

Copyright is owned by the Author of the thesis. Permission is given for a copy to be downloaded by an individual for the purpose of research and private study only. The thesis may not be reproduced elsewhere without the permission of the Author.

COMPUTATIONAL COMPLEXITY OF ELITIST POPULATION-BASED EVOLUTIONARY ALGORITHMS

A THESIS PRESENTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR
THE DEGREE OF
DOCTOR OF PHILOSOPHY
IN
COMPUTER SCIENCE
AT MASSEY UNIVERSITY, PALMERSTON NORTH,
NEW ZEALAND.

Aram Ter-Sarkisov

2012

Contents

Acknowledgements	xi
Abstract	xii
List of Publications	xiv
Notation	xvi
1 Introduction	1
1.1 Introduction to Evolutionary Computation	1
1.2 Motivation	3
1.3 Main results	3
1.4 Outline of the thesis	5
2 Related Work	7
2.1 Schemata Theorem	7
2.1.1 Criticism of Schemata Theorem	9
2.1.2 Alternative explanation of EA efficiency	9
2.2 Convergence Analysis	10
2.3 Runtime Analysis	12
2.3.1 Runtime analysis of $(\mu + 1)$ and $(1 + \lambda)$ EAs	14
2.3.2 Runtime analysis of $(\mu + \lambda)$ EAs	15
2.4 Review of tools used for analyzing EAs	16
2.4.1 Fitness-Based Partition and Artificial Fitness Levels	17
2.4.2 Gambler’s Ruin and Coupon Collector’s Problem	17
2.4.3 Potential/Auxiliary Functions	19

2.4.4	Analysis of Typical runs	20
2.4.5	Structure of individuals in the population	20
2.5	Asymptotic notation	21
2.6	The No Free Lunch theorem and Analysis of computer algorithms .	22
2.7	Parallel computers	23
3	The K-Bit-Swap Genetic Operator	24
3.1	Explanation of the K-Bit-Swap Genetic Operator	24
3.2	Algorithms and Experimental setup	26
3.2.1	Problems selected for testing	27
3.3	Setup and Analysis of Statistical Tests	33
3.3.1	Statistical Analysis	37
3.4	Conclusions	38
4	Lower Bounds on the Runtime	40
4.1	Main results	40
4.2	Structure of the population and the recombination pool	41
4.3	Algorithms	43
4.4	Problems	44
4.4.1	OneMax	45
4.4.2	Royal Roads	45
4.5	Population-Based Evolutionary Algorithms and Distribution of Species	47
4.6	Runtime analysis of $(1 + 2)EA_{1BS}$ solving OneMax Problem	49
4.7	Main model of the $(\mu + \lambda)$ Algorithm on the OneMax Test Function	51
4.7.1	Runtime analysis of $(\mu + \lambda)EA_{1BS}$ on the OneMax problem	52
4.7.2	Asymptotic runtime of $(\mu + \lambda)EA_{1BS}$ on the OneMax Test function	56
4.7.3	Runtime analysis of $(\mu + \lambda)RLS$ on the OneMax Test Function	57
4.7.4	Asymptotic runtime of $(\mu + \lambda)RLS$ on the OneMax Test function	58
4.8	Main model of the $(\mu + \lambda)$ Algorithm on the Royal Roads Test Function	59
4.8.1	Runtime analysis of $(\mu + \lambda)EA_{1BS}$ on the RR Test Function	60

4.8.2	Asymptotic runtime of $(\mu + \lambda)$ EA _{1BS} on the RR Test Function	62
4.8.3	Runtime analysis of $(\mu + \lambda)$ RLS on the RR Test Function	64
4.8.4	Asymptotic runtime of $(\mu + \lambda)$ RLS on the RR Test Function	65
4.9	Numerical results	66
4.10	Conclusions	84
5	Upper Bounds on the Runtime	86
5.1	Main results	87
5.2	The Elitism Levels Traverse Mechanism	88
5.3	Upper bounds on the OneMax test function	91
5.3.1	Simple upper bound on OneMax	92
5.3.2	Refined upper bounds on OneMax	96
5.3.3	Use of $\langle \alpha, \alpha \rangle$ pair	101
5.3.4	Generations vs Function evaluations	101
5.3.5	Comparison to earlier results	102
5.4	Upper Bounds on the Royal Roads test function	102
5.4.1	The birth-and-death Markov Chain for Royal Roads	104
5.4.2	Upper bounds on the Royal Roads problem	106
5.4.3	Proof of the lower bound on the probability of advancing to the next artificial auxiliary level	116
5.4.4	Lower bounds on the probabilities involving η species in Phase 2	119
5.5	Approximation of the quasi-stationary distribution of super-elite species in Phase 1	121
5.5.1	Slow progress rate (Poisson approximation)	123
5.5.2	Fast progress rate (Normal approximation)	123
5.6	Conclusions	125
6	Summary, Conclusions and Future Work	128
A	Results of Numerical Experiments	133
B	Concepts from Probability Theory	147

List of Tables

1.1	Canonical Genetic Algorithm	2
3.1	The K-Bit-Swap Genetic Operator	25
3.2	Pseudocode of EAs in Chapter 3	27
3.3	Parameter settings for the problem set.	32
3.4	Benchmark settings	32
3.5	Estimate of the probability of failure, Equation 3.3	36
3.6	Estimate of the conditional probability of success, Equation 3.4	36
3.7	Estimate of the conditional expectation, Equation 3.5	37
4.1	$(\mu + \lambda)$ EA _{1BS}	44
4.2	$(\mu + \lambda)$ RLS	44
4.3	Selection Function	45
4.4	Set of parameters used for OneMax test function	66
4.5	Set of parameters used for the RR test function	67

List of Figures

3.1	Comparison of K-Bit-Swap to simple (segment) crossover, 2-point simple (segment) crossover and Uniform crossover	25
3.2	The global solution for the trivial k-means problem. Darker points are data, lighter are centroids	33
4.1	Distribution of the elite species in the population of $(\mu + \lambda)$ EA _{1BS} solving OneMax Test Function for $\mu = \lambda = 500$ and stopped at the achievement of the global optimum	67
4.2	Distribution of the elite species in the population of $(\mu + \lambda)$ RLS solving OneMax Test Function for $\mu = \lambda = 500$ and stopped at the achievement of the global optimum	68
4.3	Probability of success of $(\mu + \lambda)$ RLS solving OneMax Test Function.	69
4.4	Numerical runtime estimate for $(\mu + \lambda)$ EA _{1BS} solving OneMax Test Function for different population sizes.	70
4.5	Numerical runtime estimate for $(\mu + \lambda)$ RLS solving OneMax Test Function for different population sizes.	71
4.6	Theoretical and numerical estimate for $(\mu + \lambda)$ EA _{1BS} solving OneMax Test Function	72
4.7	Theoretical and numerical estimate for $(\mu + \lambda)$ RLS solving OneMax Test Function	73
4.8	Distribution of the elite species in the population of $(\mu + \lambda)$ EA _{1BS} solving Royal Roads Test Function for $\mu = \lambda = 500$ and stopped at the achievement of the global optimum	74

4.9	Distribution of the elite species in the population of $(\mu + \lambda)$ RLS solving Royal Roads Test Function for $\mu = \lambda = 500$ and stopped at the achievement of the global optimum	75
4.10	Probability of success of $(\mu + \lambda)$ EA _{1BS} solving Royal Roads Test Function. For $n = 32, 64$ it is always almost 1	76
4.11	Probability of success of $(\mu + \lambda)$ RLS solving Royal Roads Test Function. For $n = 32, 64$ it is always almost 1	77
4.12	Numerical runtime estimate for $(\mu + \lambda)$ EA _{1BS} solving Royal Roads Test Function for different population sizes. The positive effect of the population size measured in the number of generations is obvious.	78
4.13	Numerical runtime estimate for $(\mu + \lambda)$ RLS solving Royal Roads Test Function for different population sizes. The positive effect of the population size measured in the number of generations is obvious.	79
4.14	Theoretical and numerical bounds for $(\mu + \lambda)$ EA _{1BS} solving Royal Roads Test Function	80
4.15	Theoretical and numerical bounds for $(\mu + \lambda)$ RLS solving Royal Roads Test Function	81
A.1	Conditional probability of success and runtime of $(\mu + \lambda)$ EA _{KBS} vs $(\mu + \lambda)$ EA _{-KBS} on the Rosenbrock test function	133
A.2	Conditional probability of success and runtime of $(\mu + \lambda)$ EA _{KBS} vs $(\mu + \lambda)$ EA _{-KBS} on the Rastrigin test function	134
A.3	Conditional probability of success and runtime of $(\mu + \lambda)$ EA _{KBS} vs $(\mu + \lambda)$ EA _{-KBS} on the Ackley test function	134
A.4	Conditional probability of success and runtime $(\mu + \lambda)$ EA _{KBS} on the Royal Roads test function. Algorithms with other parameter settings do not solve the problem in the set number of generations.	135
A.5	Conditional probability of success and runtime of $(\mu + \lambda)$ EA _{KBS} vs $(\mu + \lambda)$ EA _{-KBS} on the Four Peaks test function	135
A.6	Conditional probability of success and runtime of $(\mu + \lambda)$ EA _{KBS} vs $(\mu + \lambda)$ EA _{-KBS} on the trivial TSP	136

A.7	Conditional probability of success and runtime of $(\mu + \lambda)EA_{-KBS}$ on the TSP on US Capital Cities. Algorithms with KBS do not solve the problem in the set number of generations	136
A.8	Probability of success $(\mu + \lambda)EA_{KBS}$ vs $(\mu + \lambda)EA_{-KBS}$ on the trivial k-means clustering problem	137
A.9	Probability of success $(\mu + \lambda)EA_{KBS}$ vs $(\mu + \lambda)EA_{-KBS}$ on the random k-means clustering problem	137
A.10	Histograms of bootstrap estimate of the difference in means for the Rosenbrock function: probability of failure, conditional probability of success, runtime	138
A.11	Histograms of bootstrap estimate of the difference in means for the Rastrigin function: probability of failure, conditional probability of success, runtime	139
A.12	Histograms of bootstrap estimate of the difference in means for the Ackley function: probability of failure, conditional probability of success, runtime	140
A.13	Histograms of bootstrap estimate of the difference in means for the Royal Roads function: probability of failure, and conditional probability of success	141
A.14	Histograms of bootstrap estimate of the difference in means for the Four Peaks function: probability of failure, conditional probability of success, runtime	142
A.15	Histograms of bootstrap estimate of the difference in means for the TSP on a circle: probability of failure, conditional probability of success, runtime	143
A.16	Histograms of bootstrap estimate of the difference in means for the TSP on US Cities: probability of failure and conditional probability of success	144
A.17	Histograms of bootstrap estimate of the difference in means for the trivial k-means clustering problem: probability of failure, conditional probability of success, runtime	145

A.18 Histograms of bootstrap estimate of the difference in means for the random k-means clustering problem: probability of failure, condi- tional probability of success, runtime	146
---	-----

Für meine Großeltern, Bertha und Edward

Acknowledgements

I would like to thank first of all my supervisors, Associate Professor Stephen Marsland, Professor Chin-Diew Lai and Doctor Barbara Holland. Without their guidance and support this thesis would never be possible. I would like to thank everyone who was helping me in many ways throughout more than three past years, especially my parents, my girlfriend, my girlfriend's parents and grandparents.

Abstract

Evolutionary Algorithms (EAs) are a modern heuristic algorithm that have proven efficiency on a large number of real-life problems. Despite the rich history of applications understanding of both how and why EAs work is lagging far behind. This is especially true for one of the main components of EAs, that is hypothesized by many to underlie their efficiency: population.

The first problem considered in this thesis is the introduction of a recombination operator, K-Bit-Swap (KBS) and its comparison to mainstream operators, such as mutation and different types of crossover. A vast amount of statistical evidence is presented that shows that EAs using KBS outperform other algorithms on a whole range of problems. Two problems are selected for a deep theoretical analysis: OneMax and Royal Roads.

The main problem of modeling EAs that use both population and a pool of parents is the complexity of the structures that arise from the process of evolution. In most cases either one type of species is considered or certain simple assumptions are made about fitness of the species.

The main contribution of this thesis is the development of a new approach to modeling of EAs that is based on approximating the structure of the population and the evolution of subsets thereof. This approach lies at the core of the new tool presented here, the Elitism Levels Traverse Mechanism that was used to derive upper bounds on the runtime of EAs. In addition, lower bounds were found using simpler assumptions of the underlying distribution of species in the population.

The second important result of the approach is the derivation of limiting distributions of a subset of the population, a problem well-known in areas such as epidemiology. To the best of the author's knowledge, no such findings have been published in the EA community so far.

List of Publications

A. Ter-Sarkisov, S. Marsland, and B. Holland. The k-Bit-Swap: A New Genetic Algorithm Operator. In *Genetic and Evolutionary Computing Conference (GECCO) 2010*, pages 815–816, 2010

A. Ter-Sarkisov and S. Marsland. Convergence Properties of $(\mu + \lambda)$ Evolutionary Algorithms. In *25th AAAI Conference on Artificial Intelligence*, pages 1816–1817, 2011. Special Student Poster Session

A. Ter-Sarkisov and S. Marsland. Convergence Properties of Two $(\mu + \lambda)$ Evolutionary Algorithms on OneMax and Royal Roads Test Functions. In *International Conference on Evolutionary Computation Theorey and Applications (ECTA)*, pages 196–202, 2011

A. Ter-Sarkisov and S. Marsland. Convergence of a Recombination-Based Elitist Evolutionary Algorithm on the Royal Roads Test Function. In *24th Australasian Joint Conference on Artificial Intelligence*, pages 361–371, 2011

A. Ter-Sarkisov. Elitism Levels Traverse Mechanism For The Derivation of Upper Bounds on Unimodal Functions. In *WCCI 2012 IEEE World Congress on Computational Intelligence*, pages 2161–2168, 2012

A. Ter-Sarkisov and S. Marsland. Derivation of Upper Bounds on Optimization Time of Population-Based Evolutionary Algorithm on a Function with Fitness Plateaus Using Elitism Levels Traverse Mechanism. 2012. arXiv:1204.2321. To be submitted

Notation

The notation for species, $\alpha^* \dots \eta$ is used to denote both the type and the size of the type, i.e. instead of $|\alpha^*| \dots |\eta|$.

α	Elite species
α^*	Super-elite species
β	Non-elite species with the next-best fitness to α
β^*	Elite species with the next-best auxiliary value to α^*
γ	Non-elite species other than β
γ^*	Elite species other than α^* and β^*
δ	Proportion of elite species in the population
δ^*	Proportion of super-elite species in the population
η	All non-elite species in the population (both β and γ)
φ	Probability to swap bits between two parents in the recombination pool

λ	Size of the recombination pool
μ	Size of the population
K	Number of bins (plateaus of fitness) in a string
M	Size of the bin (length of the plateau of fitness)
M	Total number of types of infections in the population (only in Section 5.2)
m_j	Number of species with infection type j (only in Section 5.2)
$m_{1,\delta^*\alpha}$	Mean first hitting time of the absorbing state $\delta^*\alpha$ in a Markov Chain
n	Length of the string (total number of bits in the string)
N	Population size (only in Section 2.3.2)
(μ, λ)	Evolutionary Algorithm with population size μ and recombination pool size λ , no elitism
$(\mu + \lambda)$	Evolutionary Algorithm with population size μ and recombination pool size λ using some form of elitism
$P(H_j)$	Probability to select j pairs of elite parents (1BS) or j elite parents (RLS) into the recombination pool
$P(G_k)$	Probability to evolve at least one higher-ranked offspring given k improvements so far
$P(G_{0k})$	Probability to fail to evolve a higher-ranked offspring given k improvements so far

$P(\alpha)$	Probability to observe α elite parents in the population (Uniform)
$P_{sel,\alpha}$	Probability to select an elite pair (1BS) or species (RLS) into the recombination pool given α elite species in the population
P_{swap}	Probability to swap bits between parents using the KBS operator
P_{flip}	Probability to flip bits in a parent using RLS
rv	Random variable
S_{11}	The first expression in Phase 1
$S_{11}(\alpha^*)$	The summand in the first expression in Phase 1
S_{12}	The second expression in Phase 1
S_{21}	The first expression in Phase 2
S_{22}	The second expression in Phase 2
$\mathbf{E}\tau$	Mean first hitting time in a Markov Chain
s_k	k^{th} bin in the string
s	whole string
$V(s_k)$	Auxiliary value of k^{th} bin in the string (also V_k)
$V(s)$	Auxiliary value of the whole string (also V_s)