

Chapter 1

EC THEORY - “IN THEORY”

Towards a Unification of Evolutionary Computation Theory

Christopher R. Stephens
Instituto de Ciencias Nucleares
UNAM, Mexico
stephens@nuclecu.unam.mx

Riccardo Poli
Department of Computer Science
University of Essex, UK
rpoli@essex.ac.uk

Abstract We present a personal overview of EC theory. In particular, we try to show that recent theoretical developments have pointed the way to a grand unification of different branches of EC, such as Genetic Algorithms and Genetic Programming, and also different theoretical models, such as the Vose model and Holland's Schema theorem. We give a broad outline of this unification program showing how the different elements above are related to each other via changes of representation on the space of EC models. Based on our work we pose a series of challenges which if met, we believe, will lead to a much deeper understanding of EC and the various types of evolutionary algorithm.

Keywords: Schema theory, Vose model, Unification, Genetic Algorithms, Genetic Programming, Evolutionary strategies, Building Blocks

Introduction

Relatively speaking, Evolutionary Computation (EC) is a fairly immature subject. It exhibits many different facets without a high degree

of intellectual consensus. It sometimes seems that it is all things to all people. It is a subject that is principally empirical and phenomenological. Moreover, it is empirical and phenomenological within a very ill defined framework, in distinction to the world as seen through the eyes of physics and biology. This is because in EC the only real limit to what can be studied is the imagination. In its more “scientific” guise it is related to an older, more mature field - population genetics - but without the constraints that nature imposes and without the same degree of intellectual coherence that comes with maturity. Mathematically speaking it is the study of certain classes of heuristic algorithms based on populations of objects (Vose, 1999), though as we shall see what these classes are is far from clear. Seen from the “engineering” point of view it is an area where the analogical use of “natural selection” appears as a moulding force in the creation of more “competent” problem-solvers (Goldberg, 2002).

One salient characteristic of EC theory is that it is difficult. It is also very exciting. It is relatively simple to write an evolutionary algorithm (EA). It is exceedingly difficult to understand its behaviour other than at a superficial level. Even fairly simple EAs, such as a Genetic Algorithm (GA) with selection and mutation only, present formidable difficulties.¹ This stark contrast between the ease with which an EA can be written and the complexity of understanding its behaviour leads to a very large expectation gap between EC “practitioners”, who focus on the empirical aspects, and often seem capable of thinking of five new genetic operators before breakfast, and the theorists who, to the practitioners, seem fixated on no selection or “counting ones”.

Unlike more mature areas of science there is not even a clear consensus on what should be the task of EC theory. Is it to provide recipes for practitioners, to provide exact computational models, to allow a deeper understanding of a complex system, all of these, none of these, or what? Having established what theory should do, it is then important to ask ourselves - “What has it done?” and “Where is it going?” A goal of this article is to give, albeit briefly, our personal view on this.

Different approaches to EC theory have been proposed in the past. These include schema theories (Holland, 1975), the Vose model (Nix and Vose, 1992), the statistical mechanics approach (Prügel-Bennett and Shapiro, 1994) and more. Is there a model that is superior to all others? Often, models are judged by their clarity, simplicity, and ability to explain and predict. Is there a framework that does this best? Once we have established what the task of EC theory should be, it will be easier to answer these questions.

A theoretical model is also often judged by how well it unifies a range of phenomena. As there are many different flavours of EA - GAs, Genetic Programming (GP), Evolution Strategies (ES) etc. - one may ask if there is a theoretical framework that encompasses them all? If not, then which is the framework with the broadest applicability?

The framework with the broadest applicability is inhomogeneous Markov chain theory. However, describing EC as a subset of such a theory means very little. So, what are the essential elements common to different EAs? These are: a choice of genotype-phenotype map, a choice of fitness function and a set of evolution operators. Here, our first objective is to present a unified theoretical framework applicable to virtually any type of fitness function, any type of genotype-phenotype map, any type of selection and any type of mutation and crossover.² Our second objective is to demonstrate that all current and past theoretical models of EAs are in fact simply mathematical transformations of one another.

In no way do we want to give the impression that we have totally achieved these objectives. Rather, we are indicating in which direction we believe EC theory should move and what we see as the principal challenges ahead.

The Rôle of Theory in EC

We begin by asking - what should be the task of EC theory? Is it reasonable, for instance, to think that a theoretician should be able to deduce from first principles after exactly how many generations there's a 95% chance that a better optimal individual will not be found in the following 50 generations for a particular 555-job job-shop scheduling problem with three point crossover with probability 0.9 and mutation probability 0.02? We would categorically deny that this is the principal task of the theory. In fact, we will probably never be able to answer questions such as this. Equally, we may ask if it is the exclusive task of EC theory to consider only very general, global results such as the "No Free Lunch" theorem. Once again, we would say no, rather, theory is at its most powerful when between the very detailed and the very general - but just where?

In the EC community there is a strong distinction between "scientific" theory and "engineering" theory (Goldberg, 2002). These differ both in terms of methodology and motivation. The role of theory in science is to explain and understand phenomena (often results of controlled experiments) within a formal, well defined framework. The role of theory in engineering is to "build better bridges". Sometimes the theory used is rooted in an underlying scientific theory, but often uses rules of thumb

that are far removed from the scientific roots. We emphasize that it is not a question of which is the superior approach. They address very different concerns. Here, though, we will be very much more concerned with EC theory from the scientific perspective with, however, one eye always on the engineering point of view wherein we may also examine things from a practitioner's standpoint.

In many sciences a large part of theory is associated with taxonomy - classification with respect to natural relationships. In EC, various high-level taxonomic labels are at our disposal, such as GP, GAs, ES etc. Whether these labels are optimal, or even useful other than in an historic sense, however, is a debatable point, as we shall see. Taxonomy allows us to understand commonality between different things. Subsequently we must understand why such commonality exists. For instance, the periodic table was initially an empirical and phenomenological construct until the atomic theory gave it a firm "microscopic" foundation. What is the "periodic table" for EC? Does such a construct exist? If nothing of this nature existed it would be deeply worrying as it would mean that a theoretical treatment of each and every EA and/or problem would be different. It is clear however that there is commonality. The question is more - can it be suitably formalized?

At the other extreme one could claim a type of "hyperuniversality", such as was present in the original version of the Building Block Hypothesis (Goldberg, 1989c; Grefenstette, 1993), which claimed that all GAs behaved in the same way in finding an optimum - via fit, short schemata. We now know that this, in its strict form, is wrong, being rather an engineering rule-of-thumb with only limited validity, and that such a degree of hyperuniversality does not exist. Nevertheless, a prime job of EC theory should be to tell us what EAs and problems, or classes of EAs and problems, are likely to lead to similar outcomes or behaviour. It does not need to be elaborated on that a deeper understanding of this would be of great use to practitioners.

Passing beyond the taxonomic component of EC theory we should also ask that the theory be able to predict, at least within some more or less approximate scheme, the dynamical evolution of an EA. To address this one needs to start with a framework that at least formally captures the behaviour of an EA. This can be at the level of a theory or model which is exact or approximate from the outset. All else being equal an exact model is preferable. Great progress has been made in the last decade in exact formulations of EA dynamics. For instance, the work of Michael Vose and collaborators (Vose, 1999; Nix and Vose, 1992) in the context of the simple GA, where the transition probability matrix for the population evolution is iterated as for a Markov chain,

and the work of Stephens, Poli and collaborators (Stephens and Waelbroeck, 1997; Stephens and Waelbroeck, 1999; Stephens, 2001; Poli and McPhee, 2001a; Poli, 2000a; Poli, 2001a), where an exact dynamics is modelled in terms of schemata, thus leading to a generalized and exact form of Holland's original Schema theorem (Holland, 1975), are two such approaches.

Beyond the mathematical representation of EC theory one should also require that the theory give some intuitive framework within which an EA, or class of EAs, can be understood. The concept of a fitness landscape from population biology (Wright, 1932; Wright, 1967; Reidys and Stadler, 2002) is a prime example of a construct that offers a framework to do just that. The original Schema theorem of Holland and associated Building Block Hypothesis are another very important example. In the seventies and eighties, and to a lesser extent later, they, in fact, seemed to provide a perfectly valid and sufficient theoretical foundation for GAs. So much so that, in the early to late nineties, developing a schema theorem like Holland's became the target for GP theorists too.³

Finally, it would be useful to better understand the relationship of EC theory to other more well-established areas in computer science, mathematics, biology and physics. This, for example, would allow us to know whether what has been done in EC is novel. More generally it would make it possible to remove barriers between disciplines and allow for an easier exchange of ideas and results.

Having established some criteria by which we may judge a theory to be "good" or not we may ask: Out of the many possible approaches to EC theory and motivations to develop it, under what circumstances is one better than another, or is there one which is superior to all others under all these rôles? We will provide our answer to this in the rest of the chapter.

EC theory - the "bare necessities"

In this section we wish to give a brief, non-rigorous exposition of the fundamentals of EC theory, as we see them, that essentially could be applied to any EA. Thus, we try to maintain as much generality as possible, in particular to show how a unified theoretical framework, encompassing most, if not all, standard EAs, can be developed. Formally, an EA is an algorithm that takes as input a population of "objects" (strings, trees etc.⁴) and a fitness function, at a given time, and gives as output the population at a later time. Canonically the evolution is a Markov process generated by a set of genetic operators that act stochastically. The fact that an EA is a stochastic process has the important ramification

that we may only expect to make statistical predictions in terms of the dynamics.

The objects live on a configuration⁵ space X , of dimensionality N_X , with elements $i \in X$, where the index $i \in \{1, 2, \dots, N_X\}$. As EAs are population based one needs to consider sets of elements, some of which may be repeated multiply. Thus, we denote a population by $\mathbf{P} = (n_1, n_2, \dots, n_{N_X})$, where n_i represents the proportion of objects of type i in the population. Each object is assigned a quality or fitness, via a fitness function $f_X : X \rightarrow R^+$. This notion leads one to the important concept of a fitness landscape, \mathcal{F} , where one thinks of a topography wherein f_X represents a height function “above” the space X . The intuition behind the landscape concept is that populations will seek the peaks in the landscape and move away from the valleys. Often, always in the case of population biology, the fitness landscape possesses a degeneracy, i.e. many genotypes have the same fitness (i.e. corresponding to the same phenotype). One can speak of a “symmetry”, strictly speaking an equivalence relation, in this case and ask if this symmetry is preserved by genetic operators other than selection. A concrete example of this is the equivalence under selection of those genotypes that correspond to phenotypes of the same fitness.

A dynamics is imposed via an evolution operator, \mathcal{H} , such that in the infinite population limit, where $\mathbf{P}(t)$ is the probability distribution at time t , $\mathbf{P}(t+1) = \mathcal{H}\mathbf{P}(t)$. The specific form of \mathcal{H} depends on the specific set of genetic operators used, which in their turn depend on families of parameters. We will generically focus on the standard ones: selection, mutation and recombination. Selection is an operator that depends on the fitness values of the objects. The number of parameters necessary depends on the type of fitness function and the amount of degeneracy of f_X . For instance, for a counting-ones GA problem only N fitness values are needed, while for the Eigen model (Eigen et al., 1989) (“needle-in-a-haystack” fitness function) only two, in both cases the genotype-phenotype map being highly degenerate. Mutation, a one-body operator, usually only depends on one parameter - the mutation probability - that is applied uniformly to each locus, though more general operators can easily be considered. Two-parent recombination generically depends on the set of recombination distributions, $\{\lambda_{ijk}(m)\}$, that characterize the transferral of genetic material from parents to offspring, where $\lambda_{ijk}(m)$ is the probability to form an offspring object, i , given two parent objects, j and k , and a crossover “mode”, m , i.e. a rule for redistributing genetic material between parent and offspring objects. The complexity inherent in this representation can be appreciated by writing

down the exact string evolution equation for the simple case of three-bit strings as in (Whitley, 1992).

We mentioned previously that taxonomy is important without being specific as to what exactly should be classified. One may think that EAs themselves should be classified. An EA alone however, is in some sense a "black box" which takes a "problem" (usually a fitness landscape and an initial population) as input and then gives an output (the population at a later time). A given EA, though, may have very different characteristics with respect to a given measure on one problem versus another. Another way to see this is that an EA does not fully specify the dynamics of the system, whereas an EA and a problem together do. Hence, we are led to consider a taxonomy of EA/problem pairs. We will call an EA/problem pair a "model". In this context we may characterize a particular model, α , by a set of parameters $\{\{f_\alpha\}, \{\mu_\alpha\}, \{\lambda_\alpha\}\}$, where $\{f_\alpha\}$ represents the fitness landscape and selection mechanism, $\{\mu_\alpha\}$ mutation and $\{\lambda_\alpha\}$ recombination. With these three in hand we can specify a very large class of models. We will denote this space of models, \mathcal{E} . We believe that a better understanding of the taxonomy of EAs and fitness landscapes can be achieved by studying \mathcal{E} . In principle one could put a metric on \mathcal{E} and talk about how close one model is to another. A less rigorous, but more pragmatic, approach is to think of two models as being "close" if they lead to "similar" behaviour. Of course, to do this one must define "similarity measures". At any rate, continuity on \mathcal{E} would lead one to believe that models with similar parameter values should behave similarly, except, of course, in the neighborhood of a singularity.

As a simple example of this approach, consider a GA without mutation and selection but with m -point crossover acting on N -bit strings. In this case we can think of \mathcal{E} as containing only m distinct models. If we chose as similarity measure, $t_c(m)$, the number of generations needed for the correlation function $\langle a_x a_y \rangle$ to decrease by a factor c , where $\langle \ \rangle$ denotes population average and a_x and a_y are the allele values at loci x and y , then one would find, for example for $N = 20$, that $t_c(1) < t_c(2) < \dots < t_c(N - 1) < t_c(N)$. With the specific values one may determine, for example, that 2-point crossover is closer to 1-point crossover than 15-point crossover.

One can think of population flows as taking place on X , the configuration space, or on \mathcal{F} , the fitness landscape. All the main branches of EC - GP, GAs, ES etc. - fall into this general framework. The chief differences lie more in what objects are being represented in X and what specific operators constitute \mathcal{H} . For instance, in GAs the i represent fixed length strings. In GP they are program trees and in machine code GP (Nordin and Banzhaf, 1995a; Nordin, 1997) or Grammatical evo-

lution (O'Neill and Ryan, 2001) they are variable length strings. We shall also see that “coarse grained” representations, such as schemata, or particular sets of schemata - Building Block Schemata - also offer very useful basis representations. Interestingly, in nature, genotypes are variable length due to phenomena such as gene duplication and deletion. Additionally, a non-linear structure can also be more appropriate, when for instance modelling protein secondary or tertiary structure.

Rather than considering one type of basis as being more “general” or fundamental than another it is useful to think of passing between different basis representations via coordinate transformations on X , via embeddings of X in a larger or higher dimensional space, or, in the case of true coarse grainings, via projections.

Probably the best known alternative coordinate system is the Walsh basis (Goldberg, 1989a; Goldberg, 1989b). Another example of a coordinate transformation, whose importance and utility we will examine shortly, is the following: take fixed length binary strings of length N . In this case X is the N -dimensional Boolean hypercube, the N string loci forming a complete orthonormal basis for the hypercube. Now change to an alternative basis, which we term the Building Block Basis (BBB) (Stephens, 2003), which consists of all schemata corresponding to a given string, where the choice of string is arbitrary. Formally, $i = \sum_j \Lambda_{ij} j$, where j are strings, $\Lambda_{ij} = 1$ iff j is a member of Building Block i and is zero otherwise. The coordinate transformation engendered by Λ yields a basis which is not orthonormal. We will consider the BBB more extensively later.

An example of an embedding transformation, at least in principle, would be that of passing from variable length strings of up to maximum size N_m with binary alleles to a fixed length basis representation of size N_m by including a third allele value that specifies that there was no corresponding bit in the variable length case. Of course, for these more general transformations development of the operators and the corresponding theory necessary to maintain syntactic correctness of the offspring is a totally open issue. In this case, one might be better off using the theory for variable length structures already developed in GP.⁶ Finally, a simple projective coarse graining would be that of passing between genotype and phenotype.

The above types of map give us flexibility in terms of what particular representation we may find most suitable for a problem and also give a more unified framework within which we may view different elements of EC, such as GP and GAs, in a more coherent light. In fact, our lack of understanding and consideration of such transformations is one of the reasons why EC theory has been, and continues to be, fragmentary.

An even more important reason for considering general classes of transformation associated with X is that it facilitates an understanding of the dynamical equations associated with the true effective degrees of freedom of the model. These effective degrees of freedom will more often than not be aggregations of the underlying "microscopic" degrees of freedom and may be made more manifest via a coordinate transformation, embedding or coarse-graining/projection. Additionally, it may be the case that effective degrees of freedom most naturally emerge in an approximation to the dynamics rather than the exact dynamics.

As the model dynamics moves a population composed of individual objects around in X an important precondition for understanding the dynamics is a notion, \mathcal{X} , of neighborhood, nearness, distance, or accessibility on X . In some settings, such as binary GAs, a natural neighbourhood relation is associated with the Hamming metric. In more complicated cases, such as GP, where one requires a metric on the space of trees of variable size and shape, this is a much more subtle question. Additionally, different genetic operators are often most naturally associated with different notions of nearness. For instance, mutation is very naturally associated with Hamming distance. This is not the case for recombination however. One may be led in this way to consider a different metric for every operator (Jones, 1995). However, given that the dynamics of the model is due to a single composition of different genetic operators, it is questionable as to what extent this picture is useful.

Generic Genetic Dynamics

The space of models, \mathcal{E} , is of very high dimensionality if one thinks of all possible genetic operators. Selection, mutation and recombination form a very important subset and we will now restrict attention to them. For transparency we will also consider the dynamics in the infinite population limit, writing evolution equations for the probability distribution of objects, $\{P_i(t)\}$, where $P_i(t)$ is the probability for finding object i at time t . The extension to finite populations is relatively straightforward.

Formally at least, the following also applies to GP as well as GAs:

$$P_i(t+1) = \sum_j \mathcal{P}_{ij} P_j^c(t) \quad (1.1)$$

where $P_i^c(t)$ is the probability to find objects of type i after selection and crossover. The matrix elements of the mutation matrix, \mathcal{P} , give the probability to mutate object j to object i . In the simple case of fixed length GAs for instance, $\mathcal{P}_{ij} = p_m^{d^H(i,j)} (1-p_m)^{N-d^H(i,j)}$, where d_{ij}^H is the Hamming distance between the two strings and N is the strings'

length. For mutation Hamming distance is clearly a very natural metric. Explicitly $P_i^c(t)$ is given by

$$P_i^c(t) = (1 - p_c)P_i'(t) + \sum_m \sum_j \sum_k \lambda_{ijk}(m)P_j'(t)P_k'(t) \quad (1.2)$$

where $P_i'(t)$ is the probability to select i . In the case of proportional selection $P_i' = (f_i/\bar{f}(t))P_i$, where \bar{f} is the average population fitness. $\lambda_{ijk}(m)$ is an interaction term between objects, i.e. objects j and k are selected and crossed over (“interact”) to potentially form an object i . $\lambda_{ijk}(m)$ depends not only on the objects j and k but also on the particular recombination mode. In the case of homologous crossover the recombination modes are just crossover masks with \sum_m being the sum over all possible recombination masks. In the case of non-homologous crossover the modes are more general than masks.

Equations (1.1) and (1.2), as an exact representation of the dynamics in terms of incidence vectors for objects, in the case of fixed-length GAs, where a crossover mode is simply a mask, are equivalent to the Vose model or, indeed, to earlier formulations in population biology (see (Bürger, 2000) and references therein). It looks quite different because we are using a less condensed notation in order to bring the different roles that each operator play to the fore. These equations however are also valid for objects other than fixed-length strings. A particular criticism of the Vose model has been that although elegant it looks all but hopeless to get other than very general information from the equations. Also, the equations are far removed from older elements of GA theory such as the Schema theorem and Building Block Hypothesis. This has led proponents of the Vose approach to question both the validity and the utility of the latter. As the above equations are equivalent to the Vose equations we may understand the enormity of the task of trying to obtain either quantitative or qualitative results from them. They represent N_X coupled, simultaneous non-linear difference equations. At the level of mutation and selection the problem is linear hence, conceptually at least, the problem is easily addressed - one must find the eigenvalues and eigenvectors of the mutation/selection matrix. The introduction of recombination at first sight leads to a degree of complexity far beyond that of selection and mutation.

To write the interaction constants more explicitly we would have to consider a more definite model. However, we may make some generic comments. First of all, $\lambda_{ijk}(m) = 0$ unless the mode m creates object i from j and k . Generically, this is very unlikely and hence the vast majority of interactions are zero. For instance, in GAs with binary alleles for a given i and m , $\lambda_{ijk}(m)$ is a 2^N -dimensional square matrix.

However, only of the order of 2^N matrix elements are non-zero. Thus, the microscopic representation is very inefficient, there being very few ways of creating a given target by recombination of strings. The vast majority of string recombination events are neutral in that they lead to no non-trivial interaction. These comments also hold for more complicated types of object.

Understanding Genetic Dynamics: Fitness, is it “Effective”?

Having written down a generic dynamics how do we understand it? In population biology the concept of a fitness landscape has played an important role. Standard intuition views a fitness landscape as a rugged terrain where populations flow towards fitness peaks. Thus, natural selection can be viewed as a type of “hill climbing” on this topography.

The classical fitness concept, and associated fitness landscape, however, do not take into account the important effect the mixing genetic operators may have in determining the complete reproductive success of an individual. In particular, the effect of these genetic operators can be such that population flows on the standard fitness landscape cannot be understood with any degree of intuition. In fact, the flows can be quite counterintuitive, leading to situations where populations flow *against* the fitness gradient. A simple concrete example is, once again, the Eigen model where the fitness landscape is just one isolated fitness peak in an otherwise flat landscape. In the absence of mutation the entire population will eventually climb to the top of the fitness peak. In the presence of mutation the proportion of the population associated with the peak is less than one. However, above a certain critical mutation rate (Eigen et al., 1989), p_{crit} , the peak proportion is what it would be in a completely random population on a flat fitness landscape. The landscape remains the same, i.e. with a single peak, yet selection does not act, in the sense that there is no preference for the peak. Clearly hill climbing is not a very useful analogy here.

The mixing operators can also lead to directed flows on neutral networks due to an “induced” breaking of the genotype-phenotype symmetry (Stephens, 1999a; Angeles et al., 1998; Stephens et al., 1998; Mora et al., 1999). Such phenomenon, unlike the case of population flow due to positive selection cannot be naturally understood in terms of hill climbing on a standard fitness landscape either. However, all these phenomena can be intuitively understood within the framework of a different paradigm - *effective fitness*⁷ (Stephens and Waelbroeck, 1998; Stephens, 1999b; Stephens and Vargas, 2000; Stephens and Vargas, 2001; Poli,

2000a; Stadler and Stephens, 2003), albeit with the consequence that effective fitness is not a constant quantity but rather depends on the state of the entire system and hence is intrinsically time dependent.

We define the effective fitness in the case of objects as

$$P_i(t+1) = \frac{f_i^{\text{eff}}(t)}{\bar{f}(t)} P_i(t) \quad (1.3)$$

One may think of the effective fitness as representing the effect of all genetic operators in a single reproductive selection factor. Here, we have assumed proportional selection. Effective fitness can easily be generalized for other selection mechanisms however. $f_i^{\text{eff}}(t)$ is the fitness value at time t required to increase or decrease $P_i(t)$ by pure reproductive selection by the same amount as all the genetic operators combined in the context of a reproductive fitness f_i . If $f_i^{\text{eff}}(t) > f_i(t)$ the effect of the genetic operators other than selection is to enhance the reproductive success of object i . Obviously, the converse is true when $f_i^{\text{eff}}(t) < f_i(t)$.

The exact functional form of the effective fitness obviously depends on the set of genetic operators involved. For the fairly general case of equation (1.1) we have

$$f_i^{\text{eff}}(t) = \frac{\bar{f}(t)}{P_i(t)} \sum_j \mathcal{P}_{ij} P_j^c(t) \quad (1.4)$$

In the limit $p_m \rightarrow 0$, $\lambda_{ijk} \rightarrow 0$ (or in more general circumstances when the strengths of operators other than reproductive selection $\rightarrow 0$) $f_i^{\text{eff}}(t) \rightarrow f_i$.

The key element behind effective fitness, irrespective of its mathematical definition, is that population flows in the presence of operators other than pure reproductive selection are much more readily understood in terms of it. In fact, to go further, even in the case of pure reproductive selection, if one performs any sort of coarse graining and considers schemata rather than strings, then population flow is more readily understood in terms of an effective fitness landscape rather than the reproductive one. As an example, for the evolution of a particular order-1 schema in a population of N -bit strings it is more natural to consider the *time dependent* one-dimensional landscape associated with the schema than the collective dynamics of the $2^{(N-1)}$ string types that go up to make the 1-schema. The job of evolution at the end of the day is to produce fit offspring which in their turn produce fit offspring which in their turn... It is no use having an individual with high reproductive fitness that is associated with a high probability to mutate to a very low fitness individual.

In the case of the Eigen model the effective fitness of the master sequence or needle, under selection and mutation only, is

$$f_{\text{needle}}^{\text{eff}}(t) = f_{\text{needle}}(1 - p_m)^N + f_{\text{hay}} \sum_{i \neq \text{needle}} \frac{P_i(t)}{P_{\text{needle}}(t)} p_m^{d_{\text{needle},i}^H} (1 - p_m)^{N - d_{\text{needle},i}^H} \quad (1.5)$$

where f_{needle} and f_{hay} are the fitnesses of the needle and the “hay” respectively. In the limit $p_m \rightarrow p_{cr}$, $f_{\text{needle}}^{\text{eff}}(t) \rightarrow \bar{f}(t)$ and we see that the effective fitness landscape becomes flat thereby giving an intuitive explanation for the behaviour in the vicinity of the critical mutation rate. We can thus think of evolution as a hill-climbing process on an effective fitness landscape (which is time dependent). In this model mutation breaks the genotype-phenotype symmetry among the non-needle strings in such a way that those strings that are closer in Hamming distance to the needle have more reproductive success. Once again, this cannot be understood in terms of the fitness landscape as it is flat for the non-needle strings. The analog of equation (1.5) for non-needle strings shows us however that the effective fitness of strings that are close to the needle is higher than that of distant strings. Effective fitness in this sense is a direct measure of the strength of the breaking of the genotype-phenotype symmetry and hence offers both a qualitative and quantitative framework within which phenomena such as GP bloat and evolutionary robustness may be understood.

Understanding Genetic Dynamics: What are the right effective degrees of freedom?

All genetic operators affect what are the appropriate effective degrees of freedom⁸ for a particular model, although in potentially very different ways. For selection, almost by definition, the principal effective degree of freedom is the phenotype. For pure mutation they are the eigenvectors of the mutation matrix, the most relevant ones being those with the largest eigenvalues. When combining selection and mutation it becomes much more difficult to determine the correct effective degrees of freedom. As a simple example, consider again the needle-in-a-haystack landscape for N -bit strings. In this case there are 2^N genotypes but only two phenotypes - the “needle” and the “hay”. For selection only, due to the strong genotype-phenotype symmetry the dynamics is much more simply considered in terms of fitness equivalence classes, as there are only two of them. However, as we pointed out in the previous section mutation breaks this symmetry. In this case the more appropriate effective degrees

of freedom are the error classes (sets of strings a fixed Hamming distance from the needle.)

Schemata offer another class of effective degree of freedom, where one coarse grains to a smaller number of fixed loci than in the original model. This has been familiar in population biology for a long time, where reduction to a small number of loci is ubiquitous. There, however, traditionally the coarse graining has been posited and a reduced model for the schemata dynamics directly written down rather than, more correctly, *deriving* the schemata dynamics from the underlying microscopic dynamics. Interestingly, any schemata coarse graining, except in exceptional cases will lead to schema fitnesses that are time dependent as they depend on the dynamics of the population. Thus, if one thinks of a fitness landscape for schemata, it will inevitably be time dependent. It is obviously of great interest to then ask when and under what approximation can the time dependence be ignored? It is natural to imagine that if fitness is defined with respect to a certain phenotypic character that depends principally on a small number of genotypic loci then the resultant landscape should be approximately time independent. Although a particular schemata-type coarse graining might suggest itself, the space of schemata-type coarse grainings has huge dimensionality (e.g. $(k+1)^N$ for N -bit strings and a cardinality k alphabet). Hence, the search for a set of schemata that capture the effective degrees of freedom is in an even larger space than the original problem! The question is under what circumstances does a particular set of schemata suggest itself? The chief cornerstones of early GA theory - Holland's Schema theorem and the Building Block Hypothesis - gave an apparent answer to this question - that it is fit, short schemata that are the effective degrees of freedom, the associated intuition being intimately linked to selection and the destructive effect of crossover. At that time no exact equations that took into account schema creation were known and the apparent contradiction between an hypothesis that posited the existence of building blocks and a theorem that did not take into account how building blocks formed higher order schemata was overlooked.

Holland's Schema theorem and the Building Block Hypothesis strongly asserted that crossover plays a privileged role in the utility of EAs. The fundamental dynamical equations (1.1) and (1.2), as in the Vose model, and as in older exact population biology models, both for the case of fixed-length strings, are written in terms of the microscopic degrees of freedom, i.e. the strings themselves. However, the simple structure hides the large scale redundancy inherent in this representation (i.e. the vast majority of the λ_{ijk} are zero) and the complication associated with the sums over the j and k . Recently, it has become

possible to write these equations in a form that extends and generalizes Holland's Schema theorem, allows for a critical and rigorous analysis of the Building Block Hypothesis and is still intimately linked to a microscopic formulation that is equivalent to the Vose model. In its original formulation it encompassed fixed-length, linear genomes. Importantly, however, it has now also been extended to variable-length linear and tree-like representations (Poli, 2000a; Poli and McPhee, 2001a; Poli, 2001a).

In the fixed-length case⁹ one may understand the relationship between the two formulations in terms of a linear coordinate transformation $\Lambda_i : \mathcal{G} \rightarrow \tilde{\mathcal{G}}$ to the BBB mentioned previously. A simple example is in the case of two bits where $\mathcal{G} = \{11, 10, 01, 00\}$, while $\tilde{\mathcal{G}} = \{11, 1*, *1, **\}$. The invertible matrix Λ , the inversion leading back to the original string basis, is such that $\Lambda_{ij} = 1 \iff j \in \xi_i$, where ξ_i is any schema associated with the string i . Note that as the choice of vertex is arbitrary there are 2^N totally equivalent BBBs.

The BBB is complete but clearly not orthonormal. By construction i is a fixed point of this transformation. Apart from the vertex i , points in $\tilde{\mathcal{G}}$ correspond to higher dimensional objects in \mathcal{G} . For instance, $1*$ and $*1$ are one-planes in \mathcal{G} while $**$ is the whole space. Note that the BBs here are not necessarily short or fit and are certainly not static when considered in the context of the evolution equation. Note as well that we are not working here in the space of all schemata. The BBB, in fact, forms a very small subset of the latter. Nevertheless, these are what are being processed by recombination. In the BBB, in the case of homologous crossover, one finds

$$\tilde{P}_i^c(t+1) = (1 - p_c)\tilde{P}_i^c(t) + \sum_{m=1}^{2^N} \sum_j \sum_k \tilde{\lambda}_{ijk}(m)\tilde{P}_j^c(t)\tilde{P}_k^c(t) \quad (1.6)$$

where $\tilde{\lambda}_{ijk}(m) = \Lambda_{ii'}\lambda_{i'j'k'}\Lambda_{j'j}^{-1}\Lambda_{k'k}^{-1}$. One may ask what is the advantage of going to this new basis? In the original string basis the properties and symmetries of $\lambda_{ijk}(m)$ are very hidden. However, this is not the case for $\tilde{\lambda}_{ijk}(m)$, which has the property that for a given mask only interactions between BBs that construct the target schema are non-zero. i.e. $\tilde{\lambda}_{ijk}(m) = 0$, unless k corresponds to a schema which is the complement of j with respect to i . Furthermore, $\tilde{\lambda}_{ijk}(m) = 0$ unless j is equivalent to m , where by equivalent we mean that for any 1 in the mask we have a 1 at the corresponding locus in j and for any 0 we have a $*$.¹⁰ These two important properties mean that the two summations over j and k in (1.6) both disappear and we are left with only the sum over masks with an "interaction" constant $p_c(m)$ which depends only on the mask. For example, for three bits, mask 100 and target

string 111 recombination of 011 with 110, 100 or 101 all lead to the desired target. However, in the BBB the mask 100 specifies as first BB parent the schema 1**. The second BB parent *11 follows naturally by complementarity.

In \mathcal{G} this has the interesting interpretation that for a target schema ξ of dimensionality $(N - d)$ only geometric objects “dual” in the d -dimensional subspace of \mathcal{G} that corresponds to ξ may interact. I.e. a k -dimensional object recombines only with a $(N - d - k)$ -dimensional object. Additionally, a $(N - d)$ -dimensional object may only be formed by the interaction of higher dimensional objects. In this sense interaction is via the geometric intersection of higher dimensional objects. For example, the point 11 can be formed by the intersection of the two lines 1* and *1. Similarly, 1111 can be formed via intersection of the three-plane 1*** with the line *111, or via the intersection of the two two-planes 11** and **11.

As mentioned, one of the primary advantages of the BBB representation is that the sums over j and k disappear. One obtains, for an arbitrary string i

$$P_i^c(t) = (1 - p_c)P_i'(t) + \sum_{m=1}^{2^N} p_c(m)P_{i_m}'(t)P_{i_{\bar{m}}}'(t) \quad (1.7)$$

where $p_c = \sum_m p_c(m)$. $P_{i_m}'(t)$ is the probability to select the BB i_m (note that the mask uniquely specifies which element, i_m , of the BBB to choose) and $P_{i_{\bar{m}}}'(t)$ the probability to select the BB $i_{\bar{m}}$ which is uniquely specified as the complement of i_m in i . Both i_m and $i_{\bar{m}}$ are elements of the BBB associated with i . The above equation clearly shows that recombination is most naturally considered in terms of the BBB. In the string basis there were of the order of 2^{2N} elements of λ_{ijk} to be taken into account for a fixed i . In the BBB there is only one term. Of course, we must remember that the coarse grained averages of i_m and $i_{\bar{m}}$ contain 2^N terms, still, the reduction in complication is enormous. Thus, we see that recombination as an operator naturally introduces the idea of a coarse graining, the natural effective degrees of freedom associated with crossover being the BBs we have defined.

Inserting (1.7) in (1.1) we can try to solve for the dynamics. However, in order to do that we must know the time dependence of i_m and $i_{\bar{m}}$. Although the number of BB basis elements is 2^N we may generalize and consider the evolution of an arbitrary schema, ξ . To do this we need to sum with $\sum_{i \in \xi}$ on both sides of the equation (1.1). This can simply be done to obtain again the form (1.1), where this time the index i runs only over the 2^{N^2} elements of the schema partition and where

again $\mathcal{P}_{ij} = p_m^{d^H(i,j)}(1 - p_m)^{N-d^H(i,j)}$. In this case however $d^H(i, j)$ is the Hamming distance between the two schemata. For instance, for three bit strings the schema partition associated with the first and third bits is $\{1 * 1, 1 * 0, 0 * 1, 0 * 0\}$. In this case $d^H(1, 2) = 1$ and $d^H(1, 4) = 2$. $P_\xi^c(t) = \sum_{i \in \xi} P_i^c(t)$ is the probability of finding the schema ξ after selection and crossover. Note the form invariance of the equation after coarse graining. To complete the transformation to schema dynamics we need the schema analog of (1.7). This also can be obtained by acting with $\sum_{i \in \xi}$ on both sides of the equation. One obtains

$$P_\xi^c(t) = (1 - p_c N_\xi) P_\xi'(t) + \sum_{m \in \mathcal{M}_r} p_c(m) P_{\xi_m}'(t) P_{\xi_{\bar{m}}}'(t) \quad (1.8)$$

where ξ_m represents the part of the schema ξ inherited from the first parent and $\xi_{\bar{m}}$ that part inherited from the second and $N_\xi = \sum_{m \in \mathcal{M}_r} p_c(m)$, where \mathcal{M}_r is the set of masks that affect ξ . Obviously, these quantities depend on the type of crossover implemented and on properties of the schema such as defining length. Note that the BBB naturally coarse grains here to the BBB appropriate for the schema ξ as opposed to the string i .

Thus, we see that the evolution equation for schemata has exactly the same form as (1.7), there being only a simple multiplicative renormalization (redefinition) of the crossover probability $p_c \rightarrow p_c N_\xi$. This form invariance, first shown in (Stephens and Waelbroeck, 1997; Stephens and Waelbroeck, 1998), demonstrates that BB schemata in general are a preferred set of coarse grained variables and more particularly the BBB is a preferred basis in the presence of recombination. It has also been shown (Vose, 1999) that schemata, more generally, are the only coarse graining that leads to invariance in the presence of mutation and recombination.

Considering again the structure of (1.7) and (1.8) we see that variables associated with a certain degree of coarse graining are related to BB "precursors" at an earlier time, which in their turn ... etc. This hierarchical structure terminates at order-one BBs as these are unaffected by crossover. Thus, for example, the level-one BB combinations of 111, i.e. BBs that lead directly upon recombination to 111, are: 11*:**1, 1*1:*1* and 1**:*11. The level-two BBs are 1**, *1* and **1. Thus, a typical construction process is that BBs 1** and *1* recombine at $t = t_1$ to form the BB 11* which at some later time t_2 recombines with the BB **1 to form the string 111.

In this basis the validity of the Building Block Hypothesis can be examined. From the structure of (1.7) we see, in fact, that *in a certain sense* the Building Block Hypothesis emerges as a logical consequence of the equations. The hierarchical structure of the equation and its

solution show unequivocally how fine grained schemata are built up from more coarse grained BBs. However, the supposition that BBs are fit and short is not generally true. The BBs that are important are those of high effective fitness. These may be short or long, fit or unfit depending on the particular characteristics of the fitness landscape and the other operators. Thus we can construct an *Effective Building Block Hypothesis* (but note that this is not a conjecture, but a mathematically provable consequence of the equations) which applies not only to GAs but to other EAs, such as GP, that fall within our unified framework:

Effective BBH: *an EA with crossover works by repeatedly combining low-order schemata of above average effective fitness to form higher-order ones.*

In the above, simply to be more concrete, we have used the fixed-length representation characteristic of GAs. However, it is important to emphasize that almost everything we have said has a natural generalization, with basically exactly the same intuition, at the level of variable-length or tree-like representations, a subject we will now consider further.

The Twisted Road to Unification: GP

To this point we have tried to present as unified a view as possible of EC theory. However, when we have passed from the abstract to the concrete we have up to now used standard GAs as a point of contact. It is important to emphasize however that much of what we have previously described has been rigorously generalized to the case of GP, which presents a host of very difficult and challenging problems not present in GAs. So, if developing a theory for GP and similar evolutionary paradigms is so difficult, why do it? That is, what do we gain from moving away from fixed-length representations into a world of variable size structures?

The first obvious answer to this is that we will understand GP itself better, one motivation being to use theory in GP to achieve analogous things to those achieved using GA theory, such as explanations, predictions, engineering design principles, etc. In the wider context of EC theory in general, however, the first thing we gain is a better understanding of what we were doing before. Work on GP theory has shown us that the evolution of fixed-length strings is in fact a special case of a much broader space of algorithms which include the evolution of: non-binary strings, strings where different loci can have different numbers of alleles, strings whose alleles can take a countably or uncountably large number of different values (like in ESs), strings whose length can be changed by

the operators, trees with fixed or variable shape and size, trees built with countably or uncountably large primitive sets (e.g. imagine an ES-GP hybrid), graphs with and without labelled links, and so on.

Secondly, because of the previous point, we have a unique opportunity to completely unify EC theory. In fact, any piece of theory one can produce which is applicable in general to this larger space of algorithms, automatically leads to corresponding results for all the subclasses, and, conversely, any specific result available in one of the subclasses will indicate the possibility that there could be a corresponding, undiscovered result for the general class. Naturally, because ES and GA theory is more developed than GP theory, we should expect that, initially, GP theory will aim at extending pre-existing results, but eventually, as the unification progresses, the biggest rewards should come from working directly in the broader space.

Thirdly, until now EC theory has only borrowed from theoretical population genetics, it has never exported results. There are many reasons for this. Partly, there is a communication problem between computer scientists and geneticists. This is not just due to the different languages we use to describe our evolutionary systems: mostly it is due to the different types of systems we study. Geneticists study diploid representations and consider recombination operators where homologous strands are aligned (by content) and may have variable lengths due to gene deletion and gene duplication events. EC theorists have almost always limited their studies only to haploid representations of fixed length undergoing position-preserving recombination operators. Moving away from fixed-length representations and position-preserving operators by embracing GP theory is a good step in the direction of being able to export our theoretical results to population genetics.

For many years GP appeared to be completely unrelated to GAs, or other fixed-length representation EAs. The differences in the representation adopted and in the semantics of the structures being evolved have been two major obstacles in bridging the gap between them. However, the characteristics of the operators adopted in GP w.r.t. those of other EAs have been one other major obstacle. Mainstream GP used crossover operators that transfer genetic material without necessarily respecting its original position in the parents. Fixed-length EAs typically did the opposite, and most EA theory was based on this very assumption. So, although people felt that there had to be some way to extend GA theory to incorporate GP, in practice that was impossible until two stepping stones became available: the notion of one-point crossover in GP and a natural extension of the notion of GA schema to GP (Poli and Langdon, 1997). GP one-point crossover is an operator where the parents are first

aligned starting from their root node and recursively traversing the two trees in parallel, stopping the exploration of each branch when an arity mismatch occurs. Then a *common* crossover point is chosen among the matching nodes and the subtrees rooted in that node in the parents are swapped to produce the offspring. Despite the useful concepts of GP one-point crossover and GP schema, however, other notions like, for example, defining length, what constitutes a building block, and so on, required a good deal of trial and error to get right. (By “getting right” here we mean that those definitions represent proper generalisations of equivalent GA notions.)

Indeed, after the GP-one-point-crossover breakthrough it has required around half a decade for homologous-GP theory to generalise some of the most fundamental results in GA theory, such as Stephens’ exact schema theory for crossover (Poli, 2000a; Poli and McPhee, 2001b; Langdon and Poli, 2002), Vose’s model for crossover (Poli et al., 2001) and Geiringer’s theorem (Poli et al., 2002b; Poli et al., 2002c; Poli et al., 2002a). Only last year, thanks to some of the tools developed in this endeavour and to some good luck, it has finally been possible to write an exact schema-based model for GP with its more standard forms of crossover (Poli, 2001b).

EC theory: The Challenges Ahead

As mentioned in the introduction: EC theory is a very exciting field with a large number of challenging problems worthy of the attention of any energetic scientist dedicated enough and with enough passion to attack them. Here we give our own personal view of some of the main ones.

C1: How do we classify EC models so that we can answer the fundamental question: when do we expect the dynamics of two different models to be qualitatively (and maybe at some point quantitatively) similar? (This is also of vital importance for practitioners). We need to understand much better the space \mathcal{E} in which EC models live and the use of metrics and similarity measures. Obviously, to answer the above we need to have a formalism within which we can work - ours in just one possibility.

C2: A deeper understanding of the various transformations - coordinate transformations, embeddings, coarse-grainings/projections and others - that change basis is, we believe, crucial for obtaining a truly unified picture of different models. It is also crucial for identifying and manipulating the appropriate effective degrees of freedom of a model. We have discussed various basis representations: the genotypic (in terms of the

"microscopic" objects) and phenotypic representations, and the BB representation. Each has its advantages and disadvantages. The genotypic representation is fundamental but rarely, if ever, are the real effective degrees of freedom directly related to the genotype. The phenotypic representation would be most appropriate in a strong selection regime. BB schemata in GAs and their extension to GP appear to be perfect at capturing the regularities present in homologous crossover operators and are most obviously the appropriate effective degrees of freedom in the case of weak selection and strong crossover. The question remains though, which coarse graining and, more generally, which basis is most appropriate for a given model. We have answers only for a small set of special cases. However, the answer here is very much related to the question of model classification, i.e. find a good basis or coarse graining for a given member of a class and the same basis should be useful for other members of the class. To distinguish one basis or coarse graining from another a quality measure (which would, of course, be parametrised by the particular problem and search algorithm at hand), would be useful in order to rank them.

C3: A more general understanding of the different bases themselves is we believe also of great importance. The BBB is, effectively, a very recent development and much remains to be understood about it. The space of trees of variable size and shape is also not well understood. Furthermore, a major challenge is to move beyond trees to a world of more general graphs. Graphs are maybe the most powerful representation available in computing. Anything from linear structures, to parallel systems, to neural networks, to organisations, etc. can be represented with graphs of one type or another. Extending the EC theory to this type of structures would give it an almost all-encompassing scope. Only a tiny amount of progress has been made here (Greene, 2000).

C4: How rugged or smooth is \mathcal{E} ? This is very important for being able to estimate the degree of validity of various exact models and approximations. Too often practitioners scoff at theoreticians working with simple model fitness landscapes, such as flat landscapes, counting ones or needle-in-a-haystack believing that they have no relevance to "real-world" problems. However, these simple models are representatives of classes of models. For example, if we consider dynamics on a linear landscape with 1-pt crossover and then add a small amount of epistasis do we expect to see big qualitative changes in the resulting model? Alternatively, starting with a genepool GA on a linear landscape and then adding weak epistasis and changing to three point crossover do we expect qualitatively different behaviour? The structure of \mathcal{E} can be studied in this sense empirically.

C5: Currently, exact schema-based models only exist for homologous type of crossover operators and subtree-swapping type of operators. However, these two classes of operators are two extremes of a continuum: one, the case in which perfect alignment of structure is imposed on the trees undergoing crossover, the other, the case in which alignment between the parent trees is not even attempted. However, this continuum is full of interesting alternatives. For example, in nature, the alignment of DNA strands is based on a matching process between bases (and, consequently, between genes). It would be interesting to extend this notion to tree-like structures and be able to model theoretically this type of process. So, being able to categorise, characterize and model the operators in this continuum is an important challenge ahead. The same kind of thing should be done for unary operators, where GP theory so far is limited to subtree-type of mutations.

C6: Effective fitness seems to offer a generalization of fitness that preserves the “hill-climbing” intuition of the latter even in the presence of operators other than selection. How general is its utility in explaining phenomena, both qualitatively and quantitatively, that do not fit into the selection/hill-climbing paradigm such as bloat (Langdon and Poli, 1997), evolution on neutral networks (Reidys and Stadler, 2001), evolutionary robustness (van Nimwegen et al., 1999) etc.?

C7: Although the development of exact evolution equations has been rapid there remains a disquieting lack of tools with which solutions to the equations may be found. In particular, we know of no systematic approximation schemes that have been studied, though several come to mind, such as an expansion around the strong selection limit, perturbing in the mutation or crossover rate. Alternatively, in the strong-crossover, weak-selection limit an expansion in principle should be possible in terms of the deviation away from flatness of the landscape. Often, expansions require an exact solution around which to expand. Hence, it is important to find as many exact solutions as possible which may serve as starting points of an expansion. All this remains to be done.

C8: All of the above we would classify as challenges primarily associated with the scientific point of view of EC. Lest we forget the engineering perspective: the developed theory should be tested to see if it can provide theoretically-valid recipes for practitioners: Which genotype-phenotype map, operators, fitness function, search algorithm, population size, number of generations, number of runs, crossover probability, anti-bloat method, etc. should one use for a given problem or a given class of problems? Perhaps a simple step in this direction is to find approximations a-la Goldberg (Goldberg, 2002) which can really help a designer, but are based on something better and more rigorous than

Holland's version of the schema theorem. Or maybe we need to find ways of characterising the sampling behaviour of different operators and defining whether this behaviour matches the shape of a particular fitness landscape and to which degree? In all this we have just started scratching the surface (McPhee and Poli, 2002).

Of course, the above list is by no means exhaustive, though we believe there is enough there to keep very many EC theoreticians busy for many years to come. Hopefully, it may help to stimulate a new generation of theoreticians as it is currently stimulating both ourselves, our collaborators and our students. We strongly believe that EC benefits strongly from an interdisciplinary approach and we would hope that more talented researchers from other fields will enter the fray bringing with them their own points of view and toolboxes. In particular, EC was inspired by evolution in nature. It would be more than fitting if EC theory could offer something back to its "older, bigger brother" - population biology.

Acknowledgements

CRS is grateful for financial support from DGAPA project ES100201 and Conacyt project 30422-E. RP would like to thank the members of the NEC (Natural and Evolutionary Computation) group at Essex for helpful comments and discussion. The authors would like to thank the editor Anil Menon for exceptionally helpful comments and suggestions.

Notes

1. These problems can, in fact, be mapped to problems familiar in very well established branches of science, such as statistical mechanics. Even there, however, where there is vast experience, they remain an enormous challenge.

2. We believe a unified framework can be given for many other classes of EA, including new ones like ant systems, artificial immune systems, etc. but this is beyond the scope of this article.

3. It is worth pointing out that for many years there has been a hot debate in EC as to the strengths and weaknesses of the notion of schema and of Holland's Schema theorem, their usefulness having been widely criticised (see for example (Chung and Perez, 1994; Altenberg, 1995; Fogel and Ghozeil, 1997; Fogel and Ghozeil, 1998)), as has the Building Block Hypothesis (Grefenstette, 1993; Stephens et al., 1999). While some criticisms are really not justified, as discussed in (Radcliffe, 1997; Poli, 2000b; Holland, 2000), others are reasonable. The debate has certainly led to some degree of confusion in the field, with most EC practitioners being divided into two different camps: those who still think Holland's Schema theorem provides a satisfactory theoretical foundation for GAs, not having heard, or not caring, about the debate about its weaknesses, and those who believe there is nothing good, not just in Holland's Schema theorem, but in the notion of schema itself and any theory built on it. Until very recently most EC theoreticians belonged to this second category (see for example (Vose, 1999, preface) and (Bäck and Fogel, 2000, Page xxxiv)). Many of them thought that only Vose's model (Nix and Vose, 1992; Vose, 1999) could provide a serious and mathematically sound way of modelling GAs. Both types of practitioners and theorists are wrong.

4. We believe that this generality extends to even more complex objects such as Neural or Bayesian networks etc.
5. Configurations will most usually be thought of as genotypes.
6. Of course, using this basis representation and developing appropriate operators for it would just lead to a form of GP which is isomorphic to current forms, and therefore the theory for such a GA-type of GP would just be isomorphic to the theory already developed there.
7. Effective (or adjusted) fitness was first introduced in (Nordin and Banzhaf, 1995b; Goldberg, 1989a) in the context of accounting for the destructive effect of crossover in the framework of Holland's Schema Theorem where a simple constant factor multiplies the landscape fitness.
8. By number of degrees of freedom we mean the number of variables needed to describe the state of an "object". In many cases those variables can actually be dependent on one another. In these cases, it is often possible to identify a smaller set of *independent* variables to describe the system in an exact or approximate, but, sufficient way. We call these the effective degrees of freedom of the system.
9. We conjecture that it is also true in the case of more general objects. The very fact that the exact schema equations for GP are so close to their GA counterparts in the BBB lend weight to this conjecture. Obviously, coordinate transformations on these more complex spaces need to be better understood.
10. This happens because a particular mask projects out a particular element of the BBB, while the other building block is specified purely by complementarity.

References

- Altenberg, L. (1995). The Schema Theorem and Price's Theorem. In Whitley, L. D. and Vose, M. D., editors, *Foundations of Genetic Algorithms 3*, pages 23–49, Estes Park, Colorado, USA. Morgan Kaufmann.
- Angeles, O., Stephens, C. R., and Waelbroeck, H. (1998). Emergence of algorithmic language in genetic systems. *Biosystems*, 47:129–147.
- Bäck, T. and Fogel, D. B. (2000). Glossary. In Bäck, T., Fogel, D. B., and Michalewicz, T., editors, *Evolutionary Computation 1: Basic Algorithms and Operators*. Institute of Physics Publishing.
- Bürger, R. (2000). *The Mathematical Theory of Selection, Recombination, and Mutation*. Wiley, Chichester, UK.
- Chung, S. W. and Perez, R. A. (1994). The schema theorem considered insufficient. In *Proceedings of the Sixth IEEE International Conference on Tools with Artificial Intelligence*, pages 748–751, New Orleans.
- Eigen, M., McCaskill, J., and Schuster, P. (1989). The molecular Quasispecies. *Adv. Chem. Phys.*, 75:149–263.
- Fogel, D. B. and Ghozeil, A. (1997). Schema processing under proportional selection in the presence of random effects. *IEEE Transactions on Evolutionary Computation*, 1(4):290–293.
- Fogel, D. B. and Ghozeil, A. (1998). The schema theorem and the misallocation of trials in the presence of stochastic effects. In Porto, V. W., Saravanan, N., Waagen, D., and Eiben, A. E., editors, *Evolutionary*

- Programming VII: Proc. of the 7th Ann. Conf. on Evolutionary Programming*, pages 313–321, Berlin. Springer.
- Goldberg, D. E. (1989a). Genetic algorithms and Walsh functions: I. A gentle introduction. *Complex Systems*, 3(2):129–152.
- Goldberg, D. E. (1989b). Genetic algorithms and Walsh functions: II. Deception and its analysis. *Complex Systems*, 3(2):153–171.
- Goldberg, D. E. (1989c). *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, Reading, Massachusetts.
- Goldberg, D. E. (2002). *The Design of Innovation*. Kluwer Academic Publishers, Boston.
- Greene, W. A. (2000). A non-linear schema theorem for genetic algorithms. In Whitley, D., Goldberg, D., Cantu-Paz, E., Spector, L., Parmee, I., and Beyer, H.-G., editors, *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2000)*, pages 189–194, Las Vegas, Nevada, USA. Morgan Kaufmann.
- Grefenstette, J. J. (1993). Deception considered harmful. In Whitley, L. D., editor, *Foundations of Genetic Algorithms 2*, San Mateo, CA. Morgan Kaufman.
- Holland, J. (1975). *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor, USA.
- Holland, J. H. (2000). Building blocks, cohort genetic algorithms, and hyperplane-defined functions. *Evolutionary Computation*, 8(4):373–391.
- Jones, T. (1995). *Evolutionary Algorithms, Fitness Landscapes and Search*. PhD thesis, The University of New Mexico, Albuquerque, NM.
- Langdon, W. B. and Poli, R. (1997). Fitness causes bloat. In Chawdhry, P. K., Roy, R., and Pant, R. K., editors, *Soft Computing in Engineering Design and Manufacturing*, pages 13–22. Springer-Verlag London.
- Langdon, W. B. and Poli, R. (2002). *Foundations of Genetic Programming*. Springer-Verlag.
- McPhee, N. F. and Poli, R. (2002). Using schema theory to explore interactions of multiple operators. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2002)*, New York, USA. Morgan Kaufmann. (accepted as full paper).
- Mora, J., Stephens, C., Waelbroeck, H., and Zertuche, F. (1999). Symmetry breaking and adaptation: Evidence from a simple toy model of a viral neutralization epitope. *Biosystems*, 51:1–14.
- Nix, A. E. and Vose, M. D. (1992). Modeling genetic algorithms with Markov chains. *Annals of Mathematics and Artificial Intelligence*, 5(1):79–88.

- Nordin, P. (1997). *Evolutionary Program Induction of Binary Machine Code and its Applications*. PhD thesis, der Universitat Dortmund am Fachereich Informatik.
- Nordin, P. and Banzhaf, W. (1995a). Complexity compression and evolution. In Eshelman, L., editor, *Genetic Algorithms: Proceedings of the Sixth International Conference (ICGA95)*, pages 310–317, Pittsburgh, PA, USA. Morgan Kaufmann.
- Nordin, P. and Banzhaf, W. (1995b). Complexity compression and evolution. In Eshelman, L., editor, *Genetic Algorithms: Proceedings of the Sixth International Conference (ICGA95)*, pages 310–317, Pittsburgh, PA. Morgan Kaufmann.
- O’Neill, M. and Ryan, C. (2001). Grammatical evolution. *IEEE Transaction on Evolutionary Computation*. Forthcomming.
- Poli, R. (2000a). Exact schema theorem and effective fitness for GP with one-point crossover. In Whitley, D., Goldberg, D., Cantu-Paz, E., Spector, L., Parmee, I., and Beyer, H.-G., editors, *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 469–476, Las Vegas. Morgan Kaufmann.
- Poli, R. (2000b). Why the schema theorem is correct also in the presence of stochastic effects. In *Proceedings of the Congress on Evolutionary Computation (CEC 2000)*, pages 487–492, San Diego, USA.
- Poli, R. (2001a). Exact schema theory for genetic programming and variable-length genetic algorithms with one-point crossover. *Genetic Programming and Evolvable Machines*, 2(2):123–163.
- Poli, R. (2001b). General schema theory for genetic programming with subtree-swapping crossover. In *Genetic Programming, Proceedings of EuroGP 2001*, LNCS, Milan. Springer-Verlag.
- Poli, R. and Langdon, W. B. (1997). A new schema theory for genetic programming with one-point crossover and point mutation. In Koza, J. R., Deb, K., Dorigo, M., Fogel, D. B., Garzon, M., Iba, H., and Riolo, R. L., editors, *Genetic Programming 1997: Proceedings of the Second Annual Conference*, pages 278–285, Stanford University, CA, USA. Morgan Kaufmann.
- Poli, R. and McPhee, N. F. (2001a). Exact schema theory for GP and variable-length GAs with homologous crossover. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001)*, San Francisco, California, USA. Morgan Kaufmann.
- Poli, R. and McPhee, N. F. (2001b). Exact schema theory for GP and variable-length GAs with homologous crossover. Technical Report CSRP-01-4, University of Birmingham, School of Computer Science.

- Poli, R., Rowe, J. E., and McPhee, N. F. (2001). Markov models for gp and variable-length gas with homologous crossover. Technical Report CSRP-01-6, University of Birmingham, School of Computer Science.
- Poli, R., Rowe, J. E., Stephens, C. R., and Wright, A. H. (2002a). Allele diffusion in linear genetic programming and variable-length genetic algorithms with subtree crossover. In *Proceedings of EuroGP 2002*.
- Poli, R., Stephens, C. R., Wright, A. H., and Rowe, J. E. (2002b). On the search biases of homologous crossover in linear genetic programming and variable-length genetic algorithms. In Langdon, W. B., Cantú-Paz, E., Mathias, K., Roy, R., Davis, D., Poli, R., Balakrishnan, K., Honavar, V., Rudolph, G., Wegener, J., Bull, L., Potter, M. A., Schultz, A. C., Miller, J. F., Burke, E., and Jonoska, N., editors, *GECCO 2002: Proceedings of the Genetic and Evolutionary Computation Conference*, pages 868–876, New York. Morgan Kaufmann Publishers.
- Poli, R., Stephens, C. R., Wright, A. H., and Rowe, J. E. (2002c). A schema-theory-based extension of Geiringer’s theorem for linear GP and variable-length GAs under homologous crossover. In Jong, K. D., Poli, R., and Rowe, J., editors, *Proceedings of the Foundations of Genetic Algorithm (FOGA-VII) Workshop*, Torremolinos.
- Prügel-Bennett, A. and Shapiro, J. L. (1994). An analysis of genetic algorithms using statistical mechanics. *Physical Review Letters*, 72:1305–1309.
- Radcliffe, N. J. (1997). Schema processing. In Baeck, T., Fogel, D. B., and Michalewicz, Z., editors, *Handbook of Evolutionary Computation*, pages B2.5–1–10. Oxford University Press.
- Reidys, C. M. and Stadler, P. F. (2001). Neutrality in fitness landscapes. *Appl. Math. & Comput.*, 117:321–350.
- Reidys, C. M. and Stadler, P. F. (2002). Combinatorial landscapes. *SIAM Review*, 44:3–54.
- Stadler, P. F. and Stephens, C. R. (2003). Landscapes and effective fitness. *Comm. Theor. Biol.*, to be published. Santa Fe Institute Working Paper: 02-11-062.
- Stephens, C. R. (1999a). Effect of mutation and recombination on the genotype-phenotype map. In Banzhaf, W., Daida, J., Eiben, A. E., Garzon, M. H., Honavar, V., Jakiela, M., and Smith, R. E., editors, *Proceedings of the Genetic and Evolutionary Computation Conference*, volume 2, pages 1382–1389, Orlando, Florida, USA. Morgan Kaufmann.
- Stephens, C. R. (1999b). Effective fitness landscapes for evolutionary systems. In Angeline, P. J., Michalewicz, Z., Schoenauer, M., Yao, X., and Zalzala, A., editors, *Proceedings of the Congress on Evolutionary*

- Computation*, volume 1, pages 703–714, Mayflower Hotel, Washington D.C., USA. IEEE Press.
- Stephens, C. R. (2001). Some exact results from a coarse grained formulation of genetic dynamics. In Spector, L., Goodman, E. D., Wu, A., Langdon, W. B., Voigt, H.-M., Gen, M., Sen, S., Dorigo, M., Pezeshk, S., Garzon, M. H., and Burke, E., editors, *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001)*, pages 631–638, San Francisco, California, USA. Morgan Kaufmann.
- Stephens, C. R. (2003). The renormalization group and the dynamics of genetic systems. *Acta Phys. Slov.*, to be published. Preprint: cond-mat/0210217.
- Stephens, C. R., García Olmedo, I., Mora Vargas, J., and Waelbroeck, H. (1998). Self-adaptation in evolving systems. *Artificial Life*, 4:183–201.
- Stephens, C. R. and Vargas, J. M. (2000). Effective fitness as an alternative paradigm for evolutionary computation I: general formalism. *Genetic programming and evolvable machines*, 1(4):363–378.
- Stephens, C. R. and Vargas, J. M. (2001). Effective fitness as an alternative paradigm for evolutionary computation II: examples and applications. *Genetic programming and evolvable machines*. Forthcoming.
- Stephens, C. R. and Waelbroeck, H. (1997). Effective degrees of freedom in genetic algorithms and the block hypothesis. In Bäck, T., editor, *Proceedings of the Seventh International Conference on Genetic Algorithms (ICGA97)*, pages 34–40, East Lansing. Morgan Kaufmann.
- Stephens, C. R. and Waelbroeck, H. (1998). Effective degrees of freedom in genetic algorithms. *Phys. Rev.*, 57:3251–3264.
- Stephens, C. R. and Waelbroeck, H. (1999). Schemata evolution and building blocks. *Evolutionary Computation*, 7(2):109–124.
- Stephens, C. R., Waelbroeck, H., and Aguirre, R. (1999). Schemata as building blocks: Does size matter? In Banzhaf, W. and Reeves, C., editors, *Foundations of Genetic Algorithms 5*, pages 117–133. Morgan Kaufmann, San Francisco, CA.
- van Nimwegen, E., Crutchfield, J. P., and Huynen, M. A. (1999). Neutral evolution of mutational robustness. *Proc. Natl. Acad. Sci. USA*, 96:9716–9720.
- Vose, M. D. (1999). *The simple genetic algorithm: Foundations and theory*. MIT Press, Cambridge, MA.
- Whitley, D. (1992). An executable model of a simple genetic algorithm. In Whitley, D., editor, *Foundations of Genetic Algorithms Workshop (FOGA-92)*, Vail, Colorado.
- Wright, S. (1932). The roles of mutation, inbreeding, crossbreeding and selection in evolution. In Jones, D. F., editor, *Proceedings of the Sixth International Congress on Genetics*, volume 1, pages 356–366.

- Wright, S. (1967). "Surfaces" of selective value. *Proc. Nat. Acad. Sci. USA*, 58:165–172.