

# Theory of Evolutionary Algorithms

13.02. - 18.02.2000

organized by

H.-G. Beyer (Dortmund)

K. De Jong (Fairfax)

D. Fogel (San Diego)

I. Wegener (Dortmund)

During the last ten years computers have become fast enough to support evolutionary algorithms and a lot of applications to real-world problems have been developed. This has led to a great deal of empirical knowledge on the behavior of evolutionary algorithms and to many heuristics for choosing their associated parameters. There is also a developing theory of evolutionary algorithms based on tools from the analysis of randomized algorithms, of Markov processes, and of dynamical systems. The aim of this workshop was to contribute to this theory and to allow a discussion between researchers with different backgrounds.

The organizers are happy to report that 45 researchers accepted an invitation to Dagstuhl. They came from Germany (21), England (8), USA (5), France (2), Netherlands (2), Romania (2), Austria (1), Belgium (1), India (1), Mexico (1), and Poland (1). The 31 talks captured all the aspects of a theory of evolutionary algorithms, among them statistical dynamics, time-varying landscapes, convergence issues, complexity results, fitness landscapes, models of evolutionary algorithms, analysis of the run time of evolutionary algorithms, self-adaptation, new variants of evolutionary algorithms, and genetic programming. The discussion was extremely vivid. There was almost no talk that evoked fewer than five questions and remarks. The schedule included an informal evening session where eight topics suggested by the participants were discussed.

Besides the official schedule the participants used unscheduled time for many discussions and some informal sessions with short talks, all inspired by the special Dagstuhl atmosphere. The special event of the week was the Wednesday hike where it has snowed heavily on the way out and the sun shone on the way back through the snow.

# Seminar Program

Monday, February 14th, 2000

- 9.00 – 9.40 **Hans-Paul Schwefel**  
Still Missing Features in Current Evolutionary Algorithms
- 9.45 – 10.25 **Karsten Weicker**  
Towards a Theoretical Foundation of Dynamic Optimization
- 10.35 – 11.15 **Christopher Ronnewinkel**  
Genetic Algorithms in Time-Dependent Environments
- 11.20 – 12.00 **Dirk Arnold**  
Local Performance of Evolution Strategies in Noisy Environments
- 15.15 – 15.55 **Jonathan L. Shapiro**  
Statistical Mechanics of Genetic Algorithms
- 16.00 – 16.35 **Adam Pruegel-Bennett**  
Symmetry Breaking in Evolutionary Algorithms
- 16.45 – 17.20 **Gabriela Ochoa**  
Optimal Mutation Rates and Error Thresholds in Genetic Algorithms
- 17.25 – 18.00 **Stefan Droste**  
On the Analysis of Simple EAs with Dynamic Parameter Control

Tuesday, February 15th, 2000

- 9.00 – 9.40 **Roman Galar**  
Considering Evolution in the Space of Population States
- 9.45 – 10.25 **Riccardo Poli**  
Recursive Conditional Schema Theorem, Convergence and Population Sizing in Genetic Algorithms
- 10.35 – 11.15 **Nicole Weicker**  
Towards Qualitative NFL Results
- 11.20 – 12.00 **Xin Yao**  
Computational Time Complexity of Certain Evolutionary Algorithms
- 15.30 – 16.10 **Ivo Hofacker**  
Characterization of Landscapes
- 16.15 – 16.55 **Bart Naudts**  
Comparing Population Mean Curves
- 17.05 – 17.45 **Colin Reeves**  
Experiments with Tunable Landscapes
- 20.00 **Discussion Session**

Wednesday, February 16th, 2000

- 9.00 – 9.40 **Jeffrey Horn**  
Speciation as Computation
- 9.45 – 10.25 **Günter Rudolph**  
Recent Advances in the Theory of Multi-Objective  
Evolutionary Algorithms
- 10.35 – 11.15 **Alden Wright**  
Models of Steady State Genetic Algorithms
- 11.20 – 12.00 **Jonathan Rowe**  
Generalising the Dynamical Systems Model

Thursday, February 17th, 2000

- 9.00 – 9.40 **Thomas Jansen**  
Proving Crossover Useful
- 9.45 – 10.25 **Evelyne Lutton**  
Understanding EA: The Fractal Way
- 10.35 – 11.15 **Heinz Mühlenbein**  
From Genetic Recombination to Search Distributions and  
Graphical Models
- 11.20 – 12.00 **Ingo Wegener**  
On the Analysis of a Dynamic (1+1) EA
- 15.15 – 15.50 **Kalyanmoy Deb**  
Self-Adaptive Evolutionary Algorithms for Function  
Optimization
- 15.55 - 16.30 **Nikolaus Hansen**  
Keypoints in Strategy Parameter Control
- 16.40 – 17.15 **Martin Pelikan**  
Research on the Bayesian Optimization Algorithm or  
How the BOA Saved the Building Blocks
- 17.20 – 17.55 **Leila Kallel**  
How to Detect All Maxima of a Function?

Friday, February 18th, 2000

- 9.00 – 9.40 **William B. Langdon**  
Scaling of Program Fitness Spaces
- 9.45 – 10.25 **Wolfgang Banzhaf**  
On the Reason for Bloat in Genetic Algorithms
- 10.35 – 11.15 **Peter Stagge**  
Evolving Neural Structures
- 11.20 – 12.00 **Kenneth De Jong**  
Evolutionary Computation Theory: A Unified Approach

# Contents

Dirk Arnold: <i>Local Performance of Evolution Strategies in Noisy Environments</i> . . . . .	6
Wolfgang Banzhaf: <i>On the Reason for Bloat in Genetic Programming</i> . . .	6
Kenneth De Jong: <i>Evolutionary Computation Theory: A Unified Approach</i>	7
Kalyanmoy Deb: <i>Self-Adaptive Evolutionary Algorithms for Function Optimization</i> . . . . .	7
Stefan Droste: <i>On the Analysis of Simple EAs with Dynamic Parameter Control</i> . . . . .	8
Roman Galar: <i>Considering Evolution in the Space of Population States</i> . .	8
Nikolaus Hansen: <i>Keypoints in Strategy Parameter Control</i> . . . . .	9
Ivo Hofacker: <i>Characterization of Landscapes</i> . . . . .	10
Jeffrey Horn: <i>Speciation as Computation</i> . . . . .	10
Thomas Jansen: <i>Proving Crossover Useful</i> . . . . .	11
Leila Kallel: <i>How to Detect All Maxima of a Function?</i> . . . . .	12
William B. Langdon: <i>Scaling of Program Fitness Spaces</i> . . . . .	12
Evelyne Lutton: <i>Understanding EA: The Fractal Way</i> . . . . .	13
Heinz Mühlenbein: <i>From Genetic Recombination to Search Distributions and Graphical Models</i> . . . . .	14
Bart Naudts: <i>Comparing Population Mean Curves</i> . . . . .	15
Gabriela Ochoa: <i>Optimal Mutation Rates and Error Thresholds in Genetic Algorithms</i> . . . . .	15
Martin Pelikan: <i>Research on the Bayesian Optimization Algorithm or How the BOA Saved the Building Blocks</i> . . . . .	16
Riccardo Poli: <i>Recursive Conditional Schema Theorem, Convergence and Population Sizing in Genetic Algorithms</i> . . . . .	16
Adam Pruegel-Bennett: <i>Symmetry Breaking in Evolutionary Algorithms</i> .	16
Colin Reeves: <i>Experiments with Tunable Fitness Landscapes</i> . . . . .	17
Christopher Ronnewinkel: <i>Genetic Algorithms in Time-Dependent Environments</i> . . . . .	18
Jonathan Rowe: <i>Generalising the Dynamical Systems Model</i> . . . . .	18
Günter Rudolph: <i>Recent Advances in the Theory of Multi-Objective Evolutionary Algorithms</i> . . . . .	19
Hans-Paul Schwefel: <i>Still Missing Features in Current Evolutionary Algorithms</i> . . . . .	19
Jonathan L. Shapiro: <i>Statistical Mechanics of Genetic Algorithms</i> . . . . .	20

Peter Stagge: <i>Evolving Neural Structures</i> . . . . .	21
Ingo Wegener: <i>On the Analysis of a Dynamic (1+1) EA</i> . . . . .	22
Karsten Weicker: <i>Towards a Theoretical Foundation of Dynamic Optimization</i> . . . . .	22
Nicole Weicker: <i>Towards Qualitative NFL Results</i> . . . . .	23
Alden Wright: <i>Models of Steady State Genetic Algorithms</i> . . . . .	23
Xin Yao: <i>Computational Time Complexity of Certain Evolutionary Algorithms</i> . . . . .	24

# Local Performance of Evolution Strategies in Noisy Environments

Dirk Arnold

Noise is present in many optimization problems. Evolutionary algorithms are frequently reported to be robust with regard to the effects of noise. This talk addresses two separate aspects of the local performance of evolution strategies (ES) in a very simple fitness environment that may help explain the observed robustness.

For the  $(1 + 1)$ -ES, systematic overvaluation of the parental fitness is shown to naturally arise if the fitness of an individual is evaluated only once in its lifetime. It is demonstrated that reduced success probabilities and at the same time an increased fitness gain as compared to a strategy that reevaluates the parental fitness in every generation can be expected in the noisy environment investigated. Implications for success probability based mutation strength adaptation schemes are discussed.

For the  $(\mu/\mu_I, \lambda)$ -ES, the local performance on an infinite dimensional noisy sphere is discussed. It is shown that as in the noise free case genetic repair is present and that the performance of the strategy can be substantially improved by increasing the population size. Unlike the situation in the noise free case, the efficiency of the  $(1 + 1)$ -ES can be exceeded in the presence of noise.

## On the Reason for Bloat in Genetic Programming

Wolfgang Banzhaf

It has been well established over recent years, that in Genetic Programming as well as in other evolutionary algorithms with length-changing genotypes a tendency for growing length of genotypes occurs. This phenomenon has been termed "bloat". We introduce a very simple model for GP which does not use any representation in order to show that the tendency to growing length is a general and inherent tendency in all GP systems. We present a number of simulations and reconsider some of the explanations for bloat published in recent years, notably the "fitness causes bloat" and the "neutral code protection" hypotheses.

# **Evolutionary Computation Theory: A Unified Approach**

Kenneth De Jong

The rapid growth of interest in evolutionary computation has made it difficult for theoretical analyses to keep pace. Evolutionary computation theory continues to make progress, but in a somewhat fragmented way. In this talk a framework based on decomposition and abstraction hierarchies is presented as the basis for a unifying theory as well as encouraging sound design of new algorithms and applications.

## **Self-Adaptive Evolutionary Algorithms for Function Optimization**

Kalyanmoy Deb

Self-adaptation is an essential feature of natural evolution. However, in the context of function optimization, self-adaptation features of evolutionary search algorithms have been explored only with evolution strategy (ES) and evolutionary programming (EP). In this talk, we demonstrate the self-adaptive properties of real-parameter genetic algorithms (GAs) with a number of crossover operators, such as simulated binary crossover (SBX), blend crossover (BLX), and fuzzy recombination operator (FR). The connection between the working of self-adaptive ESs and real-parameter GAs with SBX operator is also discussed. The connection is made better by calculating the growth of population variance of the above three crossover operators and self-adaptive ESs. It is then postulated that for a good variance operator (i) the population mean should remain unchanged and (ii) the population variance should, in general, increase. Based on these two postulates, the equivalence of these EAs on flat fitness landscapes has been established. Simulation results on sphere models show similar performance of all the above self-adaptive EAs. These results suggest a unified approach to EAs: With similar population variance growth of variance operators, EAs with an identical selection operator are expected to have similar performance.

# On the Analysis of Simple EAs with Dynamic Parameter Control

Stefan Droste (joint work with Thomas Jansen)

As evolutionary algorithms (EAs) are controlled by a number of different parameters, it is an interesting open question, if some parameter control schemes are better than others. Here we present a proof, that for a carefully chosen fitness function and a very simple EA a variation of the selection probability only depending on the number of generations exponentially outperforms every static choice of this parameter with respect to expected runtime. Because the EA investigated is an instance of simulated annealing, this is another example, where simulated annealing outperforms the Metropolis algorithm. Furthermore, our proof is easy to understand, as it uses only basic stochastic formulas, like the Markov inequality.

## Considering Evolution in the Space of Population States

Roman Galar (joint work with Artur Chorazyczewski and Iwona Karcz-Duleba)

A simple model of asexual phenotypic evolution with non-overlapping generations assumes population of  $m$  individuals with  $n$  traits under proportional selection and independent  $N(0, s)$  distributed traits modifications. The state of such population is represented as a point in the  $n \cdot m$  real space and the appropriate formula for its probability distribution in the next generation is given which represents a standard Markov chains description. It is pointed that: (1) to avoid ambiguity in such representation population has to be ordered, what restricts the space of states to a specific part of  $\mathbb{R}^{n \cdot m}$ ; (2) for strongly diversified populations, i.e. with distances between individuals  $\gg s$ , the probability distribution of the next generation is distinctly multimodal, what accounts for jumps of populations state. Both points were illustrated for cases involving  $n = 1$  and  $m$  equal few, using simulator that draws population trajectory mapped in (position along identity axis, distance from identity axis) co-ordinates. It is visible then that such evolution is a two-speed process: (1) rapid concentration of individuals into population cluster with the radius of about  $s$ ; (2) drift of this cluster toward more elevated areas. The possibility of generalising these observations on large populations and multi-dimensional spaces are illustrated using appropriate simulations. Some prospects of utilising the space of states approach as a tool for better understanding of evolution and theorem proving are indicated.



# Keypoints in Strategy Parameter Control

Nikolaus Hansen

A dynamic strategy parameter control which depends more or less directly on the search points evaluated so far seems essential in evolution strategies. Four keypoints for strategy parameter control are identified:

- Adjustment of the change rate of the strategy parameters
- Stationarity of strategy parameters (for weak selection)
- Invariance (for prediction)
- Evaluation of success

Invariance is very attractive, because it raises the probability to get similar (or even identical) results on similar (not yet evaluated) problems. Any evaluation of search strategies, for example by test functions, is based on the implicit assumption of invariance. A simple evolution strategy is invariant against translation and rotation of the search space and against any strictly monotonically increasing (i. e. order-preserving) transformation of the fitness value. Invariances can be lost, if more complex operators are introduced. For example with discrete recombination or with an individual step size adaptation invariance against rotation is lost. This should be taken into account if different strategies are evaluated.

From a conceptual point of view one primary aim of strategy parameter control is to introduce new invariances. At the same time it seems desirable to retain the invariances of the simple evolution strategy. Two additional invariances are identified. (To become exactly invariant, object and strategy parameters have to be transformed accordingly to the search space.)

- Invariance against scalar multiplication of the search space. This can be facilitated by control of a global step size.
- Invariance against any (full rank) linear transformation of the search space. This can be facilitated by adaptation of the complete covariance matrix of the mutation distribution.

In evolution strategies the latter is achieved with the so-called covariance matrix adaptation.

## References

- [1] N. Hansen and A. Ostermeier, Adapting Arbitrary Normal Mutation Distributions in Evolution Strategies: The Covariance Matrix Adaptation. In *Proceedings of the 1996 IEEE International Conference on Evolutionary Computation*, pages 312-317, 1996
- [2] N. Hansen and A. Ostermeier, Completely Derandomized Self-Adaptation in Evolution Strategies. To appear in *Evolutionary Computation*, Special Issue on Self-Adaptation, 2000.

## Characterization of Landscapes

Ivo Hofacker (joint work with Peter Stadler)

A landscape is a fitness function  $f : V \rightarrow \mathbb{R}$  defined on a set of configurations  $V$  together with some notion of nearness. Typically,  $V$  is a graph with edges connecting the neighbors.

Landscapes can be characterized by their autocorrelation function along random walks. A rapidly decreasing correlation function, i.e. a short correlation length, corresponds to a rugged landscape.

Of particular interest are the “elementary landscapes” of a configuration graph, given by the eigenfunctions of the graph laplacian. For elementary landscapes the correlation function is a simple exponential, with correlation length related to the eigenvalue.

General fitness landscapes can be decomposed into a superposition of elementary landscapes through a Fourier transform. The amplitude spectrum obtained from the Fourier transform contains information about the ruggedness of the landscape and can be used for classification and comparison purposes. Using fast fourier transform techniques explicit computation of the spectrum is possible for landscapes with up to  $10^8$  configurations, for larger problems the dominant amplitudes can be estimated from the correlation function.

The technique can be generalized to landscapes with recombination. For binary strings with recombination the elementary landscapes are exactly the  $p$ -spin functions (Walsh functions), i.e. the same as the for point mutation spaces. Comparing the nearest neighbor correlation, one finds that one-point crossover always leads to a more correlated (less rugged) landscape than uniform crossover, while point-mutations are more correlated than crossover except for very high interaction orders.

## Speciation as Computation

Jeffrey Horn

The formation and interaction of different species in a population  $P$  undergoing selection is a powerful, natural process. To help us understand and control the dynamics of evolving species, it might be useful to model *species selection* as a computation (e.g., optimization of a figure of merit, or the solution of a system of equations). For an example, we analyze *sharing* of different niche resources among competing individuals:  $f_A \equiv$  resource coverage of species **A**,  $f_B \equiv$  resource coverage of species **B**, and so on. Equilibrium is defined as the distribution of the

population at which all individuals' shared fitnesses are equal:  $f_{A,shared} = f_{B,shared}$  or  $\frac{f_A}{n_A} = \frac{f_B}{n_B}$  (where  $n_A$  is the number of **A** individuals). If there is no overlap among the species (a.k.a, *perfect sharing*), it turns out that the equilibrium distribution occurs where the product of individual shared fitnesses is at a maximum:  $\prod_{i \in P} f_{i,shared}$ . Just as selection without sharing can be modeled as a process of monotonically increasing the sum of individual fitnesses,  $\sum_{i \in P} f_i$ , so selection under perfect sharing can be seen as monotonically increasing the *population fitness product*. Unfortunately, once we include overlap among species coverage, the population fitness product fails to model the selection process. However, by increasing the complexity of the equilibrium equations, overlapping niches suggest another computational model of speciation: as the process of numerically solving a set of equations in  $k$  unknowns (where  $k$  is the number of niches):

- For perfect sharing, the equilibrium equations reduce to a single equation that is easily solved.
- For *fitness sharing*, the equilibrium equations result in a system of  $k$  linear equations in  $k$  unknowns.
- For *resource sharing*, the equations yield a system of  $k$  polynomial equations, of order  $(2k - 3)$ , in  $k$  unknowns.

The degree of complexity of these three sharing regimes correlates with experimental observations:

- perfect sharing equilibrium is reached in a single generation, while
- fitness sharing equilibrium takes several generations, but is approached monotonically, and
- resource sharing equilibrium can take much longer to achieve, with “errors” (overshoots and oscillations) along the way.

Such complexity might explain the observed difficulty of achieving the steady-state distribution when many overlapping niches (e.g.,  $10 < k$ ) are present. Furthermore, the oscillating convergence to equilibrium suggests that selection might be using a successive approximation method to solving the set of polynomial equations, much as Newton's method would. Next we can try comparing the successive approximations of our selection operator with the sequence of approximations produced by various known iterative methods.

## Proving Crossover Useful

Thomas Jansen (joint work with Ingo Wegener)

It is a “well-known fact” that crossover is a very useful operator in evolutionary algorithms. It is demonstrated in many experiments and made plausible by analytical investigations based on more or less reasonable assumptions. However, a rigorous proof based on no unproven assumptions is missing. Here, one such proof is presented for a special family of objective functions and a carefully designed steady-state genetic algorithm. It is shown that for these functions the GA outperforms by far a wide class of mutation-based evolutionary algorithms. The result is formulated in terms of expected running time as well as in terms of optimization within a certain number of generations with very high probability.

## How to Detect All Maxima of a Function?

Leila Kallel (joint work with Josselin Garnier)

This paper starts by a theoretical investigation of a family of landscapes characterized by the number of their local optima  $N$  and the distribution of the sizes  $(\alpha_j)$  of their attraction basins. We then propose a practical methodology for identifying these quantities ( $N$  and  $(\alpha_j)$  distribution) for an unknown landscape, given a random sample on that landscape and a local steepest ascent search.

This methodology applies to any landscape specified with a modification operator and provides bounds on search complexity (to detect all local optima) when using the modification operator at hand. Experiments demonstrate the efficiency of this methodology for guiding the choice of modification operators, leading to the design of problem-dependent optimization heuristics.

## Scaling of Program Fitness Spaces

William B. Langdon

We investigate the distribution of fitness of programs concentrating upon those represented as parse trees, particularly how such distributions scale with respect to changes in size of the programs. By using a combination of enumeration and

Monte Carlo sampling on a large number of problems from three very different areas we are lead to suggest, in general, once some minimum size threshold has been exceeded, the distribution of performance is approximately independent of program length.

We proof this for linear programs and for simple side effect free parse trees. We give the density of solutions to the parity problems in program trees composed of XOR building blocks. We have so far only conducted limited experiments with programs including side effects and iteration. These suggest a similar result may also hold for this wider class of programs.

## References

- [1] W. B. Langdon, Scaling of Program Tree Fitness Spaces, *Evolutionary Computation*, 7(4), pages 399-428, 1999.

## Understanding EA: The Fractal Way

Evelyne Lutton

Fractals are largely known as “nice” images (Julia sets, Mandelbrot sets, Von Koch curves, Sierpinski gasket), that present the characteristic of having infinitely many details and that obey a sort of “self-similarity” law. There exists however important (but often less known) aspects of fractals, especially related to their use as analysis tools. The design and use of fractal or multifractal tools in order to perform analyses of signals that are not necessarily “fractal” is now an important trend, and has been proven successful in various domains of applications, such as image analysis, finance, physics, or network traffic analysis ...

Here, we show how these fractal tools can be used in order to carry out some theoretical analysis of Evolutionary Algorithms.

We first present how some irregularity measures (that are one of the basis of fractal and multifractal analysis of signals) can be used in a deception analysis of genetic algorithms (discrete space and infinite population). A qualitative analysis of the influence of some of the GA parameters are derived and a chromosomal encoding evaluation tool is designed.

Then, considering continuous spaces EA, we present preliminary results stemming from perturbation theory, that are a first step towards the introduction of fractal irregularity measures in a Markov model.

Indeed it seems that (fractal) irregularity may be a convenient quantity (yet of course not the only one !) for the classification of fitness functions (or landscapes) with respect to their EA behaviour (related to some extent to EA-difficulty).

# From Genetic Recombination to Search Distributions and Graphical Models

Heinz Mühlenbein

First we show that all genetic algorithms can be approximated by an algorithm which keeps the population in linkage equilibrium, i. e. the genetic population is distributed as a product of univariate marginal distributions. We describe a simple algorithm which keeps the population in linkage equilibrium. It is called the Univariate Marginal Distribution Algorithm (UMDA). Our main result is that UMDA transforms the discrete optimization problem into a continuous one defined by the average fitness  $\tilde{W}(p_1, \dots, p_n)$  as a function of the univariate marginal distributions  $p_i$ . For proportionate selection UMDA performs gradient ascent in the landscape defined by  $W(p)$ . We derive a difference equation for  $p_i$  which has already been proposed by Wright in population genetics. We show that UMDA solves difficult multi modal optimization problem. But for functions with highly correlated variables it has to be extended to marginal and conditional distributions. The Factorized Distribution Algorithm (FDA) uses a general factorization of the distribution. For decomposable functions the optimal factorization can be explicitly computed. In general it has to be computed from the data. Each distribution can be represented as a Bayesian network. Computing the structure from the data is called learning in Bayesian network theory. The problem of finding a minimal structure which explains the data is discussed in detail. It is shown that the Bayesian Information Criterion is a good score for this problem. This is used by the algorithm LFDA.

Part of the material of the talk appeared in *Journal of Heuristics* 5, 215-247 (1999) and *Evolutionary Computation* 7, 353-376 (1999). The talk will be published in *Theoretical Aspects of Evolutionary Computation*, Springer Lecture Series in Natural Computing Systems. See also <http://borneo.gmd.de/AS/art/index.html>.

# Comparing Population Mean Curves

Bart Naudts

A possible approach to qualify how easily a search problem is solved by an evolutionary algorithm is to record statistical aspects of the (average) dynamics of the (algorithm,problem)-combination, and to compare them with the recordings of a large class of well-studied combinations. This talk addresses the issue of comparing population mean ( $\kappa_1$ ) curves arising from problems with different fitness ranges. If one normalizes the curve with respect to the density of states (i.e.,  $h_1 = -\log_2(F(\kappa_1(t)))$ , with  $F$  the cdf of the DoS) then the derivative of this normalized curve can be seen as the speed of the algorithm in bits (of the leading ones problem fixed) per iteration. A first experimental validation of this work in progress suggests that the normalization catches on the aspects influencing the dynamics of evolutionary algorithms: onemax problems are solved at a constant speed from the random population to approximately 90% of the bits equal to one. Randomly generated instances of the 3SAT problem are initially solved at an identical speed independent of the ratio of clauses to variables.

# Optimal Mutation Rates and Error Thresholds in Genetic Algorithms

Gabriela Ochoa

It has been argued that optimal per-locus mutation rates in GAs are proportional to selection pressure and the reciprocal of genotype length. In this paper we suggest that the notion of error threshold, borrowed from molecular evolution, sheds new light favoring this argument. We show empirically the existence of error thresholds in GAs running on a simple abstract landscape; and then investigate a real-world industrial problem, demonstrating comparable phenomena in a practical application. We study the correspondence between error thresholds and optimal mutation rates on these two problems, and explore the effect of different selection pressures. Results suggest that error thresholds and optimal mutation rates are indeed correlated. Moreover, as the selection pressure increases, both error thresholds and optimal mutation rates increase. These findings may have practical consequences, as heuristics for measuring error thresholds in real-world applications will provide useful guidelines for setting optimal mutation rates.

## **Research on the Bayesian Optimization Algorithm or How the BOA Saved the Building Blocks**

Martin Pelikan (joint work with David E. Goldberg)

Recently, a number of evolutionary algorithms that guide the exploration of the search space by building probabilistic models of promising solutions found so far have been proposed. These algorithms have shown to perform very well on a wide variety of problems. The Bayesian optimization algorithm (BOA) used Bayesian networks as a very general probabilistic model to model promising solutions and generate the new ones. This presentation provides motivation to the proposal of the BOA algorithm. It summarizes our recent research on the BOA and outlines the directions our research in this area has been following. It settles the algorithm in the problem decomposition framework used often to understand the complex behavior of genetic algorithms. It provides the most important research issues to tackle and reviews our recent progress in each of these areas. For more detailed information on any of the topics covered in the presentation, please visit our IlliGAL web page at <http://www-illigal.ge.uiuc.edu/> where you can download all our papers. The source code of the BOA algorithm in C++ is also available on the above page.

## **Recursive Conditional Schema Theorem, Convergence and Population Sizing in Genetic Algorithms**

Riccardo Poli

In my talk I have discussed the potential role of theories on schemata in proving the convergence of genetic algorithms in a finite number of generations and with finite populations. In the talk I have presented a version of schema theorem that can be applied recursively, provided that the fitness of population and building blocks are known. At the beginning of my talk I asked how many people believed that schema theorems are pretty useless. More than half of the audience waved their hands. I hope to have convinced at least some of them that this belief is not supported anymore.



# Symmetry Breaking in Evolutionary Algorithms

Adam Pruegel-Bennett

Hard optimization problems often exhibit some symmetries in their solution space. Even when there is no explicit symmetry, the solution space will often be broken up into regions with statistically similar properties. A population exploring such a space will typically span many such regions or symmetric states. Crossover between solutions from different regions can be very costly, because the building blocks have different representations in each state. However this cost may be mitigated by the correlation of the population, since the members are more likely to come from the same region of the search space. In understanding how EAs work on hard problems it is important to investigate how the correlation in the population reflects the underlying structure of the search space. We present some preliminary empirical studies which suggest that the population of an EA can correlate in such a way as to reduce its crossover cost. This suggests that in hard problems some degree of convergence may be beneficial. This may have important implications for how to choose an optimal mutation rate.

## Experiments with Tunable Fitness Landscapes

Colin Reeves

In this talk I first described attempts to measure the difficulty of a problem for an evolutionary search algorithm. Most of these attempts are essentially the same in that they try to infer some measure of *epistasis* from a sample of points in the search space. The standard measure to which most others are related is the *epistasis variance*. Some measures also try to include the effect of the algorithm itself, but I showed that in the case of binary strings they amount to the same decomposition, based on Walsh functions. The question is whether these measures capture any landscape property that really relates to the difficulty of the search.

I discussed some empirical characteristics of NK-landscapes and showed that they are in some senses unusual, both in the distribution and the magnitudes of their Walsh coefficients. I then introduced a new class of tuneable landscapes (the  $\ell, \theta$  landscapes) that can be tuned more precisely. However, I also demonstrated that a whole set of very different landscapes with different apparent ease or difficulty of solution can be formed from a single NK-landscape, all of them having the same epistasis variance. A completely satisfactory interpretation of the meaning of these results is not yet clear, but the discussion afterwards provided several pathways to explore.

# Genetic Algorithms in Time-Dependent Environments

Christopher Ronnewinkel

The influence of time-dependent fitnesses on the infinite population dynamics of simple genetic algorithms (without crossover) is analyzed. Based on general arguments, a schematic phase diagram is constructed that allows one to characterize the asymptotic states in dependence on the mutation rate and the time scale of changes. Furthermore, the notion of regular fitness landscapes is raised for which the population can be shown to converge towards a generalized quasispecies. The case of a generational genetic algorithm with a moving needle-in-the-haystack landscape can be approximately described by a two-concentration model taking into account only the concentration of the current and future needle sequence. Based on this, error thresholds and an optimal mutation rate for most stably tracking the needle can be calculated. The so found phase diagram is fully consistent with the one from general considerations. The two-concentration model also explains the average value of selfadapted mutation rates for the considered moving needle-in-the-haystack. In future work, finite population effects will be included into the analysis, as well as landscapes with broader peaks.

## References

- [1] C. Ronnewinkel, C.O. Wilke and T. Martinetz. Genetic Algorithms in Time-Dependent Environments. In L. Kallel, B. Naudts and A. Rogers (Eds.): *Theoretical Aspects of Evolutionary Computing*. Natural Computing Series, Springer, 2000.
- [2] C.O. Wilke, C. Ronnewinkel and T. Martinetz. Dynamic Fitness Landscapes in the Quasispecies Model. Submitted to *J. Math. Bio.*

## Generalising the Dynamical Systems Model

Jonathan Rowe (joint work with Michael Vose and Alden Wright)

Michael Vose's "dynamical systems" model of GAs utilises the underlying symmetries in the set of fixed-length binary strings to simplify the equations of motion. To generalise this model to arbitrary finite search spaces, we start by explicitly defining a group structure on the search space to capture significant symmetries. Once this is done, suitable crossover and mutation operators can be defined which are invariant with respect to the group. Furthermore, if the search space group has a decomposition into normal subgroups, we can define crossover and mutation operators which respect this decomposition. This leads to a natural generalisa-

tion of the concept "schema" as being cosets of the normal subgroups of the search space. An exact schema theorem can then be proved. This is simplified further if the group is abelian by a Fourier Transform. It can be shown that such a transform exists only if the group is abelian - for example, there is no such transform if the search space is a permutation group (as in the travelling salesman problem).

## **Recent Advances in the Theory of Multi-Objective Evolutionary Algorithms**

Günter Rudolph (joint work with Alexandru Agapie)

Whereas the limit theory of Evolutionary Algorithms (EA) for single-objective optimization is almost exhaustively treated, the situation is completely different in case of multi-objective problems. First results were published in 1998, but the theoretical foundation is hardly developed yet.

Here, we took the following approach: It is assumed that the fitness "values" are only partially ordered. Therefore the target of the evolutionary search is the set of minimal elements. As soon as we find conditions that imply convergence to this set, then this result is also valid in the special cases of multi-criteria and single-criterion problems. For this purpose we defined four abstract base algorithms and postulated certain conditions regarding the Markovian kernel that describes the generation of new trial points depending on the set of current solutions. It is shown that every EA that matches a base algorithm and its associated conditions converges to the desired solution set. Next we split the Markovian kernel into three, say, Markovian kernels that only describe the transitions made by crossover, mutation, and "pre-selection". We identify special properties that must be satisfied for each kernel in order to guarantee that the product of these kernels still exhibits the desired property of the original Markovian kernel. Now it is easy to examine explicit methods for crossover, mutation and pre-selection whether they lead to stochastic kernels with the desired properties or not. It is shown that these conditions are fulfilled for a variety of existing evolutionary "operators".

## Still Missing Features in Current Evolutionary Algorithms

Hans-Paul Schwefel

Nearly four years ago, at the first Dagstuhl seminar on *evolutionary computation and its applications*, nearly the same title was used for the last presentation of the seminar. This offers the opportunity to revisit the findings from the year 1996 and to look for areas of progress as well as areas of stagnation or “premature convergence.”

Firstly, some differences between organic evolution and artificial or simulated evolution will be addressed in binary terms of evolutionary optimization or, better, amelioration, e.g., static versus dynamic, single criterion versus multiple criteria, global knowledge and long term memory versus distributed non-predictive information processing, etc.

Secondly, some emerging ideas of using nature’s “tricks” to survive and evolve will be examined for their feasibility within global, parallel, dynamic, and multi-objective optimization.

Finally, the question will be addressed whether mimicking organic evolution is compulsory or senseless, prodigal or intelligent, or just challenging during the search for understanding the real world.

## Statistical Mechanics of Genetic Algorithms

Jonathan L. Shapiro

Statistical Mechanics can be used to analyse and predict the dynamics of genetic algorithms in finite populations for specific problems. In this approach, the population is described in terms of macroscopics, which are large-scale, statistical properties of the population. If other quantities are required about the population, a maximum entropy distribution is assumed to infer these. This approach works well if the macroscopics used are appropriate for the problem.

The approach is illustrated for the one-max problem under the assumption of linkage equilibrium. Average cumulants of the fitness distribution in the population are used as the macroscopic variables. Taking the diffusion limit produces a linear first order differential equation. The dynamics is dominated by a single slow mode which contributes primarily to low order statistics. For this simple system, optimal annealing rate for the mutation rate can be found. The system can also be studied in a time-varying fitness landscapes.

Moving beyond the linkage equilibrium assumption requires a study of the full difference equations including nonlinearities which vanish only in the linkage equi-

librium limit. Maximum entropy inference is required to determine statistics of the similarity of the strings in the population from the fitness statistics. This similarity determines the efficacy of crossover. Part of the similarity is due to the fact that many strings share a common ancestor; this must be treated separately. The approach has been applied to a range of problems, including multi-modal fitness functions, learning in a perceptron, and the subset sum problem. Different selection models have been considered including truncation selection, tournament selection, ranking selection, and multiplicative selection. Predictions are in very good agreement with simulations.

## **Evolving Neural Structures**

Peter Stagge

Natural evolution does wonderful things in finding structures that survive in changing environments using quasispecies and adaptive structures. Adaptivity is also an inherent property the coding from DNS to structure has.

To evolve structures in practice we actually have to restrict to classes of structures. This choice can be guided by nature: neural fields and neural networks seem to be sufficiently powerful and general. Additionally, we can try to learn from nature as similar structures are found. So we want to learn about the optimizing process by using and comparing it.

Work is presented on noisy fitness functions with different fitness distributions and the connection to selection strength is given exactly in a model. Additionally, redundant coding for network structures is analyzed.

## On the Analysis of a Dynamic (1+1) EA

Ingo Wegener (joint work with Thomas Jansen)

The (1+1)-EA often is used to maximize functions  $f : \{0, 1\}^n \rightarrow \mathbb{R}$ . The typical mutation probability is  $p(n) = 1/n$  which is known to be optimal for linear functions. For certain functions other choices of  $p(n)$  are better. Here a dynamic variant starting with  $p_1(n) = 1/n$  is analyzed. The mutation probability is doubled in each step and values larger than  $1/2$  are replaced by  $1/n$ . This dynamic (1+1)-EA tries approximately all possible values of  $p(n)$  (up to a factor of  $\sqrt{2}$ ). One may hope that one can gain a lot if different mutation probabilities are useful in different regions of the state space. One also may hope that one does not lose too much, since every  $(\log n)$ -th step is close to a good value of  $p(n)$ . With a general technique it is proved that for ONEMAX we are losing at most a factor of  $O(\log n)$  (for the expected run time), for LEADING ONES we are losing exactly a factor of  $\Theta(\log n)$ . For each member of a well-specified class of functions only mutation probabilities of size  $\Theta(\log n/n)$  lead to polynomial run times. The dynamic (1+1)-EA is even better than all static ones. But we also have an example of a function where the dynamic (1+1)-EA is much worse than the usual static one, since the steps with larger mutation probabilities increase the chance to jump into a trap.

## Towards a Theoretical Foundation of Dynamic Optimization

Karsten Weicker

In this work a general mathematical framework is defined for dynamic fitness functions. This enables us to define exact properties of dynamic problems which help to partition all those problems into subclasses. Exemplary properties are regularity, homogeneity, periodicity, and the severity of the dynamics. The properties are inevitable for the comparison of problems. They can also help to characterize the desired behavior of optimization algorithms in dynamic environments and thus to put the term “adaptation” in concrete terms. Results on the correlation between problem properties, algorithmic techniques, and aspects of adaptation can be derived if certain working principles are assumed for the optimization algorithm. In an exemplary result it is shown how adding external memory for previous good solutions and increasing the population size affects the optimization accuracy for a simple unimodal dynamic problem.

## **Towards Qualitative NFL Results**

Nicole Weicker

In this work different kinds of no free lunch results are characterized. A general formalism is presented that shows the common basis of all these results as well as the main differences. Beside the known conservation laws a new kind of no free lunch, the so called qualitative no free lunch is identified. This is a no free lunch, that includes statements about properties of algorithms resp. problems which make it possible to find a matching between the different properties.

## **Models of Steady State Genetic Algorithms**

Alden Wright (joint work with Jonathan Rowe)

Discrete-time dynamical system expected value models for a general steady state genetic algorithm were constructed. These lead to a continuous-time dynamical system infinite population model by a process of letting the population size go to infinity while the time step goes to zero. Conditions were given that imply existence and uniqueness of solutions to this model.

For the random deletion version of the steady state genetic algorithm, the set of fixed points for the continuous-time model, the discrete-time models, and the infinite population model of the corresponding generational genetic algorithm, are all the same. An example was given that showed that a fixed point may be stable for the continuous-time model, but unstable for the generational GA model.

# Computational Time Complexity of Certain Evolutionary Algorithms

Xin Yao

Although evolutionary algorithms have often been used to solve various combinatorial optimisation problems, especially NP-hard problems, it is unclear what advantages they might offer in terms of computational time complexity. Most comparisons between evolutionary algorithms and classical algorithms have been experimental. Few results on the computational complexity of evolutionary algorithms exist. This talk presents two new results on the average time complexity of evolutionary algorithms. They established conditions under which an evolutionary algorithm will take no more than polynomial time (in problem size) to solve a problem and conditions under which an evolutionary algorithm will take at least exponential time (in problem size) to solve a problem. These two results enable us to show the complexity of an evolutionary algorithm for a certain problem. Examples of the average time complexity of an evolutionary algorithm for solving the subset sum problem are given to illustrate the use of these two general results.