On Natural Life's Tricks to Survive and Evolve

Hans–Paul Schwefel and Frank Kursawe[*]
University of Dortmund
D–44221 Dortmund, Germany
E-mail: {schwefel,kursawe}@LS11.cs.uni-dortmund.de

Abstract

Which are the fundamental principles of life? This is the main question to be addressed if one tries to create artificial life on computers. Though it has been answered only partially, evolutionary algorithms are substantially contributing already to many kinds of human problem solving by means of virtual organisms. Besides looking back on that success story and extrapolating it a bit into the future – both endeavors obviously being subjective –, a new result will be presented in the following showing the importance of multicellularity, which helps to self-adapt the error rates of the replication step to what is needed for efficacious and efficient optimum seeking without individual learning.

Keywords: evolutionary computation, evolutionary algorithms, imitating life, natural computation, binary optimization, evolution strategies, self-adaptive mutabilities, multicellularity, ontogeny, somatic mutations, phenotypic plasticity.

1 Overview

Efforts to model, algorithmically, the basic evolutionary principles *population, self-replication, variation, and selection* have been traced back to the 1950s in the *Handbook of Evolutionary Computation*[1]. They are part of what has been called *biologically inspired* or *natural computation* (see Schwefel[2]).

Currently, three vital schools from the initial phase of *Evolutionary Computation* (EC) can be distinguished:

- *Evolutionary Programming* (EP), first roots of which were laid by Lawrence Fogel[3] and which was redesigned by David Fogel[4] to its current form;
- *Genetic Algorithms* (GA), which Holland[5] used to explain the adaptive behavior of basic life forms, but later have become better known as tool for solving (mostly combinatorial) optimization tasks (see Goldberg[6]);
- *Evolution Strategies* (ES), developed by Rechenberg[7] and Schwefel[8], first as a rule set for experimental, later as algorithms for numerical optimization.

Genetic Programming (GP)[9] and *Learning Classifier Systems* (LCS) have branched from the GA philosophy. The former is a separate school now, the latter seems to wait for new ideas to develop further.

Though 'religious' wars about the 'proper' modeling of evolutionary processes are melting down – many hybrid *Evolutionary Algorithms* (EA) are currently in use and under analytic investigation – the three schools mentioned above have retained some of their initial specifics.

In Section II we briefly look at the success story of all EA and at the differences they maintain in modeling organic evolution. There are two aspects of that modeling process: On

[*] This work is a result of the *Collaborative Research Center SFB 531* at the University of Dortmund. Financial support by the German National Science Foundation (DFG) is gratefully acknowledged.

the one hand, there is the desire to make use of life's tricks for solving difficult technical or managerial problems; on the other hand, doing so successfully, one hopes to gain some insight into why nature has gone the way it obviously did. Section III tries to summarize our point of view of what we have learned during the modeling process. This part of the paper may be a bit provocative. Then it serves its goal to enhance the search for better models.

We are then going to employ one of nature's tricks for maintaining life and striving for higher and higher forms despite of the entropy law and (often self-induced) degrading environments including catastrophes in Section IV. The case we handle is multicellularity and somatic mutations. Mimicking the cell divisions during ontogeny, we found a way to adjust single mutation rates for many genes even in epistatic binary optimization. Section V is a wanted for speculation into the next decade, which, of course, may be completely wrong. We would like to call it wishful thinking, hoping that it contains some self-fulfilling prophecy.

2 Introduction

After more than twenty years of sporadic publications – Alander[10] counted 99 publications before 1980 – the GA community invited to the first *International Conference on Genetic Algorithms* (ICGA) in 1985. Since that time the group has maintained its conferences in the U.S.A. every second year. Five years later, a couple of European researchers in the field of ES, GA, and other approaches gleaned from natural processes started another biennial conference series *Parallel Problem Solving from Nature* (PPSN) with a broader scope of topics from 'imitating life'. The Evolutionary Programming Society at San Diego started *Annual Meetings of the EP Society* in 1992, the IEEE Neural Network Council annual *International Conferences on Evolutionary Computation* (ICEC) in 1994.

At the time being we count more than twenty international events per year in closely related fields[2], at least half a dozen corresponding journals, and more than 1000 papers published per year[10]. There are countless successful applications in many different fields, where EA have proven capable of solving hard design, management and planning, as well as control problems.

Why did it take the basic ideas so long to become broadly accepted? The following remarks are limited to certain perspectives. Let us try to paraphrase them briefly:

- The basic ideas were ingenious, though aiming at answering different questions and/or solving different kinds of problems in different environments. But the three schools mentioned above, being unaware of each other in the beginning, acted separately until about 1990.

- The numeric power of computers has been increased by several powers of ten within those thirty plus years, thus enabling the simulation of many generations and large populations, now, but not much earlier.

- All achievements in the world of crisp computing (see Zadeh's work[11] on fuzzy sets for the dichotomy crisp versus soft computing), have not yet lived up to the aspirations provided at the beginning. Subsymbolic information processing seems to have merits as well as symbolic knowledge processing.

- Even ad-hoc adaptations of evolutionary algorithms to specific difficult decision making problems have proven to yield results not achievable with classical problem solving approaches.

- A thorough theory of EC is still missing, despite hundreds of articles on theoretical investigations with very limited scope.

- Too many researchers in the field are clinging too closely to their origins (either EP, GA, or ES) and have lost connection to the dual view that 'imitating life' is a means of creating effective and efficient problem solving procedures as well as a means of better understanding natural life.

Many people looking for the first time onto detailed descriptions of different incarnations of EA, e.g. GA and ES, wonder why the operators for GA are usually presented in the sequence SRM (selection, recombination, mutation), whereas for ES the sequence RMS is mostly used. It has been tried to remove that discrepancy by saying that this is a question of the entry into the iterative loop, but that the loop itself is the same. This is not fully convincing, however. That is why a different type of loop, valid for all types of EA, is presented in Fig. 1.

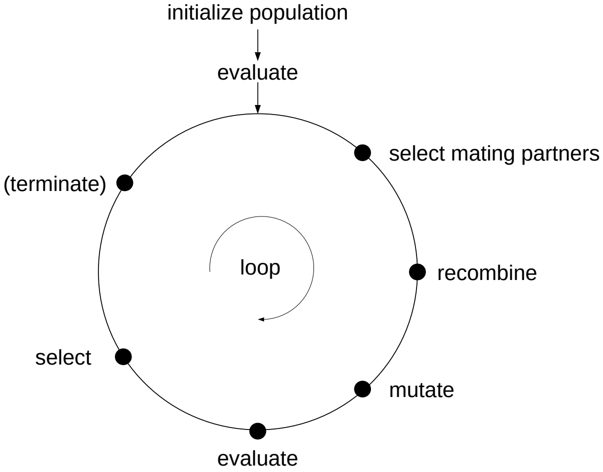

Figure 1: The basic loop of all EA.

All special incarnations now omit one or the other step within the generational loop. The following statements refer to 'canonical' versions of the algorithms and to not take account of the full variety existing or proposed.

EP obviously omits recombination since its philosophy relies on species as evolving entities, and per definition, species do not interchange genetic information (at least no 'higher' species). Thus, there is no mating selection and no crossover as to be found in the GA realm. Within GA, crossover is the basic variation mechanism, whereas mutation is subsidiary or even omitted. They let all descendants reach adulthood, where they enter the mating selection with fitness-dependent probability. The number of descendants created is never larger than the number of parents was.

ES, on the contrary, generate a birth surplus. Environmental selection after birth and before reproduction cuts the population down to a constant size of parents entering the next iteration. Mating is uniformly random, thus not selective, in ES. Whereas in non-elitist ES and GA no descendant struggles against his parents for survival, this is the normal case within canonical EP. Repeated tournaments halve the whole population of old species and equally many new species.

Isn't it astonishing that no rigorous examination of the benefits and shortcomings of these selection types modeled so differently exists? Blickle's[12] investigation of different selection operators alone fades out the other operators for variation, but only their interplay may yield a thorough understanding of evolutionary processes as a whole.

It has often wrongly been stated that ES are only good for real valued variables, GA for binary or integer ones. But currently, there are real-valued GA[13], and at the very beginning of the ES history the variables within experimental settings were discrete, not continuous. In this paper, we even show a case of binary variables solved by means of an otherwise standard ES.

3 Some Lessons Learned from Modeling Evolution

The exploration and exploitation schemes used in contemporary EA are still quite simple models of real life, taking into account just some basic features of very simple organisms[14]. Nevertheless, their use and analysis has told us some lessons already. We admit that the following remarks are essentially subjective.

3.1 Birth and Death; No Individual Immortality

Perhaps the most important distinction between living and non-living entities is that the latter just decay over time, due to the law of entropy, whereas the former – as a whole – buy endurance, adaptivity to changing environments (even if self-induced), and emergence of ever higher complexity by short-living of single individuals. Not only individuals and populations, even species and higher taxa, are mortal. Even within individual living beings, the number of cell divisions seems to be limited by genetically controlled mechanisms. In case of drastic environmental changes one has found that species with a shorter generational cycle of birth, reproduction, and death, are capable of adapting faster. Of course, there are other 'natural' tricks to deal with such situations – additionally, e.g. polyploidy and epigenesis.

Individual mortality seems to be a necessary ingredient of effective self-adaptation of internal strategy parameters[15]. The principle of forgetting good intermediate solutions with positive probability is essential for *simulated annealing*, as well. Moreover, it helps in hunting dynamic minimizers in control problems, a situation in which an elitist EA, eagerly conserving already achieved improvements (e.g. so-called plus-versions of ES) loses adaptivity[16], though theoretically, its global convergence can be proven under more general conditions than that of a non-elitist EA[17]. The latter are in danger of divergence if they are parameterized improperly.

3.2 Knowledge Propagation; No Prediction

Nature's trick to preserve acquired knowledge to some extent may be seen in storing individuals' blue prints within the genome and proliferating just this bootstrap program for a highly nonlinear self-organizing process. No long-term memory, no analysis of the history, and no prediction of the future are involved. The information processing from one generation to the next works like a simple Markov chain. The knowledge processed is just a recipe that has been successful to survive the time from inception to reproduction.

In eukaryotes the nucleus is just one part of the reproduction machinery. Other organelles are responsible for the interpretation of the program and for carrying on the building blocks to construct proteins and enzymes. The genome contains both functions for the proteins building up the phenotype and functions for the enzymes controlling the processes involved.

The genetic code, now equal for nearly all forms of life, plays an important role during the creation of each individual cell. It must have been developed in the early stages of life. Altering it now within highly sophisticated, well adapted situations, is nearly always lethal. Thus, it can be explained why the genetic code may not have reached an 'optimal' state with respect to the efficiency of the search for improvements[7]. But, there are more steps beyond the first translation from RNA/DNA to amino acids until a living being is born. Though they are not yet completely understood, it seems that altogether strong causality is achieved in most cases, i.e. small changes in the genome normally yield small changes in the phenotype. This helps to circumvent Hamming cliffs and makes the fitness landscape smooth enough for efficient adaptation/amelioration.

3.3 Error Reduction; No Precision

Reproduction by copying useful information from individuals that have managed to survive in their environment, at least for a while, to descendants is the basic trick of life. As Fisher and Eigen[18] have shown by means of a simple mathematical model, the main problem of reproduction is the correctness of the replication. The longer the chain of information to code an individual becomes, the more it is necessary to reduce errors caused by the environment. Despite of the necessity to repair copying errors – this has been achieved by proper enzymes, also encoded in the genome of all living beings – the error rate has never dropped to zero. On the contrary, there seem to exist error enhancing enzymes, as well, specifically working at certain loci of the chain of informations delivered from one generation to the next. Recently, the journal *Nature* reported that one has identified co-activators and co-repressors, a fact underpinning that the idea of correlated mutations as used in ES is not at all an artifact.

Without variation there is no improvement, thus no adaptation nor any invention of new forms of life would be possible without the risk of imperfection. The redundancy of the genetic code and all other transformation steps between genotype and phenotype finally seem to result in an error distribution at the phenotypic level that can most easily be described by a normal or a geometrical density function: Smaller errors are more frequent than larger ones. The mutation steps are neither purely random or volume-oriented, nor are they infinitesimally small or path-oriented. Organic evolution is more of a pogo-sticking trial-and-error process, a compromise between efficient local and effective global search.

Moreover, it is essential to remember that we are usually beginning from scratch when we use EA for solving optimization tasks. The low mutation rates currently observed in nature may be adequate for situations near the optimum or equilibrium, but not for starting from scratch. In each case, self-adaptation of the mutation strength is the best way to handle the search for an appropriate mutability. 'Intelligent' variation, genetically programmed, seems to be a hidden or at least difficult to detect on-line adaptation process.

3.4 Ontogeny and Multicellularity; Fuzzification

Early life forms have been unicellular, first prokaryotic only, later mostly eukaryotic, i.e. with distinct organelles in a containment being responsible for the control of different subtasks of the reproduction cycle. A very early invention of real life after merely aggregating undifferentiated cells into case-based agglomerates, was the programming of differentiated cells of multicellular systems within one genome. In this way, the ontogeny of an individual from one first cell by consecutive cell divisions could lead to an assembly of specialized tissues that are dividing the labour of solving the different tasks to be performed in order to survive – for a while – as a whole. This early invention of real life has not been taken into account in evolutionary computation so far. We shall make use of that feature in section IV to tackle the problem of self-adapting, loci specific mutation rates.

3.5 Sexual Propagation and Polygeny

Mixing of genetic information to be transmitted to a descendant from more than one ancestor seems to be an old achievement of the prokaryotic regime. It has been reduced to a now dominant bisexual hereditary scheme in multicellular eukaryotic living beings. Beyer has shown[19] that recombination can yield linear speedup with the size of the population. This needs parallel processing of the reproduction step. He argues that recombination can be regarded as *genetic repair*, i.e., two unfavorable deviations from a nearby better position compensate each other. This seems to be contradictory to the *building block hypothesis*, but both effects may contribute to the 'benefit of sex', depending on the specific situation. In epistatic convex landscapes the former effect should be more important than the latter.

Sexual propagation seems to go hand in hand with polyploidy, or at least diploidy. This feature has not yet been fully explored in connection with EA. One application has been in creating an ES to solve multiple criteria decision making (MCDM) problems[20]. The different criteria are stochastically involved in the selection step, thus driving the population toward the Pareto-optimal subset of solutions. Indeed, polyploidy is often observed in environments where drastic changes occur frequently. Maintaining a longer-term memory of past successes seems to be the main benefit of polyploidy. One may presume therefore that the efficacy of organic evolution as well as its efficiency would be substantially lower if sexuality and polyploidy had not been ‘invented’ in nature. Both features are no fancy.

3.6 External Checkup; No Pity

No individual organism is guaranteed to reproduce, due to an unfit mutated genome, a change of the environment, or due to predators. Obviously, practically all species produce more than one descendant per ancestor. Otherwise, the population at hand would die out finally. Perhaps Charles Darwin would not have insisted upon his view of natural selection if Thomas Malthus had not predicted misery to mankind due to inherent over-reproduction and limited resources a century earlier. Malthus assumed the former to be a geometrically, the latter to be a linearly increasing process. Over-reproduction can, and sometimes must, be substantial to maintain the population size against predators, accidents, and lethal variations. Generally, limited food supply as well as the check for survivability according to all kinds of tests like compatibility with ‘eternal laws’ of nature and escapability from predators, keeps down the abundance of a species to what ‘the rest of the world’ allows. Altogether, this might be called *environmental selection*, reducing the higher number of newborns to a lower number of individuals fit enough to transmit their tested information content to the next generation. Each individual or species dying prematurely gives room to others that are more fit.

The term ‘struggle for life’ often leads people to believe in some other form of selection, i.e. tournaments between individuals existing simultaneously. If this were the dominant form of selection, how could it work in the realm of plants? Since EA otherwise model very primitive life, environmental selection may be the more adequate operator. Mating selection may play an additional role sometimes. But guinea-pig researchers report that parents with an average fitness often have most progeny[21].

3.7 Parallelism; No Central Control

Parallelism is an intrinsic feature of evolution in nature. There are always many individuals at the same time involved in the ‘life game’. Whether one should model artificial evolution in a synchronous or an asynchronous manner largely depends on the type of hardware used and on the CPU time necessary for the evaluation of the fitnesses. In nature, one has seasonality playing a role, but no strong synchronization, since there is no central controller known.

Larger populations tend to split up into subgroups, more or less isolated from each other, depending on the intensity of migration between the subgroups. Modeling that kind of geographical dispersion can be done in a fine- or a course-grained manner with strictly local interactions only or many kinds of intermediary forms. The incest taboo, much earlier in place than human beings, seems to be useful in searching mates that are different enough from each other to gain the full benefit from recombination (see above). It has been demonstrated that global convergence can be enhanced by distributed searchers and local operators[22, 23].

4 An Old Idea Revisited: Somatic Fuzzification

To show the benefit of somatic mutations during the ontogeny of a multicellular organism, let us start off with a well studied situation, called the *Counting Ones Problem* (COP). The virtual individuals (we call them BW) do have $n = 256$ phenotypic characters, e.g. color patches on their surface that can be either black or white. Correspondingly, their genome contains $n = 256$ gene loci $x_i, \forall i = 1, 2, \ldots, n$ with just 2 alleles A and C, $x_i =$ ‘A’ encoding a white patch, x_i = 'C' a black one. The environment evaluates a phenotype according to a merit function $F(x)$. For the COP it can be formulated as

$$F_1(x) = \sum_{i=1}^{n} f(x_i) \text{ with } f(x_i) = \left\{ \begin{array}{l} 0 \text{ if } x_i \neq x_i^* \\ 1 \text{ if } x_i = x_i^* \end{array} \right. \quad (1)$$

where the string $\{x^*\}$ characterizes the optimal genotype. F_1 just counts the number of matching loci.

If we start with a pure random setting $\{x^{(0)}\}$ we can expect $F_1^{(0)} \approx \frac{n}{2} = 128$ as the value of the merit function from which any kind of stepwise improvement process could begin. The simplest way would be to successively switch all x_i from their current state to their counterstate and fixing them if an improvement occurred, otherwise setting them back to their old state. This would cost exactly $n = 256$ trials since it is unknown which loci must be flipped. On average, every second trial is successful, and we end up with $F_1^{(256)} = 256$.

Bäck [24] has investigated the same selection scheme, the so-called $(1+1)$ ES, but with randomized variation. Each of the loci undergoes flipping with the same mutation rate p. This probability determines the expected number of steps until the optimum is reached. If one does not choose an elitist selection like above, but e.g. the standard roulette wheel proportional selection used within GA, then the search process finally fluctuates at some distance from the optimum, the average distance itself increasing with the mutation rate chosen (see Rudolph [17]).

The last step in the elitist case would be best done with a mutation rate $p = \frac{1}{n}$, and it would take about n trials to hit the last incorrect gene. When half of the genes are correct, the optimal mutation rate is $p = \frac{1}{2}$. In general, the optimal mutation rate depends on the number of already correct positions in the genome, but this kind of knowledge is normally not available. Mühlenbein[25] derived an expression for the expected number N of steps under $(1+1)$ ES selection conditions and with $p = \frac{1}{n}$ during the whole search process and 50% correct genes at the beginning: $N = e\ n\ \ln(\frac{n}{2})$, which leaves us with $N \approx 3376$ for $n = 256$. For a more rigorous treatment see Droste et al.[26].

Multimembered (μ, λ) ES have been very successful in self-adapting mutation strengths on-line, even different ones for each gene. For the basic algorithm used in the following we refer to [27]. Local mutation rates p_i for already correct genes should decrease, those of others first increase until the mutation takes place, and thereafter decrease again in order not to lose the merit won. However, any kind of self-adapting strategy parameters relies upon improvements gained with more appropriate values for them. This condition is violated with an only stepwise changing merit function like F_1, especially in the final stage of the search.

Very early in the course of natural life, multicellular organisms appeared. Their ontogeny starts with one cell, of course. This cell divides, forming two cells with all compartments for each, including the nucleus containing the complete genetic information. The process is repeated M times so that at the end the adult individual consists of 2^M cells, in case of human beings about $2^{50} \approx 10^{15}$. Different cell types (about 256 according to Kauffman[28]) do have different tasks and are individually programmed by the genome; the others are 'duplicates', not necessarily completely equal to their prototype, however.

According to findings of guinea-pig researchers (e.g. Gärtner[29]), cloned, i.e. genetically equal, individuals differ considerably with respect to their phenotypes. One must assume that errors occur during the cell doublings. No other source of the 'intangible variance' could be identified. Errors are controlled by repair enzymes and it is thus rather straightforward to assume that the somatic mutation rate during ontogeny is similar if not equal to the genetic mutation rate.

Let us exemplify this on the basis of the COP. There are now $n = 256$ cell types, coded in the genome. They should appear after the first 8 cell divisions. Within the genome we also have n mutation rates p_i associated to the n cell types. During the latter 42 out of 50 cell divisions copying errors occur at that rate just like genetic mutations appear from one generation to the next. Thus, 256 patches of 2^{42} genetically equal but somatically different cells form each whole BW.

Instead of black and white patches we now find all kinds of grey ones, depending on their genetically determined color x_i and on the somatic mutation rate. The latter is taken to be equal to the (genetically encoded) genetic mutation rate p_i. One could speak of some kind of fuzzification of the phenotypic characteristics. Knowing the genetic value and the mutation rate p one can calculate the distribution of the grey tones. It is a binomial distribution with probability $w(k, M, p)$ that just k out of 2^M cells are mutated (see Schwefel[30])

$$\begin{aligned} w(k, M, p) &= \binom{2^M}{k} q^k (1-q)^{2^M - k} \text{ with} \\ q &= \frac{1}{2}[1 - (1-2p)^M]. \end{aligned} \tag{2}$$

Though the distribution is symmetric only for $p = \frac{1}{2}$, it can be approximated largely by a normal distribution $\mathcal{N}(\xi, \sigma)$ if M is large enough – which is the case here. After normalization to the interval $[0, 1]$ indicating the extremes 'all white' and 'all black' the mean ξ of the deviation is $\xi = q$ and the standard deviation $\sigma = \sqrt{\frac{q(1-q)}{2^M}}$. By adding a bonus term $\mathcal{N}(\xi_i, \sigma_i)$ for those loci which do not yet match and subtracting a penalty (malus term) of equal size for the already matching ones, selection now can work toward adjusting local mutation rates. Note that here is no learning during the life span of the individuals, since no feedback to fitness is assumed during the cell divisions. The smoothing effect of the this kind of phenotypic plasticity to the fitness landscape may be similar to that of ontogenetic learning according to the model of Hinton and Nowlan[31], but the mechanism is completely different.

Now, we can simulate the process using a modified merit function

$$\begin{aligned} F_1'(x) &= \sum_{i=1}^{n} f'(x_i, p_i) \text{ with} \\ f'(x_i, p_i) &= \begin{cases} 0 + \mathcal{N}(\xi_i, \sigma_i) \text{ if } x_i \neq x_i^* \\ 1 - \mathcal{N}(\xi_i, \sigma_i) \text{ if } x_i = x_i^* \end{cases} \end{aligned} \tag{3}$$

Fig. 2 illustrates the principle of a genetic 'bonus' for different numbers N of cell divisions considered.

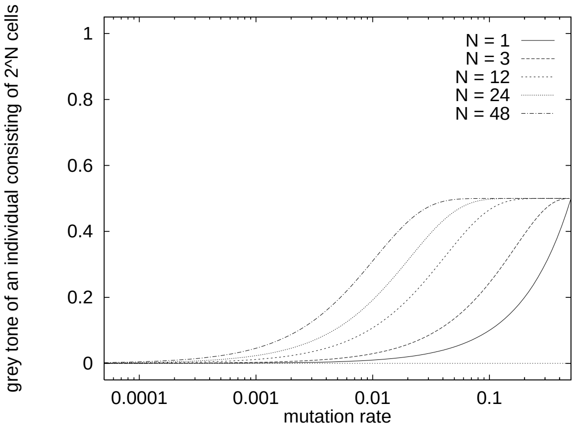

Figure 2: Intermediary grey tone values derived from the mutation rates (bonus only) for different numbers N of cell divisions.

It may happen that $f'(x_i, p_i)$ leaves the interval $[0, 1]$. In that case we simply cut off at zero or one, respectively. With that trick we can easily solve the COP. Figures 3 to 5 show results from several simulations with a $(15, 100)$ ES using global discrete recombination for both object and strategy parameters. Fig. 3 presents a plot of both F_1 as well as F_1' for the best individual over the number of generations (3 runs). Fig. 4 zooms into a section of one simulation to see the differences between the genetic merit function F_1 and the phenotypic one (F_1'). Dealing with 256 bits at the same time, the principle of a genetic bonus or malus (penalty) cannot be seen as well as in Fig. 2.

Fig. 5 shows 3 out of the 128 mutation rates p_i belonging to gene loci that are not correct at

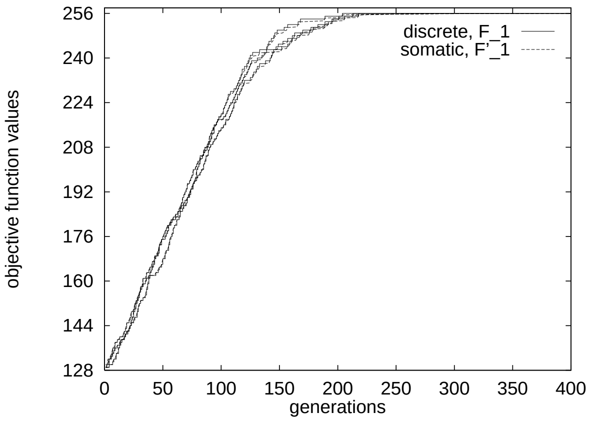

Figure 3: 3 runs of a $(15, 100)$ ES solving the COP.

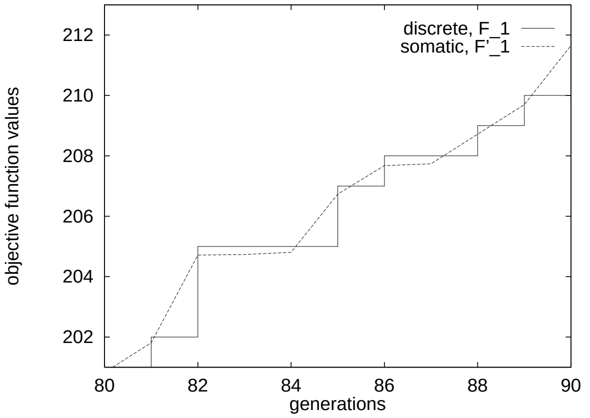

Figure 4: Zoom into a single run of a $(15, 100)$ ES solving the COP.

the start. It also contains one single and the average of those mutation rates that belong to already correct gene loci. All mutation rates are initialized with $p_i = 10^{-4}$. These plots demonstrate the rise and fall of mutation rates, just as needed for reaching and then conserving the genetic mutations.

The COP is a separable objective, thus an easy to handle problem. If we switch to the following one

$$F_2(x) = \sum_{i=1}^{n-1} f(x_i) f(x_{i+1}) + f(x_n) f(x_1) \quad (4)$$

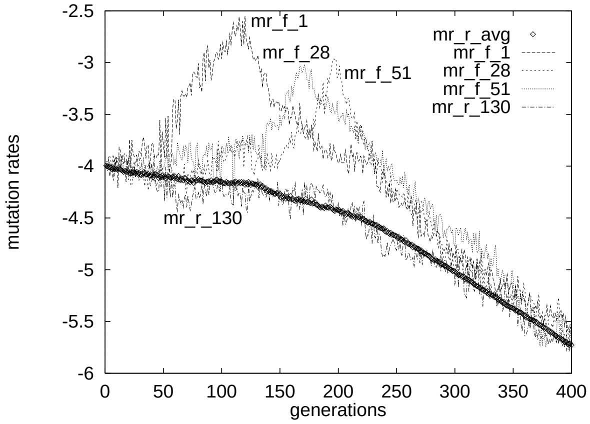

Figure 5: Self-adaptation of three mutation rates responsible for bits with a starting value of '0' $(mr - f_*)$, of the average mutation rate of those bits initialized correctly $(mr - r_{avg})$ and one mutation rate responsible for a bit with a correct starting value $(mr - r_{130})$.

and the corresponding F_2', the situation becomes more difficult insofar as now always two neighboring genes have to match with their optimal settings at the same time. Salomon[32] has shown that the effort to solve such problems with m-fold interdependencies increases with n^m in case of one common mutation rate $p = \frac{1}{n}$. With the concept of somatic mutations as above, the case $m = 2$ in F_2 is not so much more time consuming as to be expected according to[32]. The mutation rates belonging to such pairs behave correspondingly. No diagram is shown for this case. Instead, we directly turn to the most awful scenario with $m = n$, as is true for the following product sum:

$$F_3(x) = \prod_{i=1}^{n} f(x_i) \quad (5)$$

under otherwise same conditions. Now, all necessary genetic mutations must happen at the same time. If half of the genes are already matching, the probability of such a 'big jump' would be $(\frac{1}{n})^{\frac{n}{2}}(1 - \frac{1}{n})^{\frac{n}{2}}$ under a common mutation rate of $p = \frac{1}{n}$. At first sight, it seems nearly impossible to solve such a problem efficiently. Multicellular individuals with genetically encoded single mutation rates can do it by means of somatic mutations as shown in Figures 6 to 8.

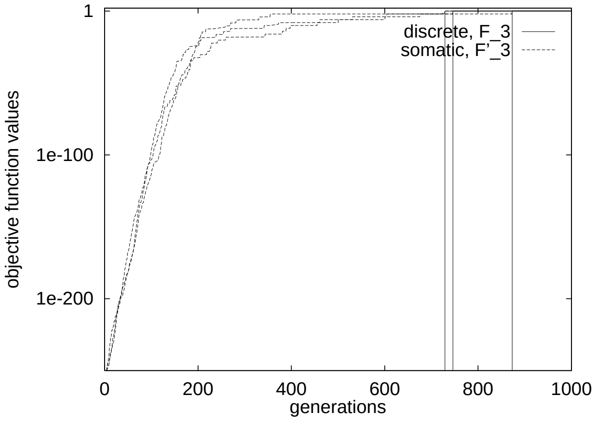

Figure 6: 3 runs of a $(15, 100)$ ES solving F_3.

Again, Fig. 6 presents a plot of both F_3 as well as F'_3 for the best individual over the number of generations (3 runs). Since there is just one genetic improvement from zero to one, no zoom is presented in this case. Instead, the number of already correct bits is shown as Fig. 7.

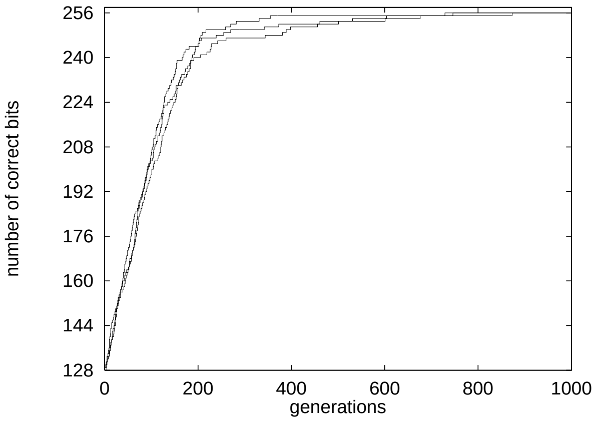

Figure 7: Number of correct bits corresponding to Fig. 6.

Fig. 8 shows selected mutation rates over time, similar to Fig. 5. Instead of three arbitrary mutation rates, those of the last three genes that still have to flip are shown, however.

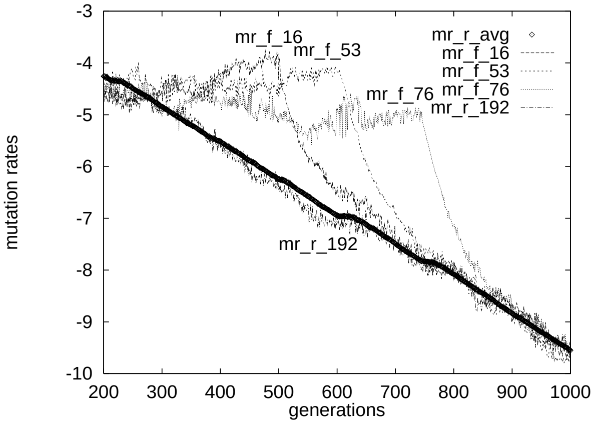

Figure 8: Self-adaptation of three mutation rates responsible for bits with a starting value of '0' $(mr - f_*)$, of the average mutation rate of those bits initialized correctly $(mr - r_{avg})$ and one mutation rate responsible for a bit with a correct starting value $(mr - r_{192})$.

A final remark after all the recommendations for EA above seems to be appropriate here: EA are not rivaling with traditional optimization methods. They cannot be more efficient than problem-specific solution procedures. Their rationale lies in the fact that it is often not economic to devise a special algorithm for just one new type of application. Then one may be better off with an optimum seeking procedure that uses no special knowledge and takes the situation at hand as a 'black-box' one – as all non-specialized EA do. Of course, domain-specific knowledge may be introduced in devising situated operators[33].

5 A Future for Evolutionary Computation?

Will the exponential growth of EC go on? No, at least not ultimately, since no exponential growth can last forever in a finite world. Thus, there are three possible futures: saturation, decline, or (mostly ir-)regular oscillations.

Applications of EA are likely to still grow during the next decade or so, since they do not yet have penetrated all reachable domains. Stagnation or decline will follow thereafter, depending on whether the thrust of the basic idea of *imitating life* will lead to even more efficacious algorithmic models of organic evolution or not. Currently, it seems as if the hunt for efficiency in particular problem solving situations drowns the search for better understanding and properly modeling real life. Only sporadically, new models emerge opening broader fields of applications. Most likely, this situation will yield further ups and downs of the field of EC as a whole.

On the other hand, there is a larger scope and potential for the *parallel problem solving from nature* paradigm. Besides phylogeny and ontogeny, there is the vast and not yet well understood realm of epigenesis, which may hold treasures of procedures worthwhile to be mimicked. The cooperative interplay between different cell types in the immune system, the social behavior of individuals in groups, and many more phenomena of real life still wait to be mimicked. They may be of use for evolvable hardware, for assemblies of autonomous hard- and software agents, for emergent computation, etc.

We should not dream, however, of machines that govern the world. Humans must remain the chief inspectors – even if they are not perfect. A machinery declared to be perfect (if it were as intelligent as a human only, it would not be perfect) could deprive us of some more evils, but also of our future, a necessary ingredient of which is uncertainty. Current experience with software technology lets us fear (hope?) that the software of an intelligent machine will never be perfect. Evolvability needs imperfection.

6 Summary

The test functions used in this study are so simple mathematically that one easily can devise more efficient solution methods than evolutionary algorithms. But this is not the point. Natural systems do not and cannot rely upon manipulations inspired analytically. They are groping in the dark. Nevertheless, they have found clever ways to find their way to top solutions. What we have shown above is the ability of a simple evolutionary algorithm to self-adapt internal strategy parameters like loci-specific mutation rates under harsh conditions like binary optimization, even with full interdependence of all variables (epistasis). Though none of the individuals knows about the landscape's model, the population searching collectively learns to adapt some kind of internal model, represented by mutabilities scaled properly. This is achieved by mimicking the trick of multicellularity. Whether this approach can be applied successfully to technical or other problems remains to be seen. That it has been successful in nature, e.g. in case of butterflies that mimic color patterns on their wings of non-savory examples to escape from predation, is without doubt.

Acknowledgments

The authors thank D. Goldberg, D. Fogel, and the reviewers for their useful hints to improve the paper.

References

[1] T. Bäck et al., *Handbook of Evolutionary Computation*, Oxford University Press, New York, 1997.

[2] H.–P. Schwefel, Parallel problem solving from nature, In *Encyclopedia of Computer Science and Technology*, A. Kent, J. G. Williams, C. M. Hall, Eds., Marcel Dekker, New York, 1997, vol. 37, suppl. 22, pp. 225–246.

[3] L. J. Fogel, A. J. Owens, and M. J. Walsh, *Artificial Intelligence through Simulated Evolution*, John Wiley & Sons, New York, 1966.

[4] D. B. Fogel, *Evolutionary Computation: Toward a New Philosophy of Machine Intelligence*, IEEE Press, New York, 1995.

[5] J. H. Holland, *Adaptation in natural and artificial systems*, The University of Michigan Press, Ann Arbor, MI, 1975.

[6] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Reading, MA, 1989.

[7] I. Rechenberg, *Evolutionsstrategie '94*, Frommann–Holzboog, Stuttgart, 1994 (enlarged edition of the PhD thesis of 1971).

[8] H.–P. Schwefel, *Evolution and Optimum Seeking*, John Wiley & Sons, New York, 1995 (enlarged translation of the PhD thesis of 1974/75).

[9] J. R. Koza, *Genetic Programming*, MIT Press, Cambridge, MA, 1992.

[10] J. T. Alander, *An Indexed Bibliography of Genetic Algorithms, preliminary edition*, J. T. Alander, Espoo, Finland, 1994.

[11] L. A. Zadeh, "Fuzzy sets", *Information and Control*, vol. 8, pp. 338-353, 1965.

[12] T. Blickle and L. Thiele, A mathematical analysis of tournament selection, In *Proc. 6th Int'l Conf. on Genetic Algorithms*, L. J. Eshelman, Ed., Morgan Kaufmann, San Mateo, CA, 1995, pp. 9–16.

[13] H. Mühlenbein and D. Schlierkamp–Voosen, "Predictive models for the breeder genetic algorithm I: Continuous parameter optimization", *Evolutionary Computation*, vol. 1, no. 1, pp. 25–49, 1993.

[14] H.–P. Schwefel, Missing features in current evolutionary algorithms, In *Evolutionary Algorithms and their Application*, V. Claus, J. Hopf, and H.-P. Schwefel, Eds., Dagstuhl-Seminar-Report No. 140, 1996, pp. 30–31.

[15] H.–P. Schwefel, Evolutionary computation — A study on collective learning, In *Proc. World Multiconference on Systemics, Cybernetics, and Informatics*, N. Callaos, C. M. Khoong, E. Cohen, Eds., Int'l Inst. of Informatics and Systemics, Orlando, FL, vol. 2, pp. 198–205.

[16] F. Kursawe, Naturanaloge Optimierverfahren — Neuere Entwicklungen in der Informatik, In *Studien zur Evolutorischen Ökonomik II*, U. Witt, Ed., Duncker & Humblot, Berlin, Schriften des Vereins für Socialpolitik, Band 195/II, 1992, pp. 11–38.

[17] G. Rudolph, *Convergence Properties of Evolutionary Algorithms*, Verlag Dr. Kovač, Hamburg, 1997.

[18] J. Maynard Smith, "Models of evolution", *Proc. Royal Soc. London B*, vol. 219, pp. 315–325, 1983.

[19] H.–G. Beyer, "Toward a theory of evolution strategies: On the benefit of sex – the $(\mu/\mu, \lambda)$ theory", *Evolutionary Computation*, vol. 3, pp. 81–111, 1996.

[20] F. Kursawe, Evolution strategies for vector optimization, In *Proc. 10th Int'l Conf. on Multiple Criteria Decision Making, Tapei, Taiwan, July 19-24, 1992*, vol. 3, pp. 187–193.

[21] K. Gärtner, Personal communication.

[22] G. Rudolph, Parallel approaches to stochastic global optimization, In *Parallel Computing: From Theory to Sound Practice*, W. Joosen and E. Milgrom, Eds., IOS Press, Amsterdam, 1992, pp. 256–267.

[23] J. Sprave, Linear neighborhood evolution strategy, In *Proc. 3rd Annual Conf. on Evolutionary Programming*, A. V. Sebald and L. J. Fogel, Eds., World Scientific, Singapore, 1992, pp. 42–51.

[24] T. Bäck, *Evolutionary Algorithms in Theory and Practice*, Oxford University Press, New York, 1996.

[25] H. Mühlenbein, How genetic algorithms really work: I. Mutation and hill-climbing, In R. Männer and B. Manderick, Eds., *Parallel Problem Solving from Nature 2*, Elsevier, Amsterdam, 1992, pp. 15–25.

[26] S. Droste, T. Jansen, and I. Wegener, "A rigorous complexity analysis of the (1+1) evolutionary algorithm for separable functions with boolean inputs", Technical Report No. CI-6/97, University of Dortmund, Collaborative Research Center (SFB) 531, June 1997.

[27] T. Bäck and H.–P. Schwefel, "An overview of evolutionary algorithms for parameter optimization", *Evolutionary Computation*, vol. 1, pp. 1–23, 1993.

[28] S. Kauffman, *At Home in the Universe*, Oxford University Press, New York, 1995.

[29] K. Gärtner, "A third component causing random variability beside environment and genotype. A reason for the limited success of a 30

year long effort to standardize laboratory animals?" *Laboratory Animals*, vol. 24, pp. 71–77, 1990.

[30] H. –P. Schwefel, Binäre Optimierung durch somatische Mutation, Technical Report of the 'Zentrales Tierlaboratorium und Abteilung für Versuchstierkunde der Medizinischen Hochschule Hannover' and the 'Fachgebiet Bionik and Evolutionstechnik im Institut für Meß- und Regelungstechnik der Technischen Universität Berlin', Final report for the period August–November 1974, sponsored by the 'Sonderforschungsbereich 146 Versuchstierforschung der Tierärztlichen Hochschule Hannover', May 1975.

[31] G. E. Hinton and S. J. Nowlan, "How learning can guide evolution", *Complex Systems*, vol. 1, no. 3, pp. 495–502, 1987.

[32] R. Salomon, "Some comments on evolutionary algorithm theory", *Evolutionary Computation*, vol. 4, pp. 405–415, 1996.

[33] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*, Springer, Berlin, 1992.