We need both rule interval bounds in which to sample when creating offspring.
Missing Rule Interval Bounds

Rule with Missing Lower Bound

\[ ? \in [8.38355]_0 \land [-5.132166, 6.423435]_3 \]

Repaired Rule Interval for Gene Zero

- **init**: \([-10, 8.38355]\)
- **global**: \([-6.4, 8.38355]\)
- **best**: \([-3.0, 8.38355]\)
Results for Spheroid

global - best medians
Unbounded Rule Interval Strategies

Results for randomly rotated Griewangk

- Fitness
- Generation
- Est. Differences in Medians

global - best medians
## Guidance for Rule Interval Strategies

- *init* unbounded rule interval strategy disruptive
- Assigning missing rule bounds from only the *best* gene extrema has higher convergence rate than from *global* extrema
Sampling Rule Intervals

- **uniform**
- **Gaussian**
- **histogram-based**
Sample histograms for $\alpha$ values in $\{0, 0.2, 0.5, 0.8\}$
Population trajectories for spheroid
Rule Interval Sampling Strategies
Rotated Griewangk

Population trajectories for randomly rotated Griewangk
Rule Interval Sampling Strategies

Guidance for Rule Interval Sampling Strategies

- **convergence rate:** uniform $<$ histogram $<$ Gaussian
- $\alpha$ modulates convergence rate for histogram-based interval;
  convergence rate $\propto 1/\alpha$
- $3\sigma$ Gaussian converges more quickly than 2
General Rule Interval Guidance

Now have road map of rule interval related design decisions for modulating ML induced selection pressure.
Outline

1. Introduction
2. Learnable Evolution Model
3. Motivation
4. Methodology
5. Results from Rule Interval Sampling
6. Results from Training Set Configuration
   - Training Set Configurations
   - By Fitness Rank
   - By Fitness Percentage
   - General Training Set Guidance
7. General Results
8. Conclusion
Population trajectories for spheroid

Est. Differences in Medians

gap - split for medians
Population trajectories for randomly rotated Griewangk

gap - split for medians
Training Set Configuration Results by Rank

**Guidance for Training Set Configuration by Rank**

- **convergence rate:** middle < split < gap
- validation of Wallin and Ryan model
Two different approaches for creating training sets, one by fitness rank and the other by fitness percentage.

Population trajectories for spheroid

**byPercentage - byRank** medians
Training Set Configuration Results by Percentage

Population trajectories for randomly rotated Griewangk

byPercentage - byRank medians
Training Set Configuration Results by Percentage

Guidance for Training Set Configuration by Percentage

**convergence rate:** percentage < rank
General Training Set Guidance

Now have road map of design decisions for modulating ML training set convergence rates.
Outline

1. Introduction
2. Learnable Evolution Model
3. Motivation
4. Methodology
5. Results from Rule Interval Sampling
6. Results from Training Set Configuration
7. General Results
   - Combined Effects
   - Implicit Constraints from Initial Population
   - Scenarios Where the ML Learns Nothing
8. Conclusion
Separate Effects

Convergence rate poset for rule interval sampling strategies

Training set configurations selection pressure poset
Combining convergence rates from rule interval sampling with training set selection effects produces partially ordered set of *exploration vs. exploitation* effects.

- Can work from design decisions that emphasize *exploitive* effects the most, \{\textit{gap,rank,Gaussian,best}\}, to those emphasizing *exploration*.
- There is no clear single design set that emphasizes *exploration* the most.
Combining convergence rates from rule interval sampling with training set selection effects produces partially ordered set of *exploration vs. exploitation* effects.

- Can work from design decisions that emphasize *exploitive* effects the most, \{\text{gap,rank,Gaussian,best}\}, to those emphasizing *exploration*
- There is no clear single design set that emphasizes *exploration* the most
Resolving Most Explorative Design Configuration
Randomly rotated Griewangk

![Graph showing Fitness and Est. Differences in Medians over generations for different factors.]

Factors:
- \{\text{middle, percentage, Gaussian, best}\}
- \{\text{gap, rank, uniform, global}\}

\{\text{middle, percentage, Gaussian, best}\} - \{\text{gap, rank, uniform, global}\} medians
Complete Combined Effects

\{gap, rank, uniform, global\} updated as design configuration with that emphasizes exploration the most
Implicit Constraints from Initial Population

(Image courtesy of Sean Luke
<sean@cs.gmu.edu>)

Design Guidance

The machine learner only knows what you tell it.
Gaussian Distribution Effects on Implicit Constraints

Design Guidance

Gaussian distribution capable of escaping implicit bounds, but not guaranteed.
Scenarios Where the ML Learns Nothing
When populations are homogeneous due to convergence

---

generation: 13
PART decision list
--------------
Gene3 > -19.12803 AND
Gene2 <= 8.842311: best (41.0/13.0)
: worst (19.0/2.0)
Number of Rules :  2

---

generation: 14
PART decision list
--------------
: best (60.0/30.0)
Number of Rules :  1

Aggregate learned rule counts for Rastrigin
and Griewangk runs
Scenarios Where the ML Learns Nothing
When the training sets are effectively noise

Langerman function

(Image courtesy of Sean Luke

<sean@cs.gmu.edu>)
Contributions

- Produced SLEM, a stripped down LEM easier to analyze and implement
- Provided design guidance for tuning exploration vs exploitation effects
  - Rule interval sampling
  - Training set configuration
- Identified previously unknown problems
  - Unbounded rule intervals
  - Initial population defines implicit bounds
  - ML cannot learn from too similar training sets
Future Work

- What are the effects of using different machine learners?
- Can we enhance exploration or exploitation effects by adding rule selection bias?
- Can we resolve observed problems by reintroducing EA operators?
# Acknowledgements

## Committee

Dr. Kenneth De Jong, Chair, Dr. Sean Luke, Dr. Carlotta Domeniconi, and Dr. Tomasz Arciszewski.

## Krasnow Institute for Advanced Study

Dr. Jeffrey K. Bassett, Eric “Siggy” Scott, Dr. Uday Kamath, and Dr. Jayshree Sarma. Dr. Claudio Cioffi-Revilla, Dr. William Kennedy, Dr. Tim Gulden, and Dr. Andrew Crooks.

## Dissertation writers group

Dr. Susan Farley, John McDowell, Changwei “Coco” Liu, Todd Gillette, David Hamilton, and others.

Dr. Guido Cervone for lengthy conversations regarding LEM, which played a significant role in this work.

My other writing group that includes Anne Brennan Hardy, Melisse Ilhan, Charles Cressey, Andy Thrasher, and Dr. April Mattix.
Related Publications

The Effects of Training Set Size and Keeping Rules on the Emergent Selection Pressure of Learnable Evolution Model.

Analysis of Emergent Selection Pressure in Evolutionary Algorithm and Machine Learner Offspring Filtering Hybrids.

Learnable Evolution Model Performance Impaired by Binary Tournament Survival Selection.
In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, Montréal, Canada. Best Paper, Graduate Student Workshop.

The relationship between evolvability and bloat.
In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, Montréal, Canada.

Preliminary Results of Learnable Evolution Methodology (LEM) Using C4.5.

In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, Orlando, Florida.
Thank you!