

# Adaptive Representations Workshop

## GECCO 2006

Marc Toussaint  
School of Informatics  
University of Edinburgh  
5 Forrest Hill  
Edinburgh EH1 2QL,  
Scotland, UK  
mtoussai@inf.ed.ac.uk

Edwin D. de Jong  
Institute of Information and  
Computing Sciences  
Utrecht University  
PO Box 80.089  
3508 TB Utrecht, The  
Netherlands  
dejong@cs.uu.nl

### 1. INTRODUCTION

Nature has developed an ingenious language to describe organisms: the genetic system. From theory we know that the choice of representation has a crucial influence on the search distribution and the chances to find solutions in a search process. In this view, how can we learn a suitable representation from previous evaluations of samples that will facilitate the search for better solutions? And how can continuous self-adaptation of the representation in evolutionary processes be performed and understood? In this workshop we gather work from different approaches to these questions and initiate a discussion, particularly between people from different theoretical, experimental, or biological backgrounds, aiming at a common framework and language to address such questions.

The contributions are of theoretical as well as experimental nature that investigate how efficient representations can be explicitly learned from data or developed adaptively, e.g. for interesting applications. Also the evolution of the natural genetic language as such is discussed, either from a phenomenological viewpoint, or with regard to the possible underlying mechanisms of this evolution, or in view of theoretical insights in the role of representations in general search processes.

### 2. WORKSHOP ABSTRACTS

#### 2.1 A Class of Statistical Landscapes for Studying the Effect of Uniform Recombination on the Composition of Bitstring Populations

Keki Burjorjee  
*Demo Lab, Computer Science Department, Brandeis Uni-*

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

GECCO'06, July 8–12, 2006, Seattle, Washington, USA.  
Copyright 2006 ACM 1-59593-186-4/06/0007 ...\$5.00.

*versity, Waltham, MA 02454, kekib@cs.brandeis.edu*

We study the behavior of Genetic Algorithms with fitness proportional selection and uniform recombination (fpurGAs) on a class of “statistical” landscapes called Recursive Low-Order Singularly-Mean-Separable Gaussian Landscapes. With a mixture of theory and visually demonstrative experiments We describe the role that uniform recombination plays in “statistical” optimization over such landscapes. We show that uniform recombination along with fitness proportional selection implicitly implements a process called Recursive Selective Fixation (RSF) that drives this type of optimization. To describe Recursive Selective Fixation in this abstract we introduce the following terminology: the fitness of a schema in some generation is the average fitness of all genotypes that belong to that schema in that generation. A schema that is a subset of some other schema is called a subschema of the containing schema. When we talk about the relative order of some subschema we mean the number of defined locii of the subschema that are ‘wildcards’ in the containing schema. Recursive Selective Fixation can be thought of as an iterative process. In the first iteration it becomes sensitive to some schema partition which contains low order schemata with fitness values that remain statistically significantly higher, over several generations, than the average fitness of all schemata in the partition. We call this schema partition the pivot partition of the iteration, and call the schemata within it the pivot schemata of the iteration. In each iteration RSF minimizes the marginal entropy of the population over the iteration’s pivot partition by sending one of the iteration’s pivot schemata to fixation (i.e. RSF drives all genotypes in the evolving population into that pivot schema). Moreover RSF achieves this entropy minimization over the pivot schema partition while keeping the marginal entropies of the population over the non-pivot schema partitions high. After each iteration Recursive Selective Fixation becomes sensitive to some new partition. This new partition contains subschemata (of the just-fixed schema) of low relative-order that have fitness values that are statistically significantly higher, over several generations, than the average fitness of all subschemata in the new partition. The new partition becomes the new pivot partition of the next iteration, and the subschemata become the new pivot schemata of the next iteration. When some subschema of low relative order gets fixed in the population

(i.e. when all the genotypes of all present and future populations belong to some subschema) then that subschema can be thought of as “being part of the representation”. Thus Recursive Selective Fixation recursively “adapts” representation when it recursively fixes subschemata of low relative order.

## 2.2 Improving Scalability: Defining & Enabling Structural Organization

Gregory Hornby  
*QSS Group Inc., Mail Stop 269-3, NASA Ames Research Center, CA 94035-1000, hornby@email.arc.nasa.gov*

For computer-automated design systems to scale to more sophisticated designs they must be able to produce designs with greater structural organization than previously achieved. Here we claim that structural organization is comprised of the characteristics of modularity, regularity and hierarchy (MR&H) – characteristics that are found both in man-made and natural designs – and that these characteristics are enabled by implementing the attributes of combination, control-flow and abstraction in the representation. To support this claim we define metrics for measuring the three components of structural organization, MR&H, and then use an evolutionary algorithm to evolve solutions to different sizes of a table design problem using five different representations, each with different combinations of modularity, regularity and hierarchy enabled and show that the best designs are achieved when all three of these attributes are enabled. In addition we find that high fitness values are achieved with high values of modularity, regularity and hierarchy and that there is a positive correlation between increases in fitness and increases in the measured values of modularity, regularity and hierarchy.

## 2.3 On-Line Adaptation of First-Order Heuristics

Cezary Z. Janikow  
*University of Missouri – St. Louis, CCB319, UMSL Saint Louis, MO 63367, 314-516-6352, janikowc@umsl.edu*

Genetic Programming uses trees to represent chromosomes. The user defines the representation space by defining the set of functions and terminals to label the nodes in the trees. This space is usually much larger than the solution space, with many redundancies or invalid subspaces. Structure-preserving crossover, STGP, CGP, and CFG-based GP give the user the power to reduce the space by specifying syntactic rules for valid tree construction. CGP also works with weak constraints, that is probabilistic heuristics, either on syntactic structure or on overloaded types. All such CGP constraints are first-order, that is on parent-child only. The strong constraints reduce the overall search space, while the weak heuristics affect the effective density of the remaining search space. In 2004, we introduced the ACGP methodology to adapt the heuristics during GP evolutionary search. In that methodology, GP evolution was iterated a number of times through standard generations, with the heuristics extracted at the end of each iteration and fed back into the next iteration. This was the off-line approach, which showed great potentials for large improvements in the quality and speed of generated solutions. Here we present a new methodology which adapts the heuristics during the evolution – the on-line approach. As the heuristics are learned, they imme-

diately affect the effective search space, pushing the search toward regions in which the evolution progresses better (better can be defined as faster, smaller solutions, or a mix of those). This paper points the limitations of standard GP when it comes to utilizing information gathered during evolution, and shows how the ACGP methodology utilizes such information to effectively narrow down the search space by adapting the GP representation during its evolution. The paper also looks at the cost-benefit analysis of learning such heuristics and using them vs. standard GP search without any extra cost but also without the benefits.

## 2.4 Geometric Crossover for Biological Sequences

Alberto Moraglio<sup>1</sup>, Riccardo Poli<sup>1</sup>, Rolv Seehuus<sup>2</sup>  
<sup>1</sup>*University of Essex, Wivenhoe Park, Colchester, CO4 3SQ, UK, {amoragn, rpoli}@essex.ac.uk*  
<sup>2</sup>*Norwegian University of Science and Technology, 7042 Trondheim, NORWAY, rolv.seehuus@idi.ntnu.no*

This paper extends a geometric framework for interpreting crossover and mutation to the case of sequences. This representation is important because it is the link between artificial evolution and biological evolution. We define and theoretically study geometric crossover for sequences under edit distance and show its intimate connection with the biological notion of sequence homology.

## 2.5 Coevolution and Selection for Evolvability

Joseph Reisinger, Risto Miikkulainen  
*University of Texas at Austin, 1 University Station, C0500, Austin TX, 78712-0233, {joeraii,risto}@cs.utexas.edu*

Choice of representation affects the underlying evolvability of genetic algorithms (GAs). The problem of representation design is well known in Machine Learning, though the state of the art still requires the algorithm designer to employ domain knowledge and hand-tailor a solution representation for each specific problem. Adaptive representations, on the other hand, address the problem of representation design by allowing the algorithm to search simultaneously through the space of representations and solutions. In nature, genotype-phenotype maps tend to be highly evolvable; although mutations made to the genotype are essentially random, the resulting phenotypic variation is highly structured and furthermore this variation is directly under genetic control. Thus the genotype-phenotype mapping can be considered to function as an adaptive representation, providing a mechanism through which artificial evolutionary systems can exhibit evolvability. Evolvability helps find good “trajectories” through the search space, not just good fitness peaks. However, selection for evolvability is not always possible with many fitness functions; in general evolution is opportunistic, and will take large immediate fitness gains over smaller fitness gains that may lead to better parts of the search space. Thus selecting for good representations and selecting for good solutions can be viewed as antagonistic goals. In other words, even given an adaptive representation, there may be no real selection pressure to optimize the genomic representation. This observation may help explain why in some cases indirect encodings do not perform well when compared to simpler direct encodings. One way in which a consistent selection pressure for evolvability can be generated is by systematically changing the fitness func-

tion over time, naturally implicating co-evolution as a possible mechanism through which evolvable representations can arise. However if the representation itself is not adaptable, as in the case of direct encodings, this selection pressure is ignored, leading to highly-optimal, but “brittle” solutions. We believe that this phenomenon may be responsible for much of the cycling and disengagement behavior seen in competitive coevolution; since direct encodings cannot store information about search, populations may be prone to “forgetting” past strategies if they are no longer required to beat the opponent population.

## 2.6 Geometric Crossover for Supervised Motif Discovery

Rolv Seehuus<sup>1</sup>, Alberto Moraglio<sup>2</sup>

<sup>1</sup>*Norwegian University of Science and Technology, 7042 Trondheim, NORWAY, rolv.seehuus@idi.ntnu.no*

<sup>2</sup>*University of Essex, Wivenhoe Park, Colchester, CO4 3SQ, UK, amoragn@essex.ac.uk*

Motif discovery is a general and important problem in bioinformatics, as motifs often are used to infer biologically important sites in bio-molecular sequences. Many problems in bioinformatics are naturally cast in terms of sequences, and distance measures for sequences derived from edit distance is fundamental in bioinformatics. Geometric Crossover is a representation-independent definition of crossover based on a distance on the solution space. Using a distance measure that is tailored to the problem at hand allows the design of crossovers that embed problem knowledge in the search. In this paper we apply this theoretically motivated operator to motif discovery in protein sequences and report encouraging experimental results.