On Hopeful Monsters, Neutral Networks and Junk Code in Evolving L-Systems

Stefan Bornhofen
LIAP5, Université Paris Descartes
45, rue des Saints-Pères
75006, Paris, France
sbornhof@math-info.univ-paris5.fr

Claude Lattaud
LIAP5, Université Paris Descartes
45, rue des Saints-Pères
75006, Paris, France
claude.lattaud@math-info.univ-paris5.fr

ABSTRACT

This paper investigates L-system evolution through experiments with a simulation platform of virtual plants. The conducted simulations vary the occurrence probability of terminal versus non-terminal symbols and study its impact on the evolutionary performance of the system. The results reveal a variant of the exploration-exploitation dilemma. A closer look at individual runs allows to discover a range of emergent evolutionary dynamics. In particular, the activation and improvement of previously dormant production rules leads to variation in the fixation rate of mutations. The corresponding fitness leaps suggest that L-system evolution derives much of its creative power from the mobilization of randomly drifting non-addressed rules. The observed patterns are related to the phenomena of positive and negative selection, neutral mutations and junk DNA in the natural genome.

Keywords

virtual plants, L-systems, artificial evolution, artificial life

1. INTRODUCTION

In the research field of Artificial Life, L-systems are considered to be an appropriate formalism to describe many growth processes in organisms, in particular if they feature self-similar and hierarchical structures [30]. A number of studies have explored the evolution of L-systems, notably in the context of plant development. They primarily investigated their adaptability toward predefined target structures [3], addressed the potentials and limits of interactive user-guided evolution [4, 26], or compared some emerging morphological traits of virtual plants to those of natural plants [5, 11].

In such previous works, L-system evolution is typically applied with major attention turned to the resulting phenotypes at the end of the simulations. However, performances in evolutionary computation also depend on the capability to explain, predict and thus optimize the behavior of the system. Therefore, it may be worthwhile to consider the activities at genotypic level during simulation in order to acquire a better understanding of its dynamics, and to increase its performance with a view to accelerating the evolutionary process. A relevant study on this subject was performed by Toussaint [37] who explored the adaptation of genetic representations in a model of L-system based artificial plants. He explicitly defined "2nd-type mutations" that reorganize the generative encodings by the application, creation or deletion of developmental rules in the genotypes without affecting the resulting phenotypes, and showed how evolutionary systems tend to adapt the phenotypic variability such that the search strategy becomes more efficient.

The present paper takes another step into this direction by a deeper investigation on the role of the occurrence probability of terminal versus non-terminal symbols by mutation. Several experiments address the impact of this parameter on the overall performance of the system, as well as on the emerging dynamics within single evolutionary runs. In contrast to [37], the genetic representation of the plants remains fixed in this paper, and neutral networks arise from mutations in currently inactive parts of the genetic code. The results not only emphasize the importance of neutral networks in L-system evolution, but also allow to relate the observed patterns to concepts of biological evolutionary theory.

The next section outlines the L-system formalism and gives an introduction to the state of the art in L-system evolution. After the presentation of the used plant model in section three, several experiments are described in section four. The discussion in section five compares the obtained results to evolutionary dynamics of the natural genome. Section six closes the paper with conclusions and perspectives on the approach.

2. L-SYSTEMS

Designed in 1968 by Lindenmayer [25], L-systems are based on formal grammars with recursive applications of production rules in a parallel rewriting process. The possibility of simultaneous productions reflects the biological motivation of L-systems, intended to capture cell division in multicellular organisms. Two major applications in computer graphics are the generation of fractals [34] and the modeling of plants [30].

A basic L-system can be described by the triplet (A, P, ω)

GECCO'08, July 12–16, 2008, Atlanta, Georgia, USA.

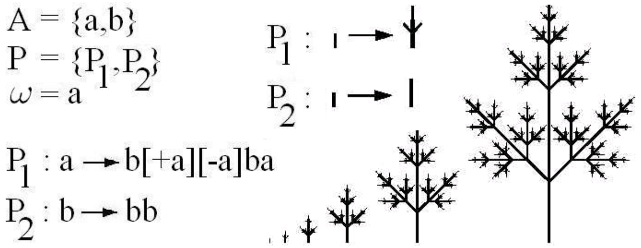

Figure 1: Iterations of the L-system (A, P, ω)

where A is an alphabet, P a set of production rules and ω an axiom denoting the initial word. Starting with the axiom, the production rules are applied to the characters in order to form a string. The string represents the entire modeled structure, whereas each character represents an elementary unit like, in the case of plants, a leaf or a bud. Positional information of the units can be integrated by using a bracketed notation for ramifications and special characters for rotations in space. The translation of a string into a graphical structure is achieved by turtle geometry [30] where a cursor, called "turtle", starting at a specified location and orientation in space, interprets the characters as a series of draw commands as well as position- and orientation-changing instructions. Figure 1 illustrates a 2D example and its resulting morphology after several iterations. The special symbols $+$ and $-$ denote fixed angle rotations of the turtle.

Although L-systems were not initially conceived to be evolved [35], their interest as a generative genetic encoding has been recognized by a number of authors. Jacob [22] formalized the evolution of context-free and context-sensitive L-systems and introduced the "Genetic L-systems Programming" paradigm as a general framework for evolutionary dynamics of parallel rewriting systems. Several performance comparisons by Hornby and Pollack [20, 19] indicate that L-systems used as encoding for automated design problems can produce better results faster than non-generative encodings. Ochoa [27] evolved 2D virtual plants and showed that L-systems are an adequate genetic representation for studies which simulate natural morphological evolution. For these reasons, L-system encodings have in many cases been chosen for the construction of models which best describe given natural target structures such as plants [3], but also the blood vessels of the eye [24] or proteins [12].

A number of applications took advantage of the flexibility of evolutionary L-systems, capable of generating an impressive variety of morphologies from a small number of underlying rules. L-system based virtual plants were fashioned by interactive evolutionary sessions where the human observer chooses the most interesting-looking individuals for further reproduction [4, 26]. Likewise, several applications such as the Nerve Garden [7] appeared on the Internet, allowing users to evolve and interact with plant communities in virtual online worlds.

Recent evolutionary models also consider interactions with the environment which allow to better compare the simulations to the development of natural plants. Ebner [11] evaluated L-system based virtual plants for their amount of captured light and showed that competition leads to an arms race such that plants grow high compared to small bushy plants which develop when evolved independently. These simulations confirmed that the L-system approach is appropriate to study evolutionary dynamics in natural plant life.

3. THE PLANT MODEL

As seen in the previous section, current evolutionary L-system based models primarily focus on applications in the context of plant growth. However they incorporate no physiological processes, and their environment possesses no or only minimal dynamics. The following section describes the major characteristics of a complementary plant model which is intended to fill this gap. Besides studies on L-system dynamics, the model is designed to understand the evolution of plant development with respect to environmental constraints such as resource deficiency or competition. Due to its morphological and physiological component, it approaches the class of functional-structural plant models [33] conceived by the scientific community of biologists.

3.1 Environment

The physical environment is a continuous 3D space composed of the soil and the sky, homogeneously divided into a number of voxels providing the two vital resources light and minerals. Other significant resources such as water and CO_2 are currently not modeled, assuming that their supply is constant and sufficient.

The sky represents the aerial part of the environment. Light, captured by the leaves in order to produce carbon via photosynthesis, is modeled as a number of light sources parameterized by a variable representing their initial irradiation and a vector indicating their direction. The total irradiation of a voxel results from the sum of all light sources. Each object of the simulation is located in a certain voxel. Since its geometry may not entirely be enclosed in one voxel, the central point is decisive. If it is situated in the aerial part, the object casts shadows. In such case, the luminosity of all voxels following the directions of the light sources are decreased. In order to avoid time-consuming computation such as geometrical calculations or the use of computer graphics in order to determine how much light a plant receives with respect to each light source, the shading factor of an object

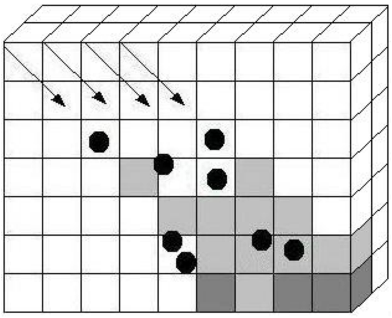

Figure 2: Shadow casting in sky voxels

Table 1: L-system alphabet of the used plant model

Character	Compartment	Geometry	Function
l	shoot	sphyl	captures virtual light to photosynthesize carbon
f	shoot	sphere	represents a flower
b	shoot	sphyl	creates a branching structure
r	root	sphyl	assimilates nutrients in the soil
c	root	sphyl	creates a branching structure
A,..Z	shoot/root	none	represents predecessors of the production rules
[,]	shoot/root	none	indicates a ramification
+-<>$&	shoot, root	none	represents 3D rotations

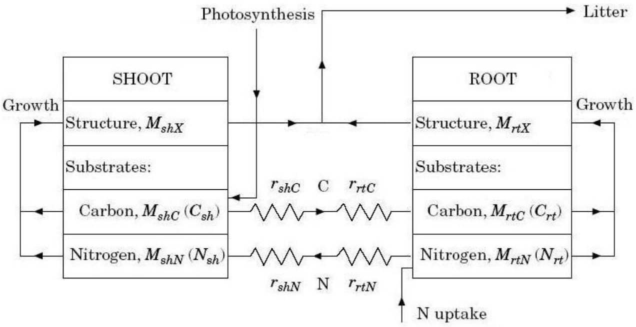

Figure 3: The transport-resistance model

does not depend on the exposed surface of the object but on its volume (figure 2).

Soil voxels contain minerals which are assimilated by the fine roots. Diffusion, a passive movement from regions of high concentration to regions of low concentration, leads to mineral balance between adjacent voxels. The flow between a soil voxel and its neighbors is determined by Fick's first law of diffusion [13]. The soil model respects a simplified mineral cycle. All the assimilated nutrients of a virtual plant are eventually redeposited in the soil so that their total amount within the environment is constant. The nutrients of dead roots are put in the corresponding voxels and those of the aerial compartment in a mold layer which gradually penetrates the upmost soil layer.

3.2 Plant phenotype

A virtual plant is divided into a shoot and root component. The morphologies are expressed by two L-systems, whose alphabet is detailed in table 1. The model offers the possibility of adding a stochastic component, associating each production rule with a triggering probability, but in the scope of this paper only deterministic context free L-systems, also called D0L-systems [30] are applied. This choice was made to disengage the evolutionary dynamics from contingencies at individual level. The physiological processes of a plant are based on a two-substrate version of the transport-resistance model [36], where an aboveground and a belowground compartment assimilate and exchange the resources carbon and minerals, and convert them into biomass (figure 3).

In the presented plant model, plant structure and function are coupled in that, contrary to the original transport-resistance model, the compartmental equations for resource assimilation are broken down into the uptake of each individual captor module. The leaves and fine roots provide carbon and minerals that depend on their local access to the resources and store them in the respective substrate pools. Likewise, new biomass is not stored in a real-valued aggregate variable, but evenly distributed to the apexes of the current plant morphology. The L-system is thus parametric in that every character representing a plant module holds variables that contain additional information such as the current biomass.

Seeds start with the axiom $[A]$ for each compartment and hold a small amount of initial biomass. An L-system rule is applied once the biomass of an apex reaches the required cost for the production of the successor string. This value is calculated from the genetically defined costs of all plant modules that will be produced. In addition to the new modules, growing apices have to pay for the thickening of the branching modules below them. This stipulation guarantees that the cost for apex growth increases with the distance from the ground and is based on the pipe model theory [32] which states that any cross sectional area in a branching system, whether shoot or root, is proportional to the biomass of the captors, leaves or fine roots, that it serves.

3.3 Plant genotype

A plant genotype contains the parameters and production rules of the two L-systems, the parameters involved in the transport-resistance model, as well as a number of physiological parameters concerning resource management, growth and reproduction. Figure 4 displays a portion of this genotype for a simple bush. Depending on the object of the study, some of its elements may be fixed and others subjected to evolution. To single out the basic dynamics of L-system evolution in the scope of this paper, only the shoot system is allowed to mutate. All physiological parameters and the root system are predefined and fixed. Moreover, although a range of genetic operators on the L-systems were introduced in previous papers [4, 5], a minimal set of recombinations and mutations are applied here.

The number of rules in the shoot L-system has been fixed to 10. This value is a compromise between maintaining genotype complexity and dealing with a manageable amount of evolving elements. The non-terminal symbols are thus represented by the uppercase characters $A, ...J$. Three genetic operators are defined. They are chosen such that any set of rules can be constructed by evolution. The recombina-

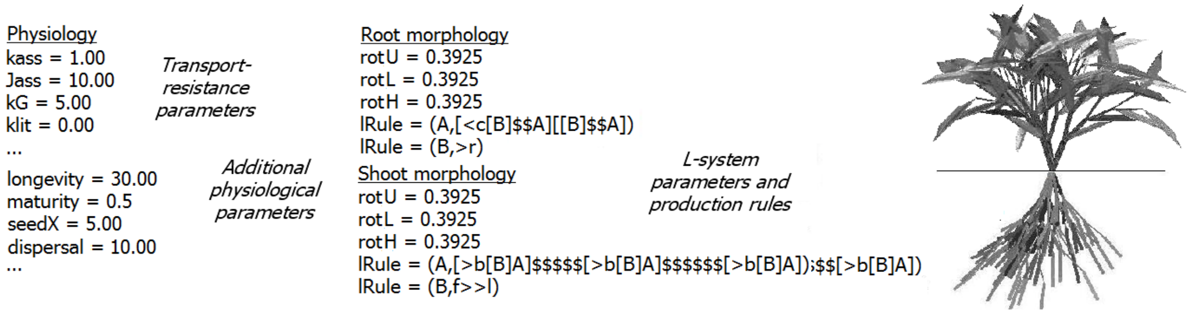

Figure 4: Sample genotype of a bush

tion operator between two genotypes randomly selects one of the two corresponding alternatives for each production rule and merges them into one L-system. Two genetic operators, insertion and deletion, act on the characters of the successor strings. Both are applied with a fixed probability p to each production rule. The deletion operator deletes one terminal or non-terminal character at random. Resulting empty brackets are removed. The insertion operator chooses a random location within the successor string, and the added symbol depends on a given occurrence probability r of terminals versus non-terminals. There is a chance of $r/2$ for a plant module (b,l,f), $r/2$ for a rotation command and $1 - r$ for an apex, i.e. a bracketed non-terminal, to be inserted. As an example, figure 5 illustrates recombination and subsequent mutation for a part of the genotype.

4. EXPERIMENTS

The model of section three has been implemented as a simulation platform. It is developed in C++ and uses the OGRE library [2] for graphical representations and the Open Dynamics Engine [1] for collision detection.

4.1 Setup

The plants are evolved by a typical evolutionary algorithm [18]. To grant evolution as much freedom as possible, a run starts with a genotype population of minimal L-systems, i.e. holding the shoot production rules $A \rightarrow lf$ and $B, ..., J \rightarrow \epsilon$. The two terminals of the first rule are predefined to accelerate the evolutionary take-off. In the phase of development, the genotypes are translated into a population of phenotypes. To do so, a seed of each genotype is placed in a sufficiently large environment featuring vertical light and grown for a fixed amount of time. Subsequently, the developed phenotype is evaluated. The fitness function is defined as

$$F = L_f * L_l$$

where L_f is the sum of light available to the flowers and L_l the sum of light available to the leaves. This is meant to express one of the most important aspects in plant life, that is the exposure of reproductive and resource capturing organs. Other fitness functions and their impact on plant development are discussed in [5, 6].

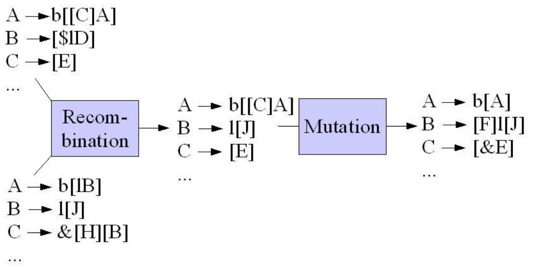

Figure 5: Recombination and mutation operators

The tournament selection applied here is inspired by competition in nature and arranges "tournaments" to compare the fitness between a few randomly chosen individuals [15]. The best performing individual of every tournament is retained. This approach additionally offers the advantage to easily adjust selection pressure by changing the tournament size. The best individuals survive and give birth to the next generation of mutated genotypes. Each succeeding population is thus composed of the former elite as well as newly created individuals. One half of the offspring descend asexually from a single parent by mutation, and the other half from two parents by recombination and subsequent mutation in order to maintain diversity within the population. Natural plant life likewise features vegetative and sexual reproduction. While not being addressed within the scope of the present paper, the impact of recombination on L-system evolution is a subject of future studies.

The probability of each genetic operator is defined as $p = 0.1$. This value may be overrated compared to natural evolution, but has been chosen to accelerate the process. The populations are composed of 40 plants grown for 30 time units. Survival ratio is set to 1/4 which means an elite of 10 individuals per generation, and tournament size to 10 individuals. This configuration has already proven to produce conclusive results in a reasonable amount of time [5]. A run over 500 generations would take about one hour on a 2 GHz PC.

4.2 Results

The constraints imposed by the fitness function lead to natural looking plant architectures exposing both flowers and leaves to the light. Figure 6 shows three examples of the evolved morphologies ($r = 0.5$). The approach could be a useful algorithm for the automated design of virtual plants. More detailed studies on the evolution of phenotypic traits can be found in [6].

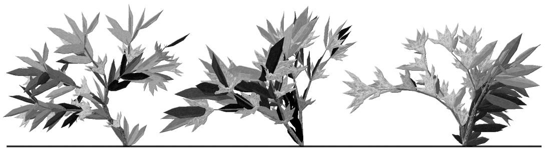

Figure 6: Evolved plant morphologies

In order to assess the impact of the occurrence probability r on the evolutionary performance, fourteen series of ten runs have been conducted over a range of different values. The number of simulations for each value was limited for reasons of time and computational resources, but turned out to be sufficient to identify several important tendencies. Figure 7a plots the averaged fitness increase of some series over 500 generations (ten out of fourteen for reasons of readability), and figure 7b resumes for each series the correlation between r and the final fitness. Extreme probabilities of terminals or non-terminals reduce the performance of the system, however high rates of terminals yield better results than equivalent rates of non-terminals. Moreover, curves for high r tend to show a late and steep evolutionary take-off compared to those with low r which exhibit a steadier fitness increase. The detected optimum among the chosen range of values, located at 0.5, is uncertain due to the limited number of runs. However, the tested values between 0.5 and 0.8 lead to similar performance, suggesting that evolution is robust to the investigated parameter.

To understand the observed patterns, a closer look at individual runs has been taken with a focus on changes in the evolutionary dynamics between low and high values of r. The upper part of figure 8 plots the fitness values of the elite members during a typical simulation with $r = 0.9$. The graph shows plateaus where evolution is slow or even stagnating, interrupted by phases during which fitness increases rapidly. Such leaps occur irregularly, and their appearance varies in extent, i.e. duration and slope. The graph additionally indicates the average number of rules involved into the plant growth, as well as the number of non-terminal characters in the successor strings of these rules. The data illustrates that only a part of the ten rules is actually applied, whereas others are not accessed because no non-terminal characters refer to them. Such inactive rules are portions of the genetic encoding that do not play any role in determining the phenotype. Evolutionary enhancements during a plateau phase concern mutations of terminal characters. In contrast, fitness leaps are typically initiated by fundamental changes in the network of active production rules, i.e. by the insertion or, less frequently, deletion of a non-terminal. The higher r, the less often structural changes are explored. Such event profoundly affects the integrity of the genotype and mostly turns out to decrease fitness, so that the individual is eliminated by selection. However in some cases, the mutation provides a selective advantage that leads to its propagation through the entire population, accompanied by a series of improving mutations of terminal symbols.

Figure 9 presents the same kind of graph for an evolutionary run with $r = 0.1$. The dynamics are different from those observed in the previous simulation. Due to the abundance of appearing non-terminal characters, the L-system adopts a major part of the ten existing rules within the first fifty generations. Multiple adoptions may occur via one mutation which implicates other rules by cascade. In many runs, the high number of active rules is maintained until the end of the simulation. However, in some cases, inefficient rules are rejected, such as observed two times in the presented run for the rule I. Moreover, fitness leaps tend to be less distinct. As r is low, there is only few potential for structural changes to be followed by improving terminal symbols.

The observations reveal a central trade-off for the evolutionary performance of L-systems. The mutations of terminal symbols allow to gradually enhance a given set of active production rules in a robust way. The mutations of non-terminals are less robust but necessary to explore new favorable combinations between the rules. As both dynamics are required to optimize the performance of the system, the trade-off can be considered as a variant of the exploration-exploitation dilemma [15].

The lower graph of the figures 8 and 9 plots the mutation history of a sample plant at generation 500, i.e. the record of all mutations that led from the initial ancestor to the final plant. The activity periods of the production rules are tinted gray, and insertions and deletions are shown by filled and empty diamonds respectively. Inactive rules are not subjected to selection pressure, and occurring mutations are therefore neutral. On average, they get fixed in the population with a rate of 0.18 in both runs. This value is theoretically equal to the mutation rate [23]. On the other hand, the fixation rates in active rules feature non-random dynamics. Once a rule has been mobilized, the arising selective pressure typically introduces further enhancements within only a few generations, which results in the observed fitness leaps. A structural change can also entail some mutations in already established rules, as it can be observed in the figure 8 for the rules A and I. During these temporary phases of positive selection, the fixation rate in active rules may exceed the mutation rate. However, due to functional constraints, these rules subsequently undergo negative selection where most mutations turn out to be deleterious and get purged. By averaging over significant periods where the fitness is stagnating, it was found that their fixation rate attains a lower limit of 0.009. The strictly positive value means that neutral mutations appear in active rules. All observed cases concerned rotation commands without effect on the growing phenotype. They were typically situated in front of a closing bracket, i.e. at the end of a ramification, so that the spatial position of all plant organs is unaffected by their presence or absence.

5. DISCUSSION

According to evolutionary biology, the natural genome holds an amount of information which has no known biological function, often referred to as “junk DNA” [29]. Several theories for their emergence were proposed such as selfish replication [8] or duplication and subsequent divergence of existing genes [28]. Non-addressed L-system production rules can be related to junk DNA. They likewise represent non-functional genetic code, are not subjected to selection pressure and drift through a series of random mutations. Just as in the simulations, junk DNA evolves faster than active

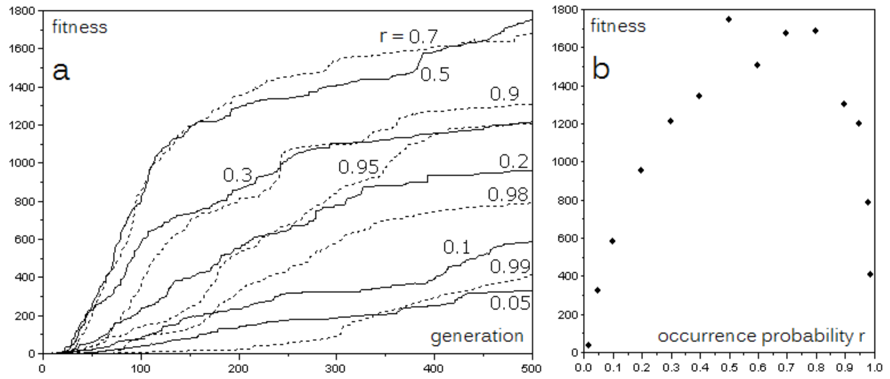

Figure 7: (a) Evolutionary performance depending on r (b) Correlation between r and final fitness

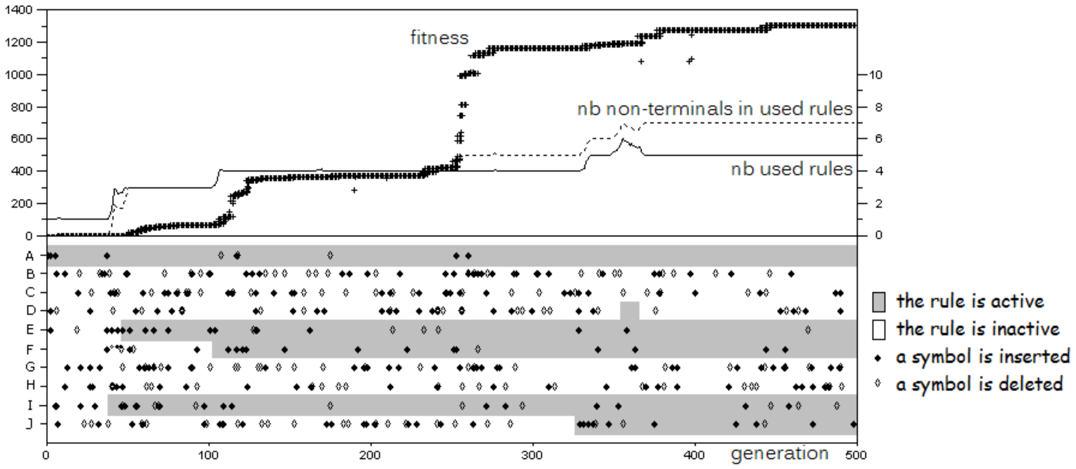

Figure 8: Evolutionary run ($r = 0.9$) and mutation history of a sample genotype at generation 500

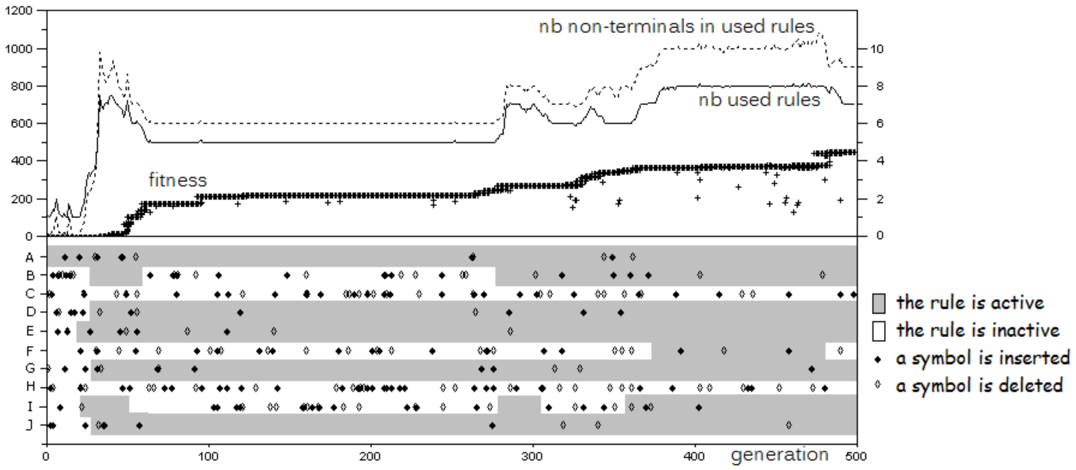

Figure 9: Evolutionary run ($r = 0.1$) and mutation history of a sample genotype at generation 500

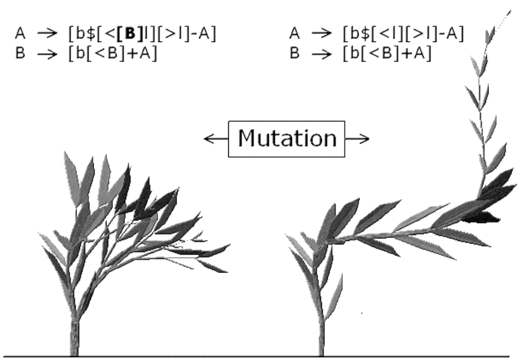

Figure 10: Rule activation or deactivation

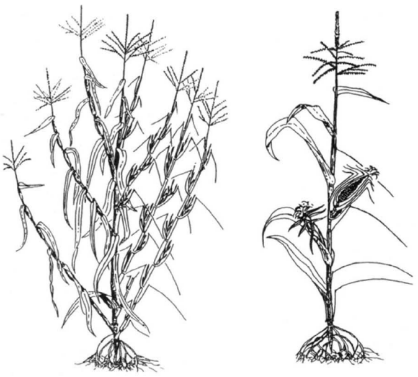

Figure 11: The domestication of the maize

genome sequences, according to the neutral theory introduced by Kimura [23].

Ohno [28] originally suggested that portions of junk DNA would sporadically reemerge from non-functionality, with a new function acquired as a result of favorable mutations. Although recent research indicates that the revival of gene sequences is not as simple as initially thought [39], their preservation in the genome may at least serve as a source for new genes by the recombination of fragments [21]. For L-systems, gene activation corresponds to the appearance of a non-terminal character which creates a reference to an inactive rule.

As seen in figure 10, rule activation or deactivation in the virtual plant model can have a drastic effect on the phenotype. Each production rule can thus be related to the role of regulatory genes in natural plants. As an example, figure 11 illustrates the morphological difference between cultivated maize (Zea mays spp. mays) and its wild progenitor teosinte (Zea mays ssp. parviglumis). The domestication of crop plants has often involved a concentration of resources in the main stem leading to straight growth. In the case of teosinte, a mutation in the regulatory gene TB1 seems to be responsible for this evolutionary change [10].

“Hopeful monster” is a term coined by Goldschmidt to describe an organism subjected to important phenotypic variation [16]. The appearance and role of such discontinuous phenotypic evolution in nature remain largely controversial [9], but in the presented model of virtual plants, hopeful monsters are common. As seen in the last section, they may initiate a series of mutations which lifts the entire population to a new fitness level. This phenomenon matches Gould’s reflection, who suggested that a hopeful monster, if ever occurring in nature, may enter an adaptive phase during which it undergoes rapid evolutionary changes. Large phenotypic variations need not produce perfected forms, but rather serve as “a key adaptation to shift its possessor toward a new mode of life. Continued success in this new mode may require a large set of collateral alterations, morphological and behavioral; these may arise by a more traditional, gradual route once the key adaptation forces a profound shift in selective pressures.” [17].

6. CONCLUSIONS

The evolution of L-systems has been the object of several scientific studies, however only a few previous works gave major attention to its dynamics during the simulations. This paper presented evolutionary experiments with a simulation platform of L-system based virtual plants. The conducted runs varied the occurrence probability of terminal versus non-terminal symbols by mutation and studied its impact on the evolutionary performance of the system. The results revealed a variant of the exploration-exploitation dilemma. High probabilities of non-terminals implicate many production rules in the plant growth, but exhibit few evolutionary potential. High probabilities of terminals lead to the successive activation and improvement of previously dormant rules. The resulting fitness leaps suggest that L-system evolution derives much of its creative power from the mobilization of randomly drifting non-addressed rules. Just as in natural evolution, “neutral networks are key to change. (...) Populations may be pinned at the phenotypic level, but they constantly change at the genetic level, drifting on neutral networks, thereby dramatically increasing their chances for phenotypic innovation” [14]. The conclusion also fits into the largely accepted view of a positive relationship between neutrality and performance in evolutionary computation [31, 38].

A closer look at individual runs allowed to discover variation in the fixation rate of mutations and to relate these patterns to positive, neutral and negative selection of the production rules. The analogy to natural junk DNA allows to better understand the observed dynamics. More evidence and measures may be needed to draw accurate quantitative conclusions, however the conducted experiments are sufficient to point out a number of significant patterns in L-system evolution.

The presented paper explored some basic dynamics of evolving L-systems. However, many questions remain unsettled such as: How can the observed fitness leaps be quantified? How does this measure change in function of the parameter r? Will r find a proper balance if it is not fixed but allowed to evolve as a part of the genotype? More generally, which mutation and recombination operators are efficient? What are the benefits and drawbacks of an evolving number of rules? Is it possible to enhance the performance by the introduction of adaptive mutation rates, where "building blocks" such as sub-trees mutate with a variable probability, just as observed in different sites of the natural genome [23]? These are issues to be addressed in future studies, and their answers may constitute further steps in the understanding and optimization of L-system evolution.

7. REFERENCES

[1] *ODE website: http://www.ode.org (April, 2008).*

[2] *OGRE website: http://www.ogre3d.org (April, 2008).*

[3] R. Bian, J. Hanan, and N. Chiba. Statistical data directed evolution of l-system models for botanical trees. In *Proceedings of FSPM04, Montpellier, France*, pages 253–256, 2004.

[4] S. Bornhofen and C. Lattaud. Evolutionary design of virtual plants. In *Proceedings of CGVR, Las Vegas, USA*, pages 28–34, 2006.

[5] S. Bornhofen and C. Lattaud. Life history evolution of virtual plants : Trading off between growth and reproduction. In *Proceedings of PPSN IX, Reykjavik, Iceland*, pages 808–817, 2006.

[6] S. Bornhofen and C. Lattaud. Evolution of virtual plants interacting with their environment. In *Proceedings of VRIC, Laval, France*, pages 172–176, 2007.

[7] B. Damer, K. Marcelo, and F. Revi. Nerve garden: A public terrarium in cyberspace. *In Heudin, J.C. (editor), Virtual Worlds, Springer-Verlag*, pages 177–185, 1998.

[8] R. Dawkins. *The Selfish Gene.* Oxford University Press, Oxford, 1976.

[9] R. Dawkins. *The Blind Watchmaker.* WW Norton, New York, 1986.

[10] J. Doebley, A. Stec, and C. Gustus. Teosinte branched1 and the origin of maize: evidence for epistasis and the evolution of dominance. *Genetics*, 141:333–346, 1995.

[11] M. Ebner, A. Grigore, A. Heffner, and J. Albert. Coevolution produces an arms race among virtual plants. In *Proceedings of EuroGP, Kinsale, Ireland*, pages 316–325, 2002.

[12] G. Escuela, G. Ochoa, and N. Krasnogor. Evolving l-systems to capture protein structure native conformations. *LNCS*, 3447:73–83, 2005.

[13] A. Fick. Über diffusion. *Ann. Phys. (Leipzig)*, 170:59–86, 1855.

[14] W. Fontana. The topology of the possible. In A. Wimmer and R. Kössler, editors, *Understanding Change: Models, Methodologies and Metaphors.* Palgrave Macmillan, Basingstoke, 2006.

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

[16] R. Goldschmidt. *The Material Basis of Evolution.* Yale University Press, 1940.

[17] S. J. Gould. The return of hopeful monsters. *Natural History*, 86(6):23–30, 1977.

[18] J. Holland. *Adaptation in Natural and Artificial Systems.* Univ. of Michigan Press, Ann Arbor, 1975.

[19] G. S. Hornby and J. B. Pollack. The advantages of generative grammatical encodings for physical design. In *Proceedings of CEC, Seoul, Korea*, pages 600–607, 2001.

[20] G. S. Hornby and J. B. Pollack. Body-brain co-evolution using l-systems as a generative encoding. In *Proceedings of GECCO, San Francisco, USA*, pages 868–875, 2001.

[21] S. G. Inge-Vechtomov. A possible role of translation ambiguity in gene evolution. *Molecular Biology*, 36(2):202–208, 2002.

[22] C. Jacob. Genetic l-system programming. In *Proceedings of PPSN III, Jerusalem, Israel*, pages 334–343, 1994.

[23] M. Kimura. *The Neutral Theory of Molecular Evolution.* Cambridge Univ. Press, Cambridge, 1983.

[24] G. Kokai, Z. Toth, and R. Vanyi. Modelling blood vessels of the eye with parametric l-systems using evolutionary algorithms. *LNCS*, 1620:433–442, 1999.

[25] A. Lindenmayer. Mathematical models for cellular interaction in development. *I and II. Journal of Theoretical Biology*, 18:280–315, 1968.

[26] K. J. Mock. Wildwood: The evolution of l-system plants for virtual environments. In *Proceedings of ICEC, Anchorage, Alaska*, pages 476–480, 1998.

[27] G. Ochoa. On genetic algorithms and lindenmayer systems. In *Proceedings of PPSN V, Amsterdam, The Netherlands*, pages 335–344, 1998.

[28] S. Ohno. *Evolution by Gene Duplication.* Springer-Verlag, Berlin-Heidelberg-New-York, 1970.

[29] S. Ohno. So much "junk" dna in our genome. *Brookhaven Symposia in Biology*, 23:366–370, 1972.

[30] P. Prusinkiewicz and A. Lindenmayer. *The Algorithmic Beauty of Plants.* Springer-Verlag, Berlin, 1990.

[31] F. Rothlauf and D. E. Goldberg. Redundant representations in evolutionary computation. *Evolutionary Computation*, 11(4):381–415, 2003.

[32] K. Shinozaki, K. Yoda, K. Hozumi, and T. Kiro. A quantitative analysis of plant form - the pipe model theory, i. basic analysis. *Jpn. J. Ecol.*, 14:97–105, 1964.

[33] R. Sievanen, E. Nikinmaa, P. Nygren, H. Ozier-Lafontaine, J. Perttunen, and H. Hakula. Components of functional-structural tree models. *Annals of Forest Sciences*, 57:399–412, 2000.

[34] A. Smith. Plants, fractals, and formal languages. *Computer Graphics*, 18(3):1–10, 1984.

[35] K. Stanley and R. Miikkulainen. A taxonomy for artificial embryogeny. *Artificial Life*, 9(2):93–130, 2003.

[36] J. H. M. Thornley. A balanced quantitative model for root:shoot ratios in vegetative plants. *Annals of Botany*, 36:431–441, 1972.

[37] M. Toussaint. Demonstrating the evolution of complex genetic representations: An evolution of artificial plants. In *Proceedings of GECCO, Chicago, USA*, pages 86–97, 2003.

[38] T. Yu and J. Miller. Neutrality and the evolvability of boolean function landscape. In *Proceedings of EuroGP, Lake Como, Italy*, pages 204–217, 2001.

[39] J. Zhang. Evolution by gene duplication: an update. *Trends in Ecology and Evolution*, 18(6):292–298, 2003.