Creative 3D Designs Using Interactive Genetic Algorithm with Structured Directed Graph[*]

Hyeun-Jeong Min and Sung-Bae Cho

Dept. of Computer Science, Yonsei University
134 Shinchon-dong, Sudaemoon-ku, Seoul 120-749, Korea
`{solusea,sbcho}@cs.yonsei.ac.kr`

Abstract. We propose a methodology for representing artificial creatures like 3D flowers. Directed graph and Lindenmayer system (L-system) are commonly involved in AI-based creativity research for encoding ceatures. It is difficult for L-systems to directly feed back real morphologies structurally from their genotypes, since they are a grammatical rewriting system and also use parameters such as loops, procedure calls, variables, and primitive parameters for representing their genotypes. In this paper flower genotypes are manifested by a knowledge-based structured directed graph (SDG) and phenotypes are represented a flower morphology resulting from the derivation and graphical representation of the genotypes. Evolution is simulated using an interactive genetic algorithm (IGA), where a SDG is useful for genotypic representation of creatures and IGA uses human evaluation for the fitness function. We have applied the creation of 3D flowers using the knowledge-based SDG and IGA. Experimental results show that realistic flower morphologies can be created by the proposed method.

1 Introduction

Art, music, and designs have been emerging from computers for many years in artificial life (A-life), which is the study of life and life-like processes having autonomy, adaptation, self-replication and self-repairing [1, 2]. It can automatically create satisfactory artificial characters having actions or morphologies and also generate new population. Besides a binary encoding, directed graph (DG) or Lindenmayer system (L-system) is used to represent genotypes for creating individuals using evolutionary algorithm. This paper provides a glimpse of the creativity in specific domains when DG is used in combination with interactive genetic algorithm (IGA) techniques. We specifically focus on artificial flowers, demonstrating how realistic-looking creatures reflecting human's preference can be modeled with genetically generated DG.

In DG proposed by K. Sims for creating 3D creatures [3], each of creatures is represented by composition of nodes and edges. It is convenient to define a morphological structure of individuals more quickly and easily and users can intuitively figure

---

[*] This work was supported by Korea Science and Engineering Foundation (KOSEF) through the Biometrics Engineering Research Center (BERC) at Yonsei University.

C. Zhang, H.W. Guesgen, W.K. Yeap (Eds.): PRICAI 2004, LNAI 3157, pp. 391–400, 2004.

out the real morphology of a genotype represented by DG. K. Sims generated various creatures and locomotion in his previous researches such as swimming, walking, jumping, and following actions of creatures having various morphologies. It takes however a long time for generating these creatures in his method because there are many cases for their genotypes.

The approach is based on the use of an IGA and a knowledge-based SDG that generates a simplification of flower morphology similar to real flowers in nature. The fitness functions must provide an evaluation score for every solution. For creative products in computers especially like design and art, a common problem in applying evolutionary computation techniques to artistic domains is the difficulty of deriving formal fitness functions to evaluate the individuals [4]. IGA is a technique which involves developing the automatic design methods for the systems that are based on user's preference and subjective evaluation. We apply the knowledge-based genotypic representation and the evaluation of phenotypes by IGA that is useful in generating natural morphology rapidly since it has no evaluation function and is evaluated by human beings, and we have to deal with few individuals in limited generations.

Here we consider a kind of SDG as genotypic representations and evolve translated phenotypes by interactive genetic algorithm (IGA) for automatically creating natural artificial flowers. The next section explains the related works of automatically generating creatures using evolutionary algorithm and genotypic representation by SDG, and section 3 describes how structured directed graph and IGA proposed in this paper can be used. Finally sections 4 and 5 provide results, discussion, and suggestions for the future work

2 Related Works

L-systems were originally introduced to model the development of simple multicellular organisms in terms of division, growth, and death of individual cells [5, 6]. Ochoa generated artificial 2D plant morphologies by mathematical formalism known as L-systems [7]. The applications of L-systems have subsequently been extended to plants and complex branching structures. The parametric L-systems, which are a particularly convenient programming tool for expressing growing models of plants having symmetric structure and have rewriting processes for reuse of rules and parameters for various morphologies. G. Hornby used L-systems for encoding in evolutionary algorithm to create virtual creatures having hundreds of parts, and presented co-evolving morphology and controller by using oscillator circuits controlling each actuated joint of creature [8]. Moreover, he defined generative representations which identified by their ability to reuse elements of the genotype and compared it against direct representation [9].

DG representation to specify the construction of creatures by K. Sims used nodes for body segments, which are composed of another nested graph for the body segment's neural controller for behaviors such as walking, swimming, jumping and following [10]. DGs were presented as an appropriate basis for a grammar that could be used to describe both the morphology and nervous systems of virtual creatures like L-system. New features and functions that are appropriate for the environment can be generated or existing ones can be removed so that the levels of complexity can also evolve. Moreover he generated virtual creatures that competed in a physically simulated three-dimensional world and had various morphologies by these DGs and nested graphs for their genotype [11]. B. Lintermann and O. Deussen presented a modeling method that allowed easy generation of many branching plants including flowers, bushes, and trees [12]. They defined a set of components describing structural and geometrical elements of plants and users could get immediate feedback on what they had created. K. Sims generated various creatures and their locomotion using DG. On the other hand, we could generate creatures fast by a simplification of representations and structures of real objects.

The GOLEM created by H. Lipson consisted of bars and actuators for structures and artificial neurons for behavior control [13]. J. Ventrella presented an animation system developed for the exploration of emergent morphology and behaviors using genetic algorithm for evolving populations composed of improved and realistic behaviors [14]. His creatures were encoded as fixed-length vectors of parameters for constructing a creature. Various morphologies were described in Framsticks simulator for modeling, simulating and optimizing virtual agents, with three-dimensional bodies and embedded control system [15]. J. Bongard and R. Pfeifer generated growing creatures under a simulated genetic process by defining gene expression rules that determine the division of body segments [16]. H. Kawamura and H. Ohmori proposed stable frame structures using genetic algorithm [17].

We can represent the genotype of creatures using L-system and DG respectively. It is difficult to grasp an entire structure of morphology and users also get no direct feedback from the generated rules because L-system uses parameters and rewriting rules [12]. The L-system is suitable for generating growing models, that is, the growth processes of plants can be captured as well, and it can also generate artificial creatures of symmetric structure. On the other hand, in the knowledge-based SDG, users can directly remind phenotypes because of structurally defined genotypes in DG. In this paper, we utilize SDG for easily representing global aspects of creatures. For this purpose we can generate natural morphology within the real world even though evolution is iterated. Additionally, we have applied IGA using user's emotional evaluation for automatically generating a simplification of real flowers in nature. In this paper, we take a chrysanthemum as an example.

3 Creating 3D Artificial Flowers

From the modeling point of view, the global characteristics of plants have to be directly dealt with. We use an evolutionary algorithm for automatically creating populations. We have to distinctly define the criteria of fitness evaluation for using an evolutionary algorithm, but it is difficult to apply the conventional evolutionary algorithm since humans' emotional evaluation is required for creating a more natural morphology of flowers. Therefore, in this paper we present the domain-specific SDG as the genotypic representation of real morphology and IGA for an automatic creation of natural flower morphology.

3.1 Structured Directed Graph

In DG, individuals are encoded with nodes and edges. Each node in the DG contains information describing rigid objects such as dimension and color of each part, and each edge also contains the information of positions and orientations for connecting edges to each part. Figure 1 shows that the structure of a real flower [18] and the corresponding genotypic representation with DG.

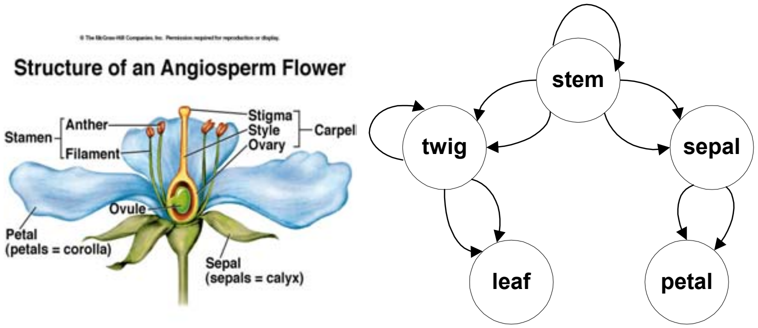

Fig. 1. Real flower morphology and the corresponding representation of genotype.

Directed graph $DG=(V, E)$ has a set of vertices and a set of edges between the vertices. Vertex and edge in directed graph described in Figure 1 are defined as follows.

$$V(DG) = \{v_{stem}, v_{twig}, v_{sepal}, v_{petal}, v_{leaf}\}$$

$$E(DG) = \{<v_{stem}, v_{stem}>, <v_{stem}, v_{twig}>, <v_{stem}, v_{sepal}>, <v_{twig}, v_{twig}>, <v_{twig}, v_{leaf}>, <v_{sepal}, v_{petal}>\}$$

We can represent adjacency matrix of DG with 5×5 matrix because $DG=(V, E)$ and $|V| = 5$, and each element of this matrix is defined as follows.

$$a_{ij} = \begin{cases} n, & \text{if } (v_i, v_j) \in E(DG) \text{ and } i \neq j \\ n/r, & \text{if } (v_i, v_j) \in E(DG) \text{ and } i = j \\ 0, & \text{otherwise} \end{cases}$$

where n and r represent the number of edges and recursion count, respectively.

We have to define parameters in nodes and edges of DG for the detailed morphology. In an evolutionary process, parameter values of each node and edge are initially chosen randomly, but they can be automatically generated through operations. Defining this SDG with genotypic representation of an individual, we can easily define the whole structure of an individual and get direct feedback to a representation of real morphology. Figure 2 shows the two-dimensional matrix representing the relation of each node and edge in the DG and an example of a flower morphology created. In the table n is the number of edges and r is the recursion count of rigid parts. In this matrix (v_i, v_j) denotes the ith column and jth row and each node has n edges and r parts for recursive connection of the same node. Also (v_i, v_j) has n connections with different edges and nodes. For instance, if a stem node has 5 edges with twig node, a stem part can connect 5 twig parts.

Node	stem	twig	sepal	petal	leaf
stem	*n/r*	*n*	*n*	0	*n*
twig	0	*n/r*	*n*	0	*n*
sepal	0	0	0	*n*	0
petal	0	0	0	0	0
leaf	0	0	0	0	0

Fig. 2. The relation of edges and nodes in a genotype and the phenotype.

For representing the genotype with DG, each node is defined from the components of a real flower, and has parameters such as color, shape, and dimension of each component. Moreover, each edge has connection parameters such as position, orientation, and joint type for connection of nodes. Table 1 shows the parameters and values of a genotype within this DG. The range of colors represents the kind of colors in each part such as green, dark green, yellow, etc, and the shape represents the shape of rigid objects for its phenotype.

Table 1. Genotypic representation using directed graph.

DG	Component	stem	twig	petal	sepal	leaf
Node	Color	0~3	0~2	0~12	0~5	0~5
	Dimension	0~8	0~8	0~12	0~3	0~12
	Shape	Cylinder	Cylinder	Ellipsoid	Ellipsoid	Ellipsoid
Edge	Position	(x, y, z)	(x, y, z)	(x, y, z)	(x, y, z)	(x, y, z)
	Orientation	angle	angle	angle	angle	angle
	Joint type	fixed	fixed	fixed	fixed	fixed

3.2 Interactive Genetic Algorithm

GA proposed by John Holland in early 1970s applies some of natural evolution mechanisms such as crossover, mutation, and selection of the fittest to optimization and machine learning. GA is a very efficient search method, and has been applied to many problems concerning optimization and classification [19]. In IGA that is the similar method with GA except fitness evaluation part, a user evaluates the fitness to each individual in a population. IGA can interact with user and can stir up user's emotions or preferences in the course of evolution [20]. Therefore, the IGA is suitable for solving problems that cannot be easily solved by GA, like the generation of natural flower morphology in this paper.

The knowledge-based SDG is encoded by the information of their parts and nodes. The parts consist of dimensions, colors, shapes, and counts about each part of a flower, and the nodes consist of positions and orientations for connection and counts. The criterion used for the selection of individuals, who pass their genetic information from one generation to the next, is rank-based selection which is used to avoid undesirable convergence effects. Selection is not determined by the actual fitness value but by an individual's position within a fitness rank scale.

node	stem	twig	sepal	petal	leaf
stem	3	2	1	0	2
twig	0	2	1	0	4
sepal	0	0	0	10	0
petal	0	0	0	0	0
leaf	0	0	0	0	0

×

node	stem	twig	sepal	petal	leaf
stem	2	2	1	0	0
twig	0	2	1	0	2
sepal	0	0	0	12	0
petal	0	0	0	0	0
leaf	0	0	0	0	0

⇩

node	stem	twig	sepal	petal	leaf
stem	2	2	1	0	2
twig	0	2	1	0	2
sepal	0	0	0	10	0
petal	0	0	0	0	0
leaf	0	0	0	0	0

node	stem	twig	sepal	petal	leaf
stem	3	2	1	0	0
twig	0	2	1	0	2
sepal	0	0	0	12	0
petal	0	0	0	0	0
leaf	0	0	0	0	0

Fig. 3. An example of crossover operation in IGA.

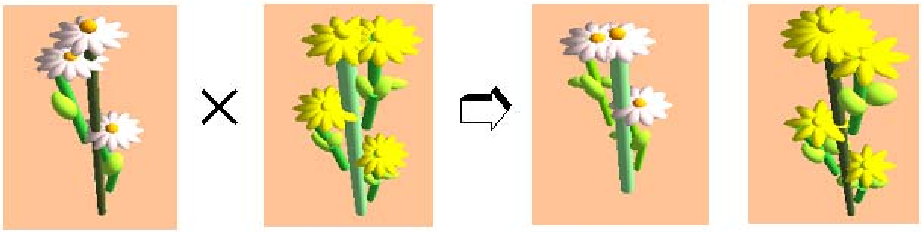

Fig. 4. An example of phenotypic representations with crossover in Figure 3.

Each individual in a population evolves to get higher fitness as it goes from generation to generation. A crossover operation exchanges edges and parameters in randomly selected parts between two individuals, and a mutation operation modifies the parameters in edges and nodes for the selected individual. We define the nodes in genotypic representation for IGA operations as the structures of a real flower such as stem, twig, sepal, petal, and leaf. We can create various phenotypes from the primitive structure of a flower by changing the values of the parameters in edges and nodes. For example, if the (stem, twig) and (sepal, petal, leaf) of nodes cross each other, the parameters in nodes and edges of 2 individuals selected to cross over each other. The relations of nodes and edges in DG are represented by 5×5 matrix, and an example of IGA is shown in Figure 3 and Figure 4 using this matrix.

4 Experimental Results

In this section we describe the experimental set-up to generate flower morphologies by the proposed method. First of all, we define the genotype of the flower structure by DG. Secondly, we represent each individual acquired by changing parameters in edges and nodes of each part, and evaluate each individual from IGA up to the last generation. We use an API program named 3D MathEngine that is now used in commercial to represent morphology and locomotion of individuals [21]. Using this program, we can represent and define a position and orientation of 3D rigid parts for generating individuals. We simplify the components of phenotypes and use rigid objects given by MathEngine such as a cube, cylinder, ellipsoid, and corn for shapes of each part.

We have developed the interface of the prototype system, and in IGA we have defined 6 individuals, 10 generations, and 5 levels of fitness evaluation. Nodes in DG are divided into 5 parts which are the stem, twig, leaf, sepal, and petal, and each node is connected to the edges of corresponding node by randomly generated parameters.

In this experiment, the conditions of IGA are given as follows.

- The number of individuals in a population is 6,
- The maximum number of generations is 10,
- The selection rate is 0.8,
- The crossover rate is 0.8,
- The mutation rate is 0.05,
- The fitness value is between 1 and 5 (the worst is 1), and
- IGA evaluation by 13 subjects.

Since IGA differs from the conventional GA due to humans' subjective evaluation, it is difficult to show the usefulness of IGA. We have carried out a convergence test and a schema analysis to evaluate the performance of this method.

4.1 Convergence Test

Convergence test uses the variation of mean values of fitness evaluations by subjects. To show the convergence of the method as an experimental result, we have requested 13 subjects to find natural flower morphology using the proposed method. Figure 5 shows the variations of fitness on average and the best, while subjects search for a more natural morphology. In this figure, the x-axis is the number of generations and the y-axis relates to the fitness values from 1 to 5. By the iteration of generations in IGA, we can observe that a more natural flower is generated.

4.2 Analysis of Schemas

This method analyzes the frequency of genes of individuals in populations at each generation of IGA process. It shows how the good characteristic schema is inherited through IGA. We define schemas such as the components like shape, color, and count of nodes for representing actual morphology. Each node consists of stem, twig, sepal, petal, and leaf, and the morphology of flower changes to various shapes by the defined schemas. Figure 6 shows how many schemas appear in populations in each generation and the phenotype having the best schema of fitness 5. The schemas selected in Figure 6 are shown in Table 2.

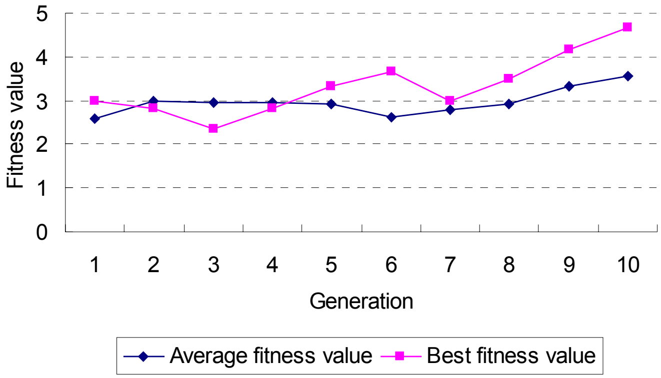

Fig. 5. Fitness changes on searching for natural flower.

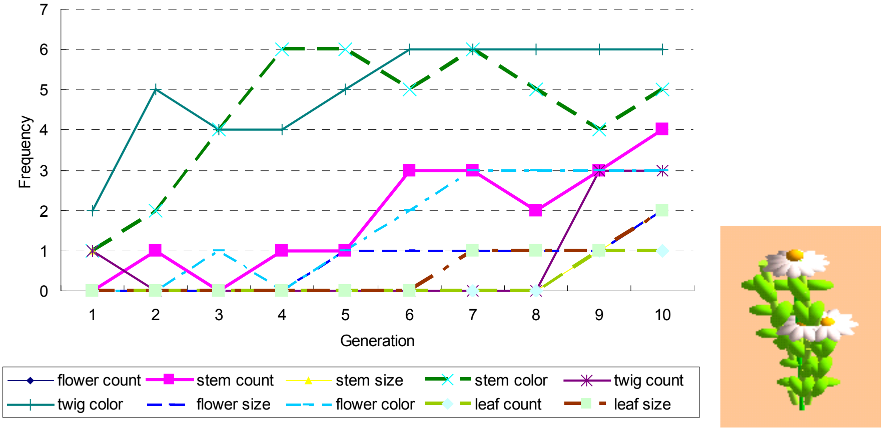

Fig. 6. The variation in generations of frequency of schemas and the fitness of populations.

Table 2. The schemas in the best solution of Figure 6.

Schema	Value	Schema	Value
Flower count	3	Stem count	4
Flower color	2 (Floral white)	Stem color	3 (Green)
Flower size	8 {0.12, 0.5, 0.1}	Stem size	0 {0.05, 1.2}
Twig count	12	Twig color	1 (Green yellow)
Leaf count	18	Leaf size	0 {0.3, 0.15, 0.1}

We have randomly selected an individual from the last population through the evaluation of a convergence test. In Figure 6, we realize that the characteristic schemas which do not appear in a population of the first generation increase. To analyze these schemas, we select the individual having the best fitness value in the last population. We analyze schemas of each population through the IGA process. Since we define ellipsoid shape for leaf, petal, and sepal and the cylinder shape for stem and twig, we define schemas for the shapes of each part as its size like {radius, height} or {x, y, z}.

To analyze the schemas included in the best solution, we define more effective ones which consist of natural flower morphology as the number of flowers, flower size, flower color, the number of stems, stem color, the number of twigs, twig color, the number of leaves, and leaf size. The size, color, and count of each part affect the evaluation to a more natural morphology similar to real flowers. Figure 6 shows the frequency of these schemas, and we can observe that they appear more and more as iterations even though they do not appear in individuals at the first population.

5 Conclusions

We have presented an automatic creation method of morphologies by using IGA which is suitable for sensitive problems due to humans' evaluation, and a creation of natural morphology to real character with the knowledge-based SDG which can create structures similar to real shapes. We have created flowers in three-dimensional world with evolutionary process.

We can reproduce the best characteristic parts by iteration of nodes in genotypic representation using SDG. Moreover, in contrast to previous approaches, our system lets users get direct feedback because it is represented structurally by SDG. In the future work we will focus on a representation of natural locomotion and the creation of other morphologies using SDG.

References

1. Langton, C. G.: Artificial life. *Santa Fe Institute Studies in the Sciences of Complexity*, vol. 6 (1989) 1-44
2. Bentley, P. J. and Corne, D. W.: *Creative Evolutionary Systems*. Morgan Kaufmann (2002)
3. Sims, K.: Artificial evolution for computer graphics. *Published in Computer Graphics*, vol. 25, no. 4 (1991) 319-328
4. Biles, J. A.: Life with GenJam: Interacting with a musical IGA. *Proc. of 1999 IEEE Int. Conf. on Systems, Man, and Cybernetics*, vol. 5, (1999) 652-656
5. Lindenmayer, B.: Mathematical models for cellular interaction in development I+II. *Journal of Theoretical Biology* (1968) 280-315
6. Hammel, M. and Prusinkiewicz, P.: Simulating the development of Fraxinus pennsylvanica shoots using L-systems. *Proceedings of the Sixth Western Computer Graphics Symposium* (1995) 49-58

7. Ochoa, G.: On genetic algorithms and Lindenmayer systems. *Parallel Problem Solving from Nature V* (1998) 335-344
8. Hornby, G. S. and Pollack, J. B.: Evolving L-systems to generate virtual creatures. *Computers & Graphics*, vol. 25 (2001) 1041-1048
9. Hornby, G. S. and Pollack, J. B.: Creating high-level components with a generative representation for body-brain evolution. *Artificial Life*, vol. 8, no. 3 (2002) 223-246
10. Sims, K.: Evolving virtual creatures. Computer Graphics, Annual Conference Series, *SIGGRAPH'94 Proceedings* (1994) 15-22
11. Sims, K.: Evolving 3D morphology and behavior by competition. *The Fourth International Workshop on the Synthesis and Simulation of Living Systems* (1994) 28-39
12. Lintermann, B. and Deussen, O.: Interactive modeling of plants. *IEEE Computer Graphics*, vol. 19, no. 1 (1999) 56-65
13. Lipson, H. and Pollack, J. B.: Automatic design and manufacture of robotic lifeforms. *Nature* 406 (2000) 974-978
14. Ventrella, J.: Explorations in the emergence of morphology and locomotion behavior in animated characters. *Proceedings of the Forth Workshop on Artificial Life* (1994) 463-441
15. Komosinski, M.: The framsticks system: Versatile simulator of 3D agents and their evolution. *The International Journal of Systems & Cybernetics*, vol. 32, no. 1/2 (2003) 156-173
16. Bongard, J. C. and Pfeifer, R.: Repeated structure and dissociation of genotypic and phenotypic complexity in artificial ontogeny. *Genetic and Evolutionary Computation Conference* (2001) 829-836
17. Kawamura, H. and Ohmori, H.: Computational morphogenesis of discrete structures with concideration of connection variables. *Memoris of the School of Engineering*, Nagoya Univ. vol. 53 (2002) 28-55
18. University of Oklahoma's botany 1114, "Flowers and flower morphology," http://bomi.ou.edu/bot1114/botany10/lab/flowers/flower10.html
19. Goldberg, D. E.: Genetic Algorithms in Search, *Optimization & Machine Learning*, Addison-Wesley Publishing Co. Inc. (1989)
20. Taylor, T. and Massey, C.: Recent developments in the evolution of morphologies and controllers for physically simulated creatures. *Artificial Life*, vol. 7, no. 1 (2001) 77-87
21. Kim, H.-S. and Cho, S.-B.: Application of interactive genetic algorithm to fashion design. *Engineering Applications of Artificial Intelligence*, vol. 13, no. 6 (2000) 635-644