Demonstrating the Evolution of Complex Genetic Representations: An Evolution of Artificial Plants

Marc Toussaint

Institut für Neuroinformatik, Chair for Theoretical Biology
Ruhr-Universität Bochum ND-04, 44780 Bochum, Germany
toussaint@neuroinformatik.rub.de

Abstract. A common idea is that complex evolutionary adaptation is enabled by complex genetic representations of phenotypic traits. This paper demonstrates how, according to a recently developed theory, genetic representations can self-adapt in favor of evolvability, i.e., the chance of adaptive mutations. The key for the adaptability of genetic representations is neutrality inherent in non-trivial genotype-phenotype mappings and neutral mutations that allow for transitions between genetic representations of the same phenotype. We model an evolution of artificial plants, encoded by grammar-like genotypes, to demonstrate this theory.

1 Introduction

In ordinary evolutionary systems, including natural evolution and standard Genetic Algorithms (GAs), the search strategy is determined by uncorrelated mutations and recombinations on the level of genes. In view of this trial-and-error search strategy, it might seem surprising that natural evolution was that effective in finding highly *structured*, complex solutions. One might have expected that the search strategy needs to be more sophisticated, more structured and adapted to the "objective function" to accomplish this efficiency.

An example for sophisticated, adapted, and structured search strategies are recent developments in evolutionary computation, which can be collected under the name of *Estimation-of-Distribution Algorithms* (EDAs, see [10] for a review). These algorithms structure search by learning probabilistic models of the distribution of good solutions.

Now, it hardly seems that ordinary evolutionary systems like natural evolution or GAs learn "models about the distribution of good solutions" and adapt their mutational exploration accordingly. However, it is possible for ordinary evolutionary systems to adapt their search strategy and natural evolution certainly exploits this possibility. The key to understand this possibility is to consider a non-trivial relation between genotype and phenotype, between the genes and the phenes (phenotypic traits that are fitness-relevant). In this case it is possible that the same phenotype can be represented by different genotypes. In other words, evolution could principally adapt the way it represents solutions.

Adapting the genetic representation of phenes while keeping mutation operators fixed is in some sense dual to adapting the mutation operators while keeping

E. Cant'u-Paz et al. (Eds.): GECCO 2003, LNCS 2723, pp. 86–97, 2003.

the representation of phenes fixed. Especially in the biology literature it is often argued that this is the way natural evolution adapts its search strategy: The way genes encode phenotypic traits is surely not an incident but the outcome of a long adaptative process in favor of evolvability [15]. Much research is spend to understand these principles of gene interactions (like epistasis or canalization) [1, 16, 17, 4, 5].

But there would be another open question. Do genetic representations really adapt such that the search strategy becomes "better"? Is the induced adaptation of the search strategy similar to the adaptation in the case of EDAs, such that indeed a model of the distribution of good solutions is indirectly learned by adapting the representations of solutions? Recently, a theory on the adaptation of phenotypic search distributions (*exploration distributions*) in the case of non-trivial genotype-phenotype mappings was proposed [14]. Since the present paper aims to illustrate this process and thereby to open a more intuitive access to this theory, we want, at this place, briefly review the main results.

A non-trivial genotype-phenotype mapping is such that (1) the same phenotype is encoded by more than one genotype, and (2), at least for some of these phenotypes, the induced phenotypic variability depends on the genotype it is encoded with. In this case, every genotype g may be written as a tuple (x, σ), where x is its phenotype and σ generally represents any kinds of *neutral traits* of that genotype (strategy parameters are only a special case). Different σ for the same x mean different genetic representations of the same phenotype. In [14] it is shown that a σ is one-to-one identifiable with a mutation distribution (i.e., search distribution) "around" g. Hence, the tuple (x, σ) actually lives in the product space of phenotype space and the space of distributions.

Now, the main approach is the following: Instead of investigating the evolution of genotypes g in a given evolutionary process, one can also analyze how x and σ evolve in that same process. The most interesting result is that the equation that describes the evolution of σ's turns out to be itself an equation similar to evolutionary dynamics. The equation allows to associate a quality measure to each σ and says that σ's evolve as to increase this quality measure just as normal evolutionary processes evolve as to increase the fitness of genotypes. This quality measure of σ's can be interpreted as a measure of "how good the mutational phenotypic variability (the phenotypic search distribution) matches the distribution of good organisms"—just as for EDAs. The results are summarized in following corollary:

Corollary 1 ([14]). *The evolution of genetic representations, or equivalently, of exploration distributions (σ-evolution) naturally has a selection pressure towards*

- *minimizing the Kullback-Leibler divergence between exploration and the exponential fitness distribution, and*
- *minimizing the entropy of exploration.*

Here, the exponential fitness distribution describes the distribution of good solutions and the Kullback-Leibler divergence is a distance measure between two distributions and thus measures the match between the search strategy (as given by the exploration distribution) and this exponential fitness distribution.

This paper will present a first case study for the evolution of genetic representations by modeling an evolution of artificial plants. We chose this case study for several reasons: First, the genotype-phenotype mapping that we will introduce shortly is highly non-trivial in the strict sense we defined it for σ-evolution, and it is interpretable from an (abstract) biological point of view. Further, this kind of encoding and the possibility to visually display the phenotypes allows to really grasp the genetic encodings, how completely different genetic representations of the same phenotype are possible, that complex features like gene interactions and modularity of the genetic representations can emerge.

The next section introduces the genetic encoding and the mutability we assume on this encoding. The major novelty here are 2nd-type mutations that allow for neutral transitions between equivalent genetic representations within the grammar-like encoding. Section 3 then explains the experimental setup and presents the experiments that demonstrate the evolution of complex genetic representations. Conclusions follow thereafter.

2 A Genotype Model and 2nd-Type Mutations

The genotype-phenotype mapping we will assume is a variant of the L-systems proposed by [11] to encode plant-like structures. Let us introduce this encoding from another point of view that puts more emphasis on the principles of ontogenesis and gene interactions.

Let us consider the development of an organism as an interaction of its state Ψ (describing, e.g., its cellular configuration) and a genetic system Π, neglecting environmental interactions. Development begins with an initial organism in state $\Psi^{(0)}$, the "egg cell", which is also inherited and which we model as part of the genotype. Then, by interaction with the genetic system, the organism develops through time, $\Psi^{(1)}, \Psi^{(2)}, .. \in P$, where P is the space of all possible organism states. Hence, the genetic system may be formalized as an operator $\Pi : P \to P$ modifying organism states such that $\Psi^{(t)} = \Pi^t \Psi^{(0)}$. We make a specific assumption about this operator: we assume that Π comprises a whole sequence of operators, $\Pi = \langle \pi_1, \pi_2, .., \pi_r \rangle$, each $\pi_i : P \to P$. A single operator π_i (also called production rule) is meant to represent a transcription module, i.e., a single gene or an operon. Based on these ideas we define the general concept of a genotype and a genotype-phenotype mapping for our model:

A genotype consists of an initial organism $\Psi^{(0)} \in P$ and a sequence $\Pi = \langle \pi_1, \pi_2, .., \pi_r \rangle$, $\pi_i : P \to P$ of operators. A genotype-phenotype mapping ϕ develops the final phenotype $\Psi^{(T)}$ by recursively applying all operators on $\Psi^{(0)}$.

This definition is somewhat incomplete because it does not explain the stopping time T of development and in which order operators are applied. We keep to the simplest options: we apply the operators in sequential order and will fix T to some chosen value.

For the experiments we need to define how we represent an organism state Ψ and how operators are applied. We represent an organism by a sequence of symbols $\langle \psi_1, \psi_2, .. \rangle$, $\psi_i \in \mathcal{A}$. Each symbol may be interpreted, e.g., as the state of a cell; we choose the sequence representation as the simplest spatially organized assembly of such states. Operators are represented as replacement rules $\langle a_0 {:} a_1, a_2, .. \rangle$, $a_i \in \mathcal{A}$, that apply on the organism by replacing all occurrences of a symbol a_0 by the sequence $\langle a_1, a_2, .. \rangle$. If the sequence $\langle a_1, a_2, .. \rangle$ has length greater than 1, the organism is growing; if it has length 0, the organism shrinks. Calling a_0 promoter and $\langle a_1, a_2, .. \rangle$ the structural genes gives the analogy to operons in natural genetic systems. For example, if the initial organism is given by $\Psi^{(0)} = \langle \mathsf{a} \rangle$ and the genetic system is $\Pi = \big\langle \langle \mathsf{a{:}ab} \rangle, \langle \mathsf{a{:}cd} \rangle, \langle \mathsf{b{:}adc} \rangle \big\rangle$, then the organism grows as: $\Psi^{(0)} = \langle \mathsf{a} \rangle$, $\Psi^{(1)} = \langle \mathsf{cdadc} \rangle$, $\Psi^{(2)} = \langle \mathsf{cdcdadcdc} \rangle$, etc.

The general idea is that these operators are basic mechanisms which introduce correlating effects between phenotypic traits. [13] already claimed that the essence of the operon is to introduce correlations between formerly independent genes in order to adopt the *functional* dependence between the genes and their phenotypic effects and thereby increase the probability of successful variations.

The proposed model is very similar to the models by [8], [3], and [9], who use grammar-encodings to represent neural networks. It is also comparable to new approaches to evolve complex structures by means of so-called *symbiotic composition* [6, 7, 18].

The crucial novelty in our model are *2nd-type mutations* . These allow for genetic variations that explore the neutral sets which are typical for any grammar-like encoding. Without these neutral variations, self-adaptation of genetic representations and of exploration distributions is not possible.

Consider the three genotypes given in the first column,

genotype	phenotype	phenotypic neighbors
$\Psi^{(0)} = \langle \mathsf{a} \rangle$, $\Pi = \big\langle \langle \mathsf{a{:}bcbc} \rangle \big\rangle$	$\langle \mathsf{bcbc} \rangle$	$\langle * \rangle$, $\langle \mathsf{*cbc} \rangle$, $\langle \mathsf{b*bc} \rangle$, $\langle \mathsf{bc*c} \rangle$, $\langle \mathsf{bcb*} \rangle$
$\Psi^{(0)} = \langle \mathsf{a} \rangle$, $\Pi = \big\langle \langle \mathsf{a{:}dd} \rangle, \langle \mathsf{d{:}bc} \rangle \big\rangle$	$\langle \mathsf{bcbc} \rangle$	$\langle * \rangle$, $\langle \mathsf{*bc} \rangle$, $\langle \mathsf{bc*} \rangle$, $\langle \mathsf{*c*c} \rangle$, $\langle \mathsf{b*b*} \rangle$
$\Psi^{(0)} = \langle \mathsf{bcbc} \rangle$, $\Pi = \langle\ \rangle$	$\langle \mathsf{bcbc} \rangle$	$\langle \mathsf{*cbc} \rangle$, $\langle \mathsf{b*bc} \rangle$, $\langle \mathsf{bc*c} \rangle$, $\langle \mathsf{bcb*} \rangle$

All three genotypes have, after developing for at least two time steps, the same phenotype $\Psi^{(t)} = \langle \mathsf{bcbc} \rangle$, $t \geq 2$. The third genotype resembles what one would call a direct encoding, where the phenotype is directly inherited as $\Psi^{(0)}$. Assume that, during mutation, all symbols, except for the promoters, mutate with fixed, small probability. By considering all one-point mutations of the three genotypes, we get the phenotypic neighbors of $\langle \mathsf{bcbc} \rangle$ as given in the third column of the table, where a star $*$ indicates the mutated random symbol. This shows that, although all the three genotypes represent the same phenotype, they induce completely different phenotypic variabilities, i.e., completely different phenotypic search distributions. Note that, wherever a phenotypic neighbor comprises two stars, these two phenotypic variations are completely correlated.

In order to enable a variability of genetic representations within such a neutral set we need to allow for mutational transitions between phenotypically equivalent genotypes. A transition from the 1st to the 3rd genotype requires a genetic mutation that applies the operator $\langle \mathsf{a{:}bcbc} \rangle$ on the egg cell $\langle \mathsf{a} \rangle$ and deletes it thereafter. Both, the application of an operator on some sequence (be it the egg cell or another operator's) and the deletion of operators will be mutations we provide in our model. The transition from the 2nd to the 1st genotype is similar: the 2nd operator $\langle \mathsf{d{:}ab} \rangle$ is applied on the sequence of the first operator $\langle \mathsf{a{:}dd} \rangle$ and deleted thereafter. But we must also account for the inverse of these transitions. A transition from the 3rd genotype to the 1st is possible if a new operator is created by randomly extracting a subsequence (here bcbc from the egg cell) and encoding it in a new operator (here $\langle \mathsf{a{:}bcbc} \rangle$). The original subsequence is then replaced by the promoter. Similarly, a transition from the 1st to the 2nd geno-

Table 1. The mutation operators in our model.

- First type mutations are ordinary symbol mutations that occur in every sequence (i.e., promoter or rhs of an operator) of the genotype; namely symbol replacement, symbol duplication, and symbol deletion, which occur with equal probabilities. The mutation frequency for every sequence is Poisson distributed with the mean number of mutations given by ($\alpha \cdot$ sequence-length), where α is a global mutation rate parameter.
- The second type mutations aim at a neutral restructuring of the genetic system. A 2nd-type mutation occurs by randomly choosing an operator π and a sequence p from the genome, followed by one of the following operations:
  - application of an operator π on a sequence p,
  - inverse application of an operator π on a sequence p; this means that all matches of the operator's rhs sequence with subsequences of p are replaced in p by the operator's promoter,
  - deletion of an operator π, only if it was never applied during ontogenesis,
  - application of an operator π on *all* sequences of the genotype followed by deletion of this operator,
  - generation of a new operator ν by extracting a random subsequence of stochastic length $2 + \text{Poisson}(1)$ from a sequence p and encoding it in an operator with random promoter. The new operator ν is inserted in the genome behind the sequence p, followed by the inverse application of ν on p.

  All these mutations occur with equal probabilities. The total number of second type mutations for a genotype is Poisson distributed with mean β. The second type mutations are not necessarily neutral but they are neutral with sufficient probability to enable an exploration of neutral sets.
- A genotype is mutated by first applying second type mutations and thereafter first type mutations, the frequencies of which are given by β and α, respectively.

type occurs when the subsequence bc is extracted from the operator ⟨a:bcbc⟩ and encoded in a new operator ⟨d:bc⟩.

Basically, the mutations we will provide in our model are the generation of new operators by extraction of subsequences (*deflation*), and the application and deletion of existing operators (*inflation*). Technical details can be found in table 1; the main point of these mutation operators though is not their details but that they in principle enable a transition between phenotypically equivalent representations in our encoding. We call these mutations 2nd-type mutations to distinguish them from ordinary symbol mutation, deletion, and insertion.

3 Evolving Plants

The symbols for encoding plants. The sequences we are evolving are strings of the alphabet {A,..,P} which are mapped on plant-describing strings according to {A,..,I} ↦ {`F,+,-,&,^,\,/,[,]`} and {J,..,P} ↦ {`.`}. The meanings of the symbols of the plant-describing strings are summarized in table 2. For example, the sequence ⟨`FF[+F][-F]`⟩ represents a plant for which the stem grows two units upward before it branches in two arms, one to the right, the other to the left, each of which has one unit length and a leave attached at the end.

Table 3 demonstrates the implications of this encoding. The plant on the left is an examples taken from [12]. To the right of this "original" you find three

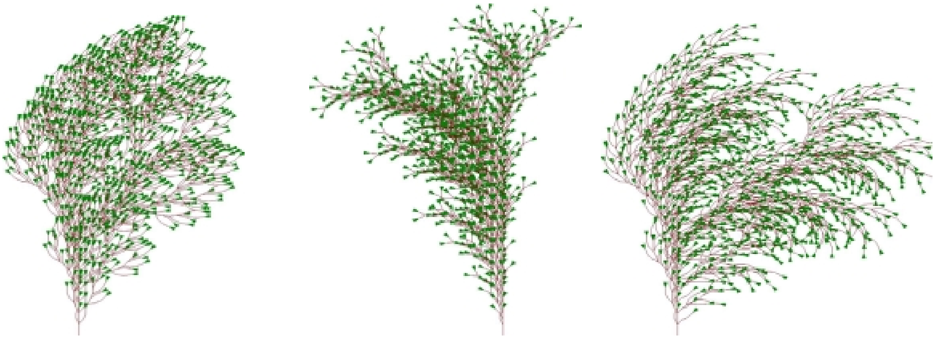
Table 2. Description of the plant grammars symbols, cf. [11].

`F`	attach a unit length stem

`+,-,&`
`^,\,/`
	rotations of the "local coordinate system" (that apply on following attached units): four rotation of δ degree in all four directions away from the stem (`+,-,&,^`) and two rotations of $\pm\delta$ degree around the axis of the stem (`\,/`).

`[`	a branching (instantiation of a new local coordinate system)
`]`	the end of that branch: attaches a leave and proceeds with the previous branch (return to the previous local coordinate system); also if the very last symbol of the total sequence is not `]`, another leave is attached
`.`	does nothing

Table 3. An example for a 2D plant structure and its phenotypic variability induced by *single* symbol mutations in the genotype. For all of them $T=4$, $\delta=20$, and $\Psi^{(0)}=\langle\texttt{F}\rangle$. Below each illustration, the single production rule that was mutated is given.



⟨`F:FF-[-F+F+F]+[+F-F-F]`⟩ ⟨`F:FF-[+F+F+F]+[+F-F-F]`⟩ ⟨`F:FF-[-F-F+F]+[+F-F-F]`⟩

different variations, each of which was produced by a *single* symbol mutation in the genotype. Obviously, there are large-scale correlations in the phenotypic variability induced by uncorrelated genetic mutations.

The fitness function. Given a sequence we evaluate its fitness by first drawing the corresponding 3D plant in a virtual environment (the OpenGL 3D graphics environment). We chop off everything of the plant that is outside a bounding cube of size $b \times b \times b$. Then we grab a bird's view perspective of this plant and measure the area of green leaves as observed from this perspective. The measurement is also height dependent: the higher a leave (measured by OpenGL's depth buffer in logarithmic scale, where 0 corresponds to the cube's floor and 1 to the cube's ceiling), the more it contributes to the green area integral

$$L = \int_{x \in \text{bird's view area}} \Big[\text{color of } x = \text{green}\Big] \cdot \Big[\text{height at } x\Big] \, \frac{d\text{Area}}{b^2} \quad \in [0,1]\,. \qquad (1)$$

This integral is the positive component of a plant's fitness. The negative component is related to the number and "weight" of branch elements: To each element i we associate a weight w_i which is defined recursively. The weight of a leave is 1; the total weight of a subtree is the sum of weights of all the elements of that subtree; and the weight of a branch element i is 1 plus the total weight of the subtree that is attached to this branch element. E.g., a branch that has a single leave attached has weight $1+1=2$, a branch that has two branches each with a single leave attached has weight $1+(2+1)+(2+1)=7$, etc. The idea is that w_i

roughly reflects how “thick” i has to be in order to carry the attached weight. The total weight of the whole tree,

$$W = \sum_i w_i \ ,$$

gives the negative component of a plant's fitness. In our experiments we used

$$f = L \ - \ \varrho W$$

as the fitness function, where the penalty factor ϱ was chosen $\varrho \in \{10^{-6}, 10^{-7}\}$ in the different experiments.

Details of the implementation. Evolving such plant structures already gets close to the limits of today's computers, both, with respect to memory and computation time. Hence, we use some additional techniques to improve efficiency: First, we impose different limits on the memory resource that a genotype and phenotype is allowed to allocate, namely three: (1) The number of symbols in a phenotype was limited to be lower or equal than a number $M_{\max}$. This holds in particular during ontogenesis: If the application of an operator results in a phenotype with too many symbols, then the operator is simply skipped. (2) The number of operators in a genotype is limited to be $\leq R_{\max}$. If a mutation operator would lead to more chromosomes, this mutation operator is simply skipped (no other mutation is made in place). (3) There is a soft limit on the number of symbols in a single chromosome: A duplication mutation is skipped if the chromosome already has length $\geq U_{\max}$. The limit though does not effect 2nd-type mutations; an inflative mutation $\pi \cdot p$ may very well lead to chromosomes of length greater than $U_{\max}$.

Second, we adopt an elaborated technique of self-adaptation of the mutation frequencies. We used the scheme similar to the self-adaptation of strategy parameters proposed by [2]. Every genome i additionally encodes two real numbers α_i and β_i. Before any other mutations are made, they are mutated by

$$\begin{aligned} \alpha_i &\leftarrow \alpha_i \left(\mathcal{S}\,\mathcal{N}(0,\tau) + \tau' \right) \ , \\ \beta_i &\leftarrow \beta_i \left(\mathcal{S}\,\mathcal{N}(0,\tau) + \tau' \right) \ , \end{aligned} \tag{2}$$

where $\mathcal{S}\,\mathcal{N}(0,\tau)$ is a random sample (independently drawn for α_i and β_i) from the Gaussian distribution $\mathcal{N}(0,\tau)$ with zero mean and standard deviation τ. The parameter τ' allows to induce a pressure towards increasing mutation rates. After this mutation, α_i and β_i determine the mutation frequencies of 1st- and 2nd-type mutations respectively.

The 1st trial. Let us discuss two of the trials made with different parameters. Table 4 summarizes the experimental setup. See figure 1. For the 1st trial, the curves show some sudden changes at generation ∼4000 where the fitness, the number of phenotypic elements, the number of operators in the genomes, and the total genome length explodes. Between generation ∼4000 and ∼5400, the most significant curve in the graph is the repeatedly decaying genome size. Indeed we will find that the genomes in this period are too large and mutationally unstable. The innovations extinct and genome size decays until at generation ∼5400 a comparably large number of phenotypic elements can be encoded by much smaller genomes that comprise more operators. In table 5, the illustrations of

Table 4. Parameter settings of the two trials. The "$\hookleftarrow$" means "same as for the previous trial" and shows that only few parameters are changed from trial to trial.

	1st trial	2nd trial	
δ, b	20,20	$\hookleftarrow$	L-system angle δ, size of the bounding cube
T	1	$\hookleftarrow$	stopping time of development
ϱ	10^{-7}	10^{-6}	factor of the weight term W in a plant's fitness f
α	.01	$\hookleftarrow$	(initial) frequency of first type mutations
β	.01	.3	(initial) frequency of second type mutations
τ (τ')	.5 (.1)	0 (0)	rate of self-adaptation of α and β
	μ, λ	$\hookleftarrow$	type of selection
	no	$\hookleftarrow$	crossover turned on?
	$\langle$AAAFFFJ$\rangle$	$\hookleftarrow$	initialization of $\Psi^{(0)}$ of all genotypes in the first population (they have no operators)
$\mathcal{A}$	{A,..,P}	$\hookleftarrow$	symbol alphabet
λ	100	$\hookleftarrow$	(offspring) population size
μ	30	$\hookleftarrow$	(selected) parent population size
$M_{\max}$	100 000	1 000 000	maximal number of symbols in a phenotype
$R_{\max}$	100	$\hookleftarrow$	maximal number of operators allowed in one genotype
$U_{\max}$	40	$\hookleftarrow$	symbol duplication mutations are allowed only if a sequence has less than $U_{\max}$ symbols

the best individual in selected generations explain in more detail what happened. For a long time this is not much until, in generation 4000, a couple of leaves turn up at certain places of the phenotype. Then, very rapidly, more leaves pop up until, in generation 4025, every phenotypic segment has a leave attached. This is exactly what we call a correlated phenotypic adaptation and was enabled by encoding all the segments that now carry a leave within one operator, namely the A-operator. The resulting "long-arm-building-block" triggers a revolution in phenotypic variability and leads to the large structures as illustrated for generation 4400 (3467 elements). However, these structures are not encoded efficiently, the genome size is too large (512) and phenotypic variability becomes chaotic. The species almost extinguishes until, in generation 5100, evolution finds a much better structured genome to encode large phenotypes. The J-operator becomes dominant and allows to encode 1479 phenotypic elements with a genome size of 217. This concept is further improved and evolves until, in generation 8000, a genome of size 141 with 2 operators encodes a regularly structured phenotype of 3652 elements.

The 2nd trial. For the 2nd trial we turned off the self-adaptation mechanism for the mutation frequencies (based on the experience with previous trials we can now estimate a good choice of $\alpha = .01$ and $\beta = .3$ for the mutation frequencies and fix it) and increase the limit $M_{\max}$ to maximally 1 000 000 elements per phenotype. The severe change in the resulting structures is also due to the increase of the weight penalty factor ϱ to 10^{-6}—the final structure of the 1st trial has a weight of about $.3 \cdot 10^{-6}$ which would now lead to a crucial penalty. The weight punishing factor ϱ enforces structures that are regularly branched instead of long curling arms. Table 6 presents the results of the 2nd trial. Comparing the illustrations for generation 950 and 1010 we see that evolution very quickly developed a fan-like structure that is attached at various places of the

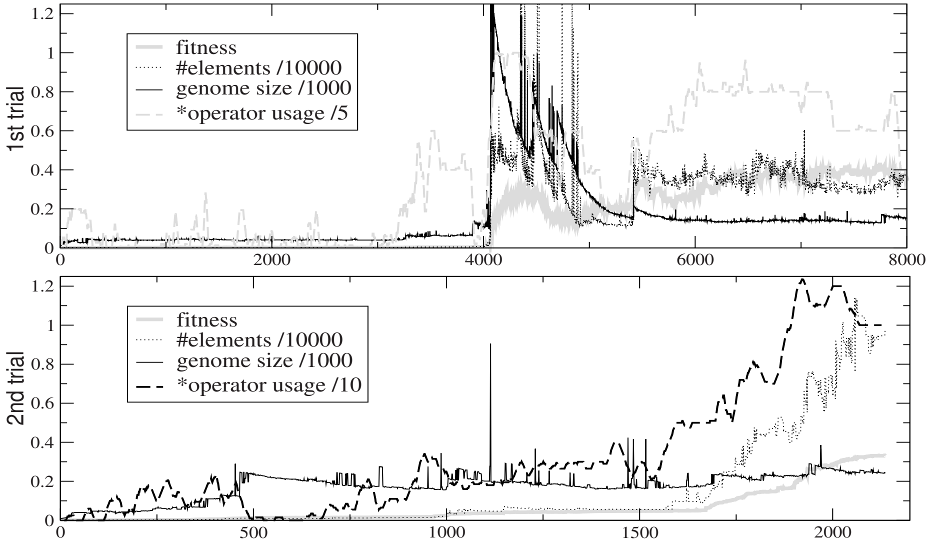


Fig. 1. The graphs display the curves of the fitness, the number of phenotypic elements, the genome size, and the operators usage of the best individual in all generations of the trials. Note that every quantity has been rescaled to fit in the range of the ordinate, e.g., the number of phenotype elements has been divided by 10 000 as indicated by the notation "#elements/10000". The * for the operator usages indicates that the curve has been smoothed by calculating the running average over an interval of about a hundredth of the abscissae (80 and 20 in the respective graphs).

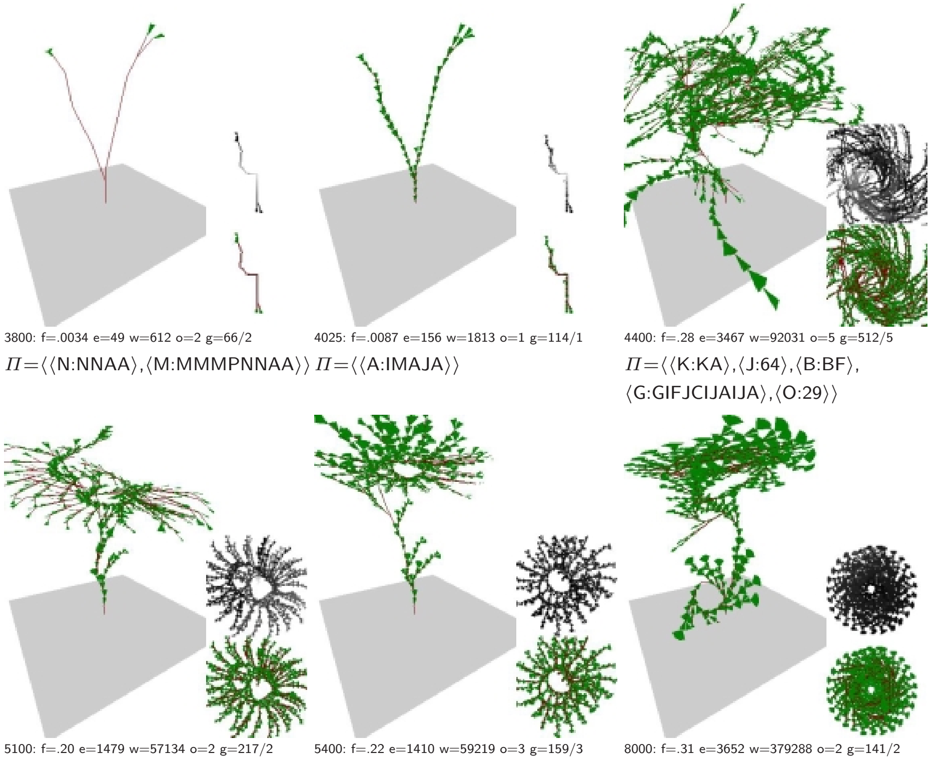
phenotype. The fans arise from an interplay of two operators: The N-operator encodes the fan-like structures while the F-operator encodes the spokes of these fans. Adaptation of these fans is a beautiful example for correlated exploration. The N-operator encodes more and more spokes until the fan is complete in generation 1010, while the F-operator makes the spokes longer. Elongation proceeds and results in the "hairy", long-armed structures. Note that, in generation 1650, one N- and two B-operators are redundant. Until generation 1900, leaves are attached to each segment of the arms, similar to generation 4025 of the 1st trial. At that time, the plant's weight is already 105 099 and probably prohibits to make the arms even longer (since weight would increase exponentially). Instead a new concept develops: At the tip of each arm two leaves are now attached instead of one and this quickly evolves until there are three leaves, in generation 1910, and eventually a complete fan of six leaves attached at the tip of each arm. In generation 2100, a comparably short genome with 10 used operators encodes a very dense phenotype structure of 9483 elements.

More trials, data, and source code can be found at the author's home page.

4 Conclusions

Let us briefly discuss whether similar result could have been produced with a more conventional GA that uses, instead of our non-trivial genotype-phenotype mapping, a direct encoding of sequences in {F,+,-,&,^,\,/,[,]} that describe the plants. For example, setting $\beta = 0$ in our model corresponds to such a

Table 5. The 1st trial. The illustrations display the phenotypes at selected generations. The two squared pictures in the lower right corner of each illustration display exactly the bird's view perspective that is used to calculate the fitness: The lower (colored) picture displays the plant as seen from above and determines which area enters the green area integral in equation (1), and the upper gray-scale picture displays the height value of each element which enters the same equation (where white and black refer to height 0 and 1, respectively). Below each illustration you find some data corresponding to this phenotype: ⟨generation⟩: f=⟨fitness⟩ e=⟨number of elements⟩ w=⟨plant's total weight⟩ o=⟨number of used operators⟩ g=⟨genome size⟩/⟨number of operators in the genome⟩. The genetic system Π is also displayed. ($\Psi^{(0)}$ is generally too large to be displayed here.) For some operators, the size of the rhs is given instead of the sequence. See the text for a discussion of this evolution.



3800: f=.0034 e=49 w=612 o=2 g=66/2
Π=⟨⟨N:NNAA⟩,⟨M:MMMPNNAA⟩⟩

4025: f=.0087 e=156 w=1813 o=1 g=114/1
Π=⟨⟨A:IMAJA⟩⟩

4400: f=.28 e=3467 w=92031 o=5 g=512/5
Π=⟨⟨K:KA⟩,⟨J:64⟩,⟨B:BF⟩,
⟨G:GIFJCIJAIJA⟩,⟨O:29⟩⟩

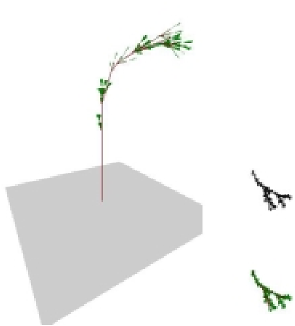
5100: f=.20 e=1479 w=57134 o=2 g=217/2
Π=⟨⟨K:⟩,⟨J:77⟩⟩

5400: f=.22 e=1410 w=59219 o=3 g=159/3
Π=⟨⟨I:AJ⟩,⟨C:JJCJ⟩,⟨J:64⟩⟩

8000: f=.31 e=3652 w=379288 o=2 g=141/2
Π=⟨⟨I:39⟩,⟨J:60⟩⟩

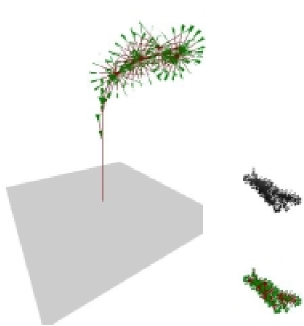
GA since no operators will be created and the evolution takes places solely on the "egg cell" $\Psi^{(0)}$, which is equal to the final phenotype in the absence of operators. We do not need to present the results of such a trial—not much happens. The obvious reason is the unsolvable dilemma of long sequences in a direct encoding: On the one hand, mutability must be small such that long sequences can be represented stably below the error threshold of reproduction; on the other hand mutability should not vanish in order to allow evolutionary progress. This dilemma becomes predominant when trying to evolve sequences of length $\sim 10^4$, as it is the case for the plants evolved in the 2nd trial. Also elaborate methods of self-adaptation of the mutation rate cannot circumvent

Table 6. The 2nd trial. Please see the caption of table 5 for explanations.



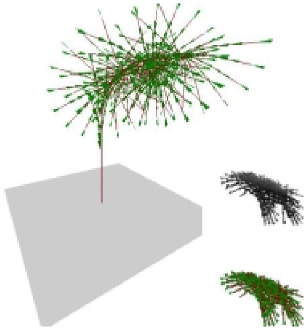
950: f=.018 e=144 w=2681 o=3 g=163/3

Π=⟨⟨N:IP⟩,⟨P:JF⟩,⟨F:IIAJJF⟩⟩



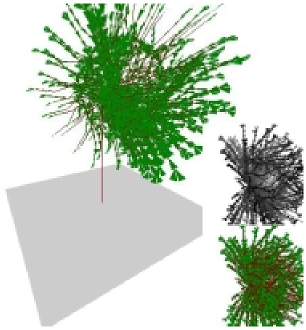
1010: f=.032 e=506 w=10250 o=2 g=211/2

Π=⟨⟨N:NIJFFFFFFFFFF⟩, ⟨F:IIAAJJF⟩⟩



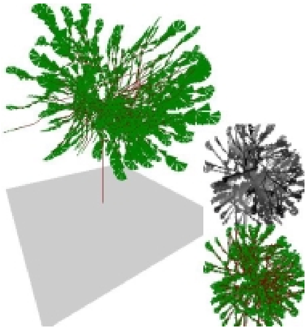
1650: f=.052 e=1434 w=31476 o=5 g=180/8

Π=⟨⟨B:IJNN⟩,⟨N:IAAJFIAAJF⟩, ⟨N:4⟩,⟨B:3⟩,⟨B:9⟩,⟨F:IINFNNKK KCBJF⟩,⟨B:IJNNAJ⟩,⟨N:IJBA⟩⟩



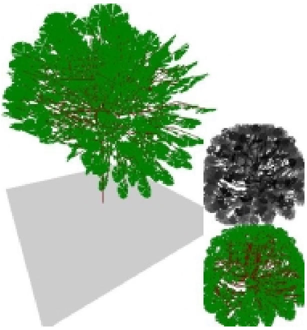
1900: f=.17 e=4915 w=105099 o=10 g=230/12

Π=⟨⟨B:NN⟩,⟨N:IABFIAAJF⟩,⟨N: 33⟩,⟨J:MOFJ⟩,⟨D:KNME⟩,⟨B:BB BBJ⟩,⟨B:FB⟩,⟨B:MNNLDDM⟩,⟨F: 32⟩,⟨B:28⟩,⟨N:NLA⟩,⟨L:IJ⟩⟩



1910: f=.20 e=4340 w=89996 o=12 g=226/14

Π=⟨⟨B:NN⟩,⟨N:IABFIAAJF⟩,⟨N: 33⟩,⟨J:MMJ⟩,⟨J:NJ⟩,⟨D:KNME⟩, ⟨B:BBBBJ⟩,⟨B:FB⟩,⟨B:MNNLD DM⟩,⟨F:30⟩,⟨B:18⟩,⟨C:KCKKC LK⟩,⟨N:NLA⟩,⟨L:IJ⟩⟩



2100: f=.33 e=9483 w=192235 o=10 g=261/15

Π=⟨⟨B:NN⟩,⟨N:IAABFIAAJF⟩, ⟨N:57⟩,⟨J:B⟩,⟨J:JJ⟩,⟨D:DOIE⟩, ⟨B:FBFBFBFBHJ⟩,⟨B:ENNDD⟩, ⟨F:36⟩,⟨B:25⟩,⟨B:6⟩,⟨C:CCGLB⟩, ⟨B:CLK⟩,⟨N:NNLA⟩,⟨L:IJ⟩⟩

this problem completely; the only way to solve the dilemma is to allow for an adaptation of genetic representations. The key novelty in our model that enabled the adaptation of genetic representations are the 2nd-type mutations we introduced.

In our example, two important features of the genetic representations coincide. First, this is the capability to find compact representations that allow to encode large phenotypes with small genotypes solving the error threshold dilemma. Second, this is the ability for complex adaptation, i.e., to induce highly structured search distributions that incorporate large-scale correlations between phenotypic traits. For example, the variability of one leave is, in certain representations, not independent of the variability of another leave. A GA with direct encoding would have to optimize each single phenotypic element by itself, step by step. The advantage of correlated exploration is that many phenotypic elements can be adapted simultaneously in dependence of each other.

Our experiments demonstrated the theory of σ-evolution which mainly states that the evolution of genetic representations is guided by a fundamental principle: they evolve such that the match between the evolutionary search distribution and the distribution of good solutions becomes better. *The way genetic systems are organized is a mirror of what evolution has learned about the problem.*

References

1. L. Altenberg. Genome growth and the evolution of the genotype-phenotype map. In W. Banzhaf and F. H. Eeckman, editors, *Evolution and Biocomputation: Computational Models of Evolution*, pages 205–259. Springer, Berlin, 1995.
2. T. Bäck. *Evolutionary Algorithms in Theory and Practice.* Oxford University Press, 1996.
3. F. Gruau. Automatic definition of modular neural networks. *Adaptive Behaviour*, 3:151–183, 1995.
4. T. F. Hansen and G. P. Wagner. Epistasis and the mutation load: A measurement-theoretical approach. *Genetics*, 158:477–485, 2001.
5. T. F. Hansen and G. P. Wagner. Modeling genetic architecture: A multilinear model of gene interaction. *Theoretical Population Biology*, 59:61–86, 2001.
6. G. S. Hornby and J. B. Pollack. The advantages of generative grammatical encodings for physical design. In *Proceedings of the 2001 Congress on Evolutionary Computation (CEC 2001)*, pages 600–607. IEEE Press, 2001.
7. G. S. Hornby and J. B. Pollack. Evolving L-systems to generate virtual creatures. *Computers and Graphics*, 25:1041–1048, 2001.
8. H. Kitano. Designing neural networks using genetic algorithms with graph generation systems. *Complex Systems*, 4:461–476, 1990.
9. S. Lucas. Growing adaptive neural networks with graph grammars. In *Proc. of European Symp. on Artificial Neural Netw. (ESANN 1995)*, pages 235–240, 1995.
10. M. Pelikan, D. E. Goldberg, and F. Lobo. A survey of optimization by building and using probabilistic models. Technical Report IlliGAL-99018, Illinois Genetic Algorithms Laboratory, 1999.
11. P. Prusinkiewicz and J. Hanan. *Lindenmayer Systems, Fractals, and Plants*, volume 79 of *Lecture Notes in Biomathematics.* Springer, New York, 1989.
12. P. Prusinkiewicz and A. Lindenmayer. *The Algorithmic Beauty of Plants.* Springer, New York, 1990.
13. R. Riedl. A systems-analytical approach to macro-evolutionary phenomena. *Quarterly Review of Biology*, 52:351–370, 1977.
14. M. Toussaint. On the evolution of phenotypic exploration distributions. In C. Cotta, K. De Jong, R. Poli, and J. Rowe, editors, *Foundations of Genetic Algorithms 7 (FOGA VII).* Morgan Kaufmann, 2003. In press.
15. G. P. Wagner and L. Altenberg. Complex adaptations and the evolution of evolvability. *Evolution*, 50:967–976, 1996.
16. G. P. Wagner, G. Booth, and H. Bagheri-Chaichian. A population genetic theory of canalization. *Evolution*, 51:329–347, 1997.
17. G. P. Wagner, M. D. Laubichler, and H. Bagheri-Chaichian. Genetic measurement theory of epistatic effects. *Genetica*, 102/103:569–580, 1998.
18. R. Watson and J. Pollack. A computational model of symbiotic composition in evolutionary transitions. *Biosystems, Special Issue on Evolvability*, 2002.