MEMOIRS OF THE FACULTY OF SCIENCE, KYOTO UNIVERSITY, SERIES OF BIOLOGY
Vol. VIII, pp. 97–122, March 1980

K0L-System Simulating Almost but not Exactly the Same Development*

——Case of Japanese Cypress——

By

Taishin NISHIDA

Department of Biophysics, Faculty of Science
Kyoto University, Kyoto 606, Japan

(Received October 5, 1979)

ABSTRACT In this paper we introduce probabilistic 0L-system in order to investigate development of scale leaves of Japanese Cypress. We construct a special K0L-system to generate a set of tree-like structures which have as the same variety as natural trees.

We make observation of development of *Chamaecyparis Obtusa* (Japanese Cypress). Then we make probabilistic rewriting rules of K0L-system. The tree-like shapes produced by the rewriting rules are displayed on a CRT display to examin the rules and we amend them in order to produce more similar shapes to the observed trees. Finally we obtain a set of probabilistic rewriting rules which produces similar tree-like shapes to the observed trees.

We extract some elementary statistical features of the branching pattern, i.e. the number of symbols, the irregular branchings and the double branchings. The simulated trees have the same tendency in these statistical features with the natural trees, including the varieties.

For the foundation of the simulation, we construct a basic theory of K0L-system. Some results concerning the number of symbols in strings generated by K0L-system are obtained.

0. Introduction

In case of natural trees, it seems that individual trees belonging to a species have their common shape. But, when we investigate them in detail, we find some differences among them. This naturally occurs, even if we examine shapes of the trees, which have exactly the same genetic substance i.e. belong to the same clone. When the genomes are realized as phenotypes, morphological variety might occur because of slight environmental changes or the like. Thus far the authors, Honda (1971), Frijters & Lindenmayer (1974) and Hogeweg & Hespor (1974) and so on

* "K0L-system" is the abbreviation of Japanese word "kakuritsuteki" 0L-system, which means "Probabilistic". Since P is already another standard abbreviation, we adopt K in this way.

have used deterministic models for simulating morphological development and drawing tree shapes by means of computers. We are interested in the contrast between similarity and nonuniformity of nature, that is, "almost but not exactly the same", and so make a mathematical model which might simulate such phenomena.

As the theoretical tool we introduce here the probabilistic 0L-system in order to investigate development of scale leaves of Japanese Cypress. We construct a special PK0L-system to generate a set of tree-like structures which have as the same variety as natural trees. For references on 0L-system and generally on Lindenmayer system, see Herman & Rozenberg (1975).

For our purpose, we took the branching structure of apical parts (10–15 centimeters from the apex of each branch) among many possible materials. This is because, especially for young trees, apical part determines the shape of the whole tree.

To describe development of tree we use a P0L-system, which plays the role of deterministic part in our model. Since we cannot decide precisely various environmental effects on the apical parts, for example, of sun shine, water, temperature, CO_2 and nutrients, we introduce the independent stochastic process in order to express the variety of growths.

We selected *Chamaecyparis Obtusa* (Japanese Cypress) as the sample tree. The reason is as follows:

(1) The apical part of cypress develops in a plane, so that we may avoid the complexity of three dimensional branching structure.

(2) Since it develops by scale leaves, a symbol of 0L-system naturally corresponds to a scale leaf.

This paper contains the following parts:

(1) To culture young trees of *Chamaecyparis Obtusa* and make observation of their development.

(2) To make rules of K0L-system.

(3) Simulation of the K0L-system on a computer.

(4) Comparison of the results of the simulation with data from the cultured trees.

(5) Mathematical considerations of K0L-system.

(6) Discussion.

1. Observation of development of Japanese Cypress

The Cypress trees under the observation are shown in Figure 1a. These trees were cultured at the Forest, Faculty of Agriculture, Kyoto University. These trees are about 2–3 meters high. We set up eleven observing points on these trees. These observing points are at apical parts of trunks or branches which emerge from trunks or from other branches. Heights of the observing points are about 1 meter from the earth. A part of branch, which is mounted on the rectangular photographic stage of 14×13 centimeters is called an apical part.

We took the photographs of these observing points once a week from the tenth of April till the thirteenth of July in 1978. Some of them are shown in Figure 1b, 1c, 1d and 1e. The observed trees began to grow early in April and continued to grow till autumn. But during the summer they developed so greatly in length and width

Figure 1a. The Cypress trees under the observation and the author.

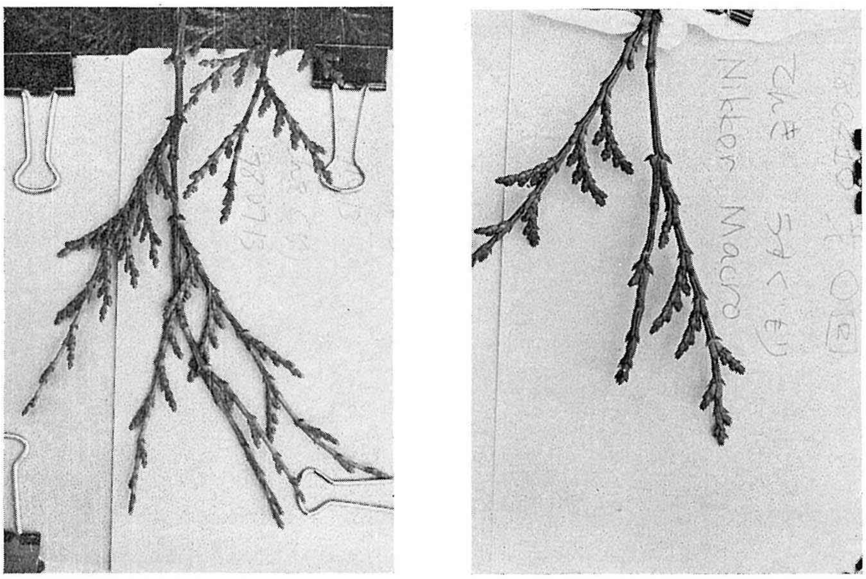

Figure 1b. Photos of the observing point H–4.

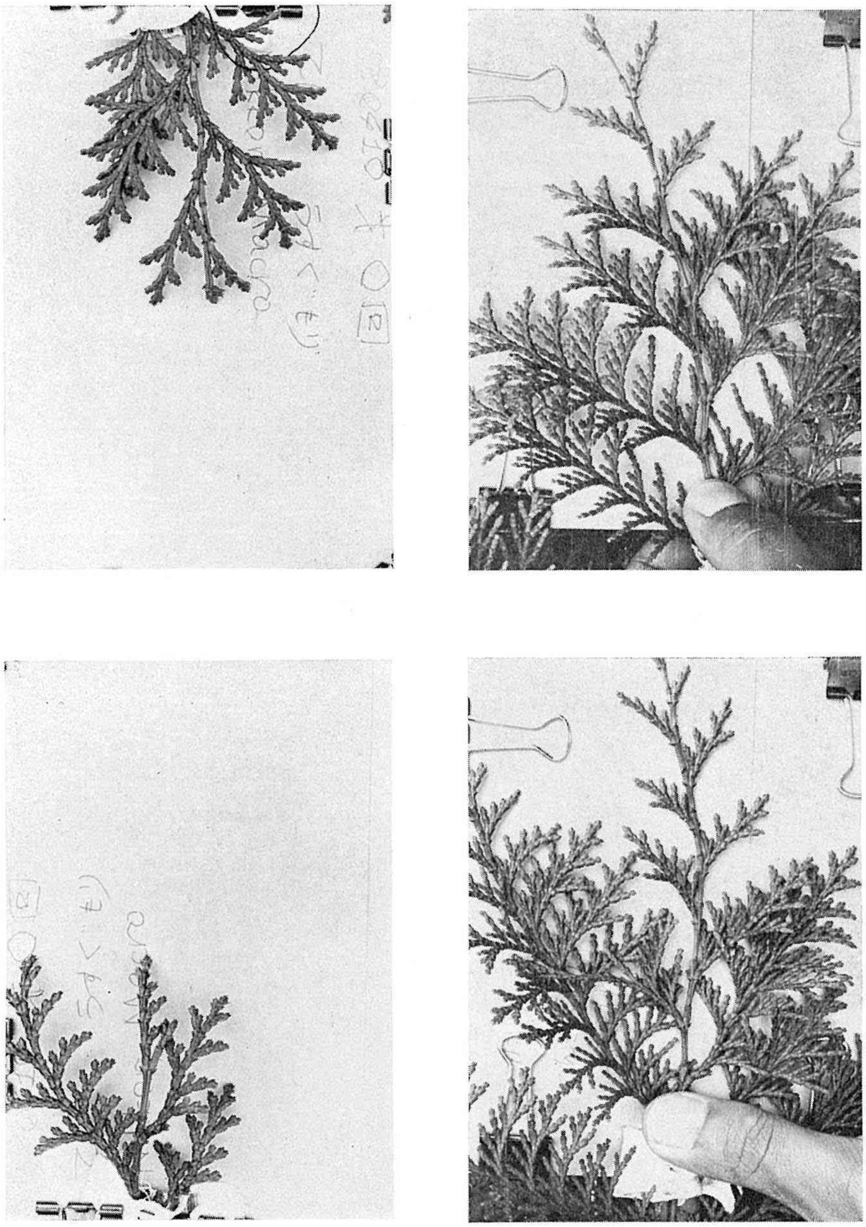

Figure 1c. Photos of the observing points H–5 and H–6.

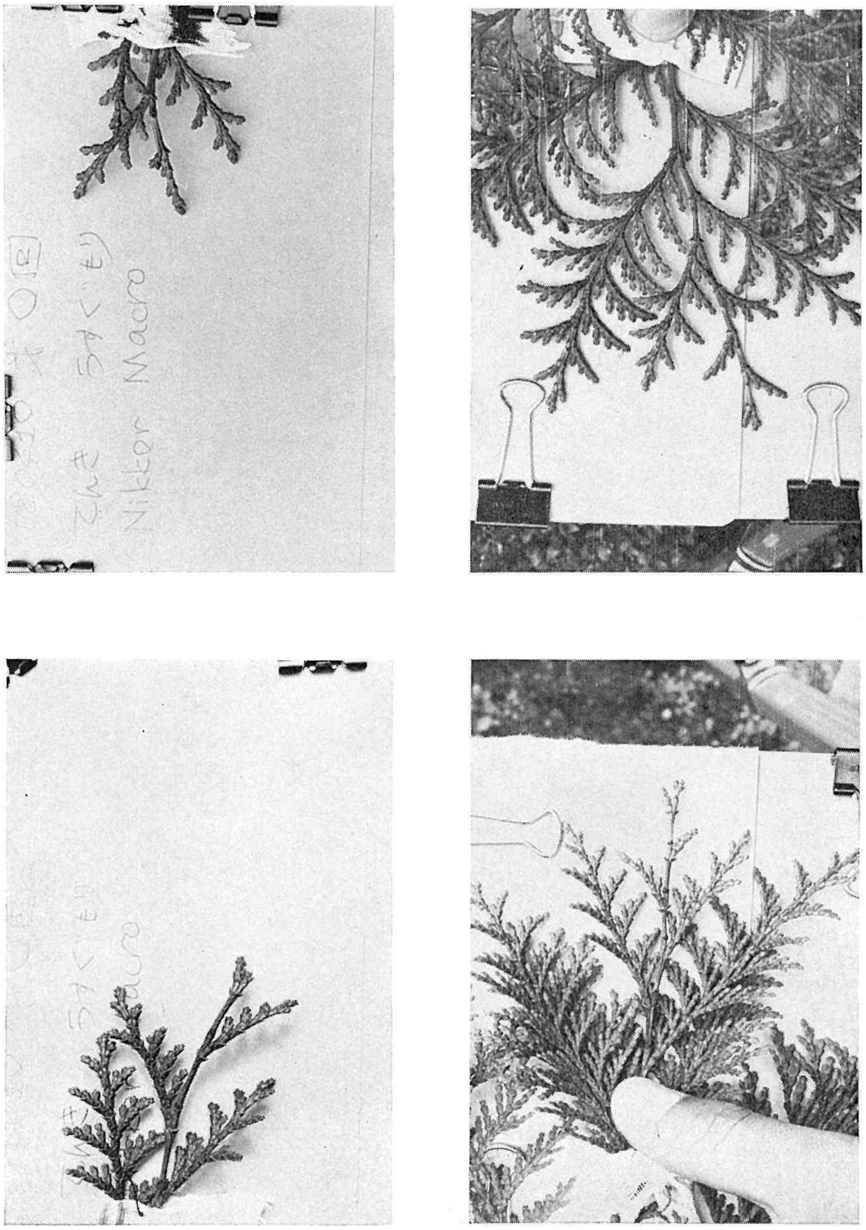

Figure 1d. Photos of the observing points H–7 and H–9.

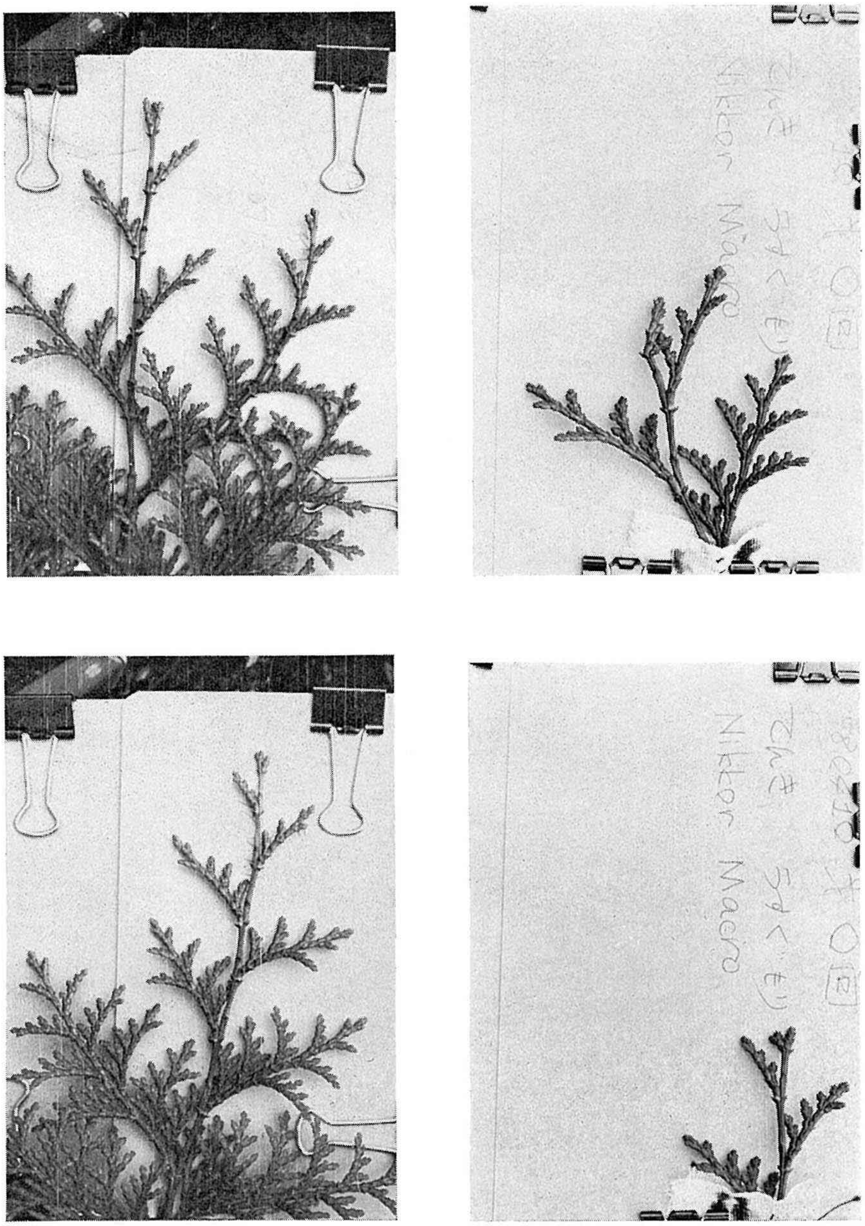

Figure 1e. Photos of the observing points H–10 and H–11.

that they overscaled our photographic stage. An apex of a tree (observing point H–4) grew by more than 20 centimeters in length during the observing period. Furthermore H–4 grew in a different manner from the other observing points. So we have not used the data of H–4.

The other observing points grew by about seven scale leaves or by 10–15 centimeters in the direction of central axis.

2. Rewriting rules

Formally K0L-system is a string generating system, where a string is a sequence of symbols such as "ABC+D/A". A K0L-system consists of a starting string called the axiom and a set of rewriting rules. A K0L-system generates a new string by rewriting the symbols of old string according to the rewriting rules. A string can be interpreted as a tree-like structure. For example the string "ABC+D/A" represents the branch shown in Figure 2. Our main work is to make the set of rewriting rules of K0L-system which might represent the growth of our trees well.

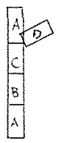

Figure 2. The tree-like structure corresponds to "ABC+D/A"

We use the alphabet {A, B, C,..., Z} for expressing states of the scale leaf and +, −, / for representing branching points. + represents the branching point of right branch, − that of left branch and / the end point of a branch, respectively.

From the biological point of view the rewriting rules must satisfy the following conditions;

1. Since only the apex develops, only the symbols corresponding to it should be rewritten by two or more symbols.
2. No symbol disappears.
3. In principle, when we neglect the branching symbols, one symbol is rewritten with at most two symbols.
4. The system to be established may generate indefinitely large simulated trees. The trees generated by the K0L-system should match well with the real trees at early simulation steps, though this is not required for later steps than a predetermined step.

From the observation in section 1 we draw several growth diagrams shown in

Figure 3. From the diagrams we can see that the apex of a branch can be rewritten by one of the following rewriting rules in principle,

$$\left.\begin{array}{l} A \rightarrow CB \\ B \rightarrow DA \\ C \rightarrow E+B/ \\ D \rightarrow E-A/ \\ E \rightarrow E \end{array}\right\} P1$$

Figure 4 shows the strings which are successively generated from A (the axiom of the system is A) up to the eighth step using the rules of P1 and the tree-like structures which correspond to them. These tree-like shapes are not necessarily similar to the natural trees.

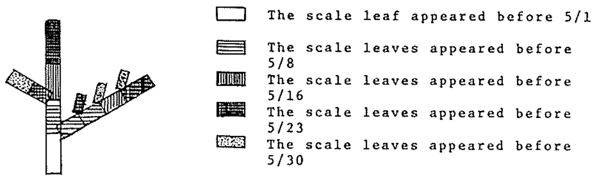

Figure 3. One of the growth diagram, this shows the growth of H-3.

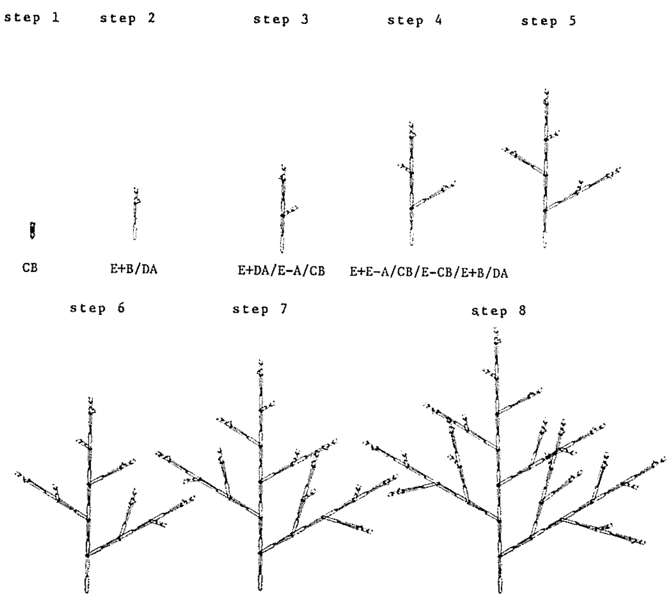

Figure 4. Development of P1.
The strings for steps 5, 6, 7 and 8 are omitted because they are very long.

In fact when they are compared with the natural trees, we find differences among them in the speed of growth and the direction of branches. The speed of growth of the observed trees are shown in Figures 5a, 5b and 5c. Where the branching level for the main axis is 0 and the branching level for the other branches are obtained by adding one to the value of its mother branch at the occasion of each branching. The growth speed varies among the branching levels. Therefore, we must use the other symbols to represent the side buds instead of A and B in the rules $C \to E+B/$ and $D \to E-A/$ in P1. We can adjust the average growth speed by introducing the probability. That is, we consider the following rules,

$$A \to CB \quad p \qquad B \to DA \quad p$$
$$A \to A \quad q \qquad B \to B \quad q$$

where p and q are the selecting probability and $p+q=1$.

Since the average length of the strings of k steps starting from an A is $pk+1$, we have

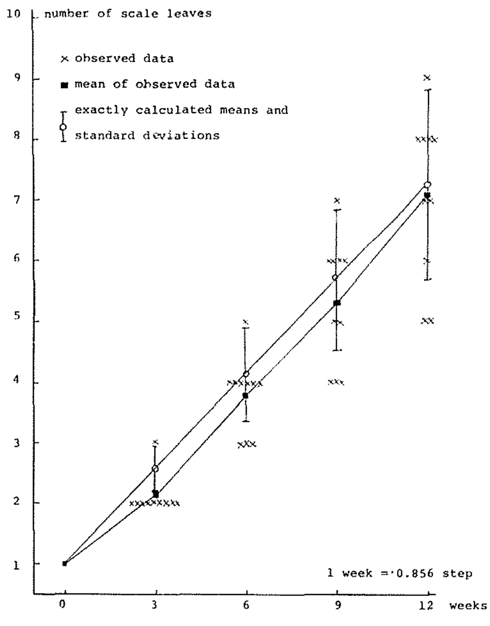

Figure 5a. The growth of branching level 0.

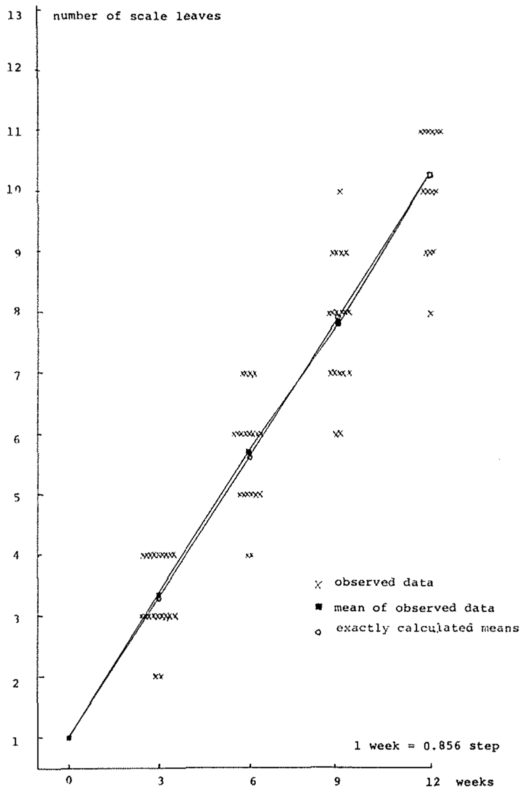

Figure 5b. The growth of branching level 1.

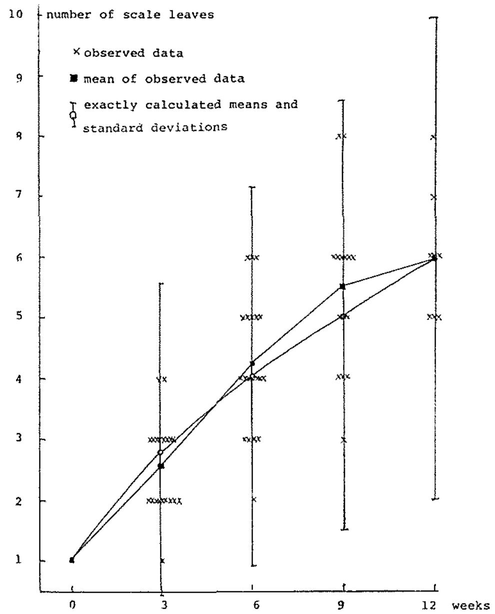

Figure 5c. The growth of branching level 2. 1 week=0.856 step

the growth speed of the natural trees shown in Figure 5a by putting $p=0.61$. The exactly calculated means and standard deviations in Figures 5a, 5b and 5c are obtained from the rewriting rules shown in Table 1 using the method of generating function, which will be explained in section 5.

One can see that in the natural trees the basal parts of the branches with branching level more than one are likely to branch toward the opposite direction to their direction. These phenomena greatly affect the shapes of branches and perhaps have physiological significance. We use several symbols to simulate these phenomena well. For example symbols L and P (M and Q) in Table 1 produce right (left) branches only.

We again and again examined the rules by displaying the tree-like shapes produced with them on a CRT-display and amended the rules in order to produce more realistic shapes. At the beginning we often added some new symbols and rules to the set of rules, but at the end we could adjust the tree-like shapes only by changing the probability. The final result of the procedure is shown in Table 1.

Table 1. Rewriting rules. The axiom (starting string) is ‘A’.

A → CB	0.61	F → JH	0.7	L → NP	0.6
A → A	0.39	F → JF	0.3	L → NL	0.4
B → DA	0.61	G → KI	0.7	M → OQ	0.6
B → B	0.39	G → KG	0.3	M → OM	0.4
C → E+F/	1	H → JI	1	N → N	0.6
D → E−G/	1	I → KH	1	N → V+R/	0.39
E → E	1	J → V−L/	0.99	N → V−S/+R/	0.01
		J → V−L/+M/	0.01	O → O	0.6
R → WR	0.23	K → V+M/	0.99	O → V−S/	0.39
R → OR	0.02	K → V−L/+M/	0.01	O → V−S/+R/	0.01
R → R	0.75	V → V	1	P → P	0.5
S → WS	0.23			P → NT	0.5
S → NS	0.02			Q → Q	0.5
S → S	0.75			Q → OU	0.5
W → W	1			T → T	0.65
				T → NU	0.35
				U → U	0.65
				U → OT	0.35

Table 2. Rewriting rules for calculation.

A → CB	0.61	F → JH	0.7	L → NP	0.6
A → A	0.39	F → JF	0.3	L → NL	0.4
B → DA	0.61	G → KI	0.7	M → OQ	0.6
B → B	0.39	G → KG	0.3	M → OM	0.4
C → E+F/	1	H → JI	1	N → N	0.6
D → E−G/	1	I → KH	1	N → X+R/	0.39
E → E	1	J → V−L/	0.99	N → X−S/+R/$_2$	0.01
		J → V−L/+M/$_1$	0.01	O → O	0.6
R → WR	0.23	K → V+M/	0.99	O → X−S/	0.39
R → YR	0.02	K → V−L/+M/$_1$	0.01	O → X−S/+R/$_2$	0.01
R → R	0.75	V → V	1	P → P	0.5
S → WS	0.23			P → NT	0.5
S → YS	0.02			Q → Q	0.5
S → S	0.75			Q → OU	0.5
W → W	1			T → T	0.65
Y → Y	1			T → NU	0.35
				U → U	0.65
				U → OT	0.35
				X → X	1

3. Simulations on a computer

Figure 6 shows the programs and flow of data with which the simulations are performed. These programs ran on the FACOM M–190 and FACOM 230–48 at the Data Processing Center, Kyoto University.

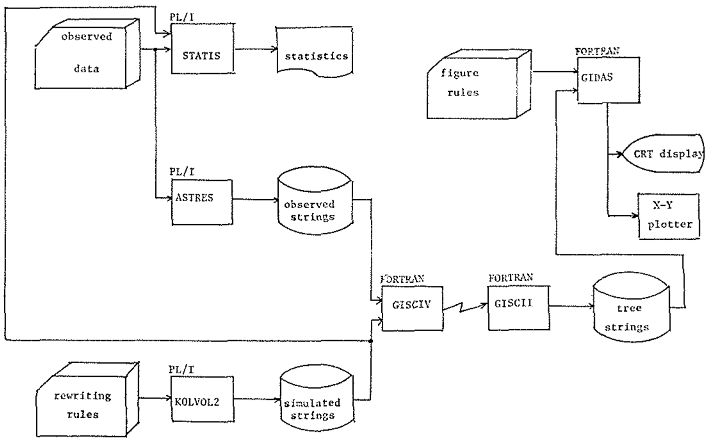

Figure 6. The programs and the data flow.

KOLVOL2 plays the most important role of the simulation. KOLVOL2 reads the rewriting rules and computes the probability distribution functions of these rewriting rules. Then KOLVOL2 generates strings one after another starting from a given axiom (initial string). For every symbols in the string of previous step KOLVOL2 produces a random number x which is greater than 0 and less than 1, and selects the rewriting rule which has x as the value of its distribution function, then the symbol is rewritten with the selected rule. When all the symbols in a string are rewritten in this way, a derivation of one step is finished and a string of the next step is generated. A different initial value of the random number generator is given at each time of simulation. The length of the string generated is less than 1200 symbols because of the limitation of memory capacity of the computer.

GIDAS interprets the strings and draws the tree-like shapes on CRT-display or X–Y plotter. Every symbol has its basic figure and the relation between the symbol and the basic figure is called the figure rule. GIDAS replaces the symbols with figures according to the figure rule. Whenever it finds the branching symbols (+ or −), GIDAS makes a right (+) or left (−) branch which has the angle calculated by the following formula,

$\frac{\pi}{2+m}$ (rad) where m is the branching level.

We explain STATIS in the next section. GISCIV, GISCII and ASTRES are used for conversion of data-form in the computer. Therefore they do not participate in the main body of simulation.

4. Comparison of simulation with observed trees

Figure 7 shows the tree-like shapes produced with the rules of Table 1 and Figure 8 the observed trees drawn by GIDAS using the same figure rules as in Figure 7.

As the examples of their statistical features, we count the number of symbols, irregularities of branchings and double branchings (right and left branches attached to the same point). These features are illustrated in Figure 9, 10 and 11. The "exactly calculated means" in these figures are obtained by means of the averaged growth matrix, which will be explained in section 5. The simple "means and standard deviations" are calculated from the values of simulated trees and observed trees. It is impossible to get the number of irregularities of branchings with the method of averaged growth matrix. The program STATIS is used to extract these statistical features.

The normalized number of symbols denotes the number of all symbols in each branching level divided by the number of symbols in the main axis (i.e. the branch of branching level 0), and are shown in Figure 9. We think that these values indicate the width of a branch compared with the length and the density of the scale leaves. A branch which spreads greatly toward right and left has a large value at the branching level 1. A branch in which the scale leaves are dense has large value at the branching level 2 and 3. From the Figure 9 we see that the simulated trees have the same tendency as the natural trees.

In general, branches grow to the right and the left alternatively. But we sometimes observe irregular branchings. So we define here an index of irregularity. Let Ir denote the number of irregularities of branchings. When we check an axis from its base to its apex, we define the new value of Ir by the following formula whenever we find a branching point,

$$\mathrm{Ir}=\begin{cases}\mathrm{Ir}+1 & \text{if this branch has the same direction with the previous branch.}\\ \mathrm{Ir} & \text{if this branch has the opposite direction with the previous branch.}\end{cases}$$

For example the irregularities of branchings for Figure 12 are 2. Figure 10 shows the normalized values of number of irregularities of branchings, that is, the total number of irregularities of branchings divided by the total number of branching points at each branching level. The number of irregularities tells us whether the

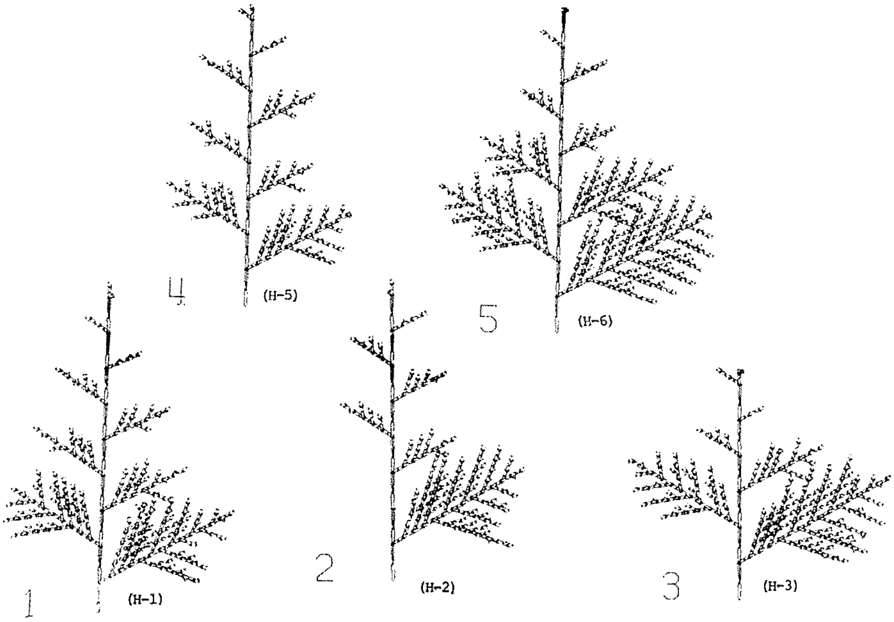

Figure 7a. The simulated trees. (15 step)

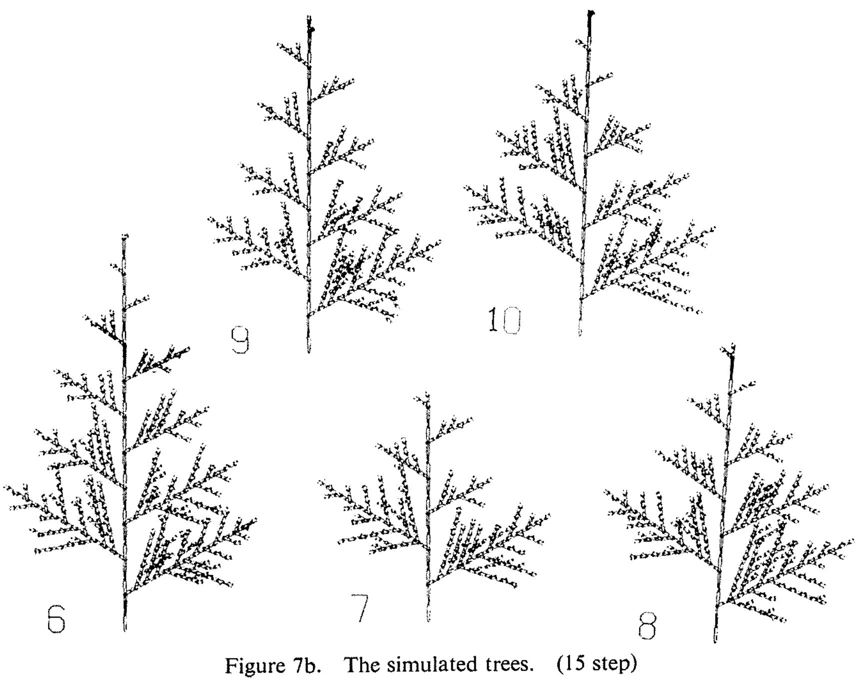

Figure 7b. The simulated trees. (15 step)

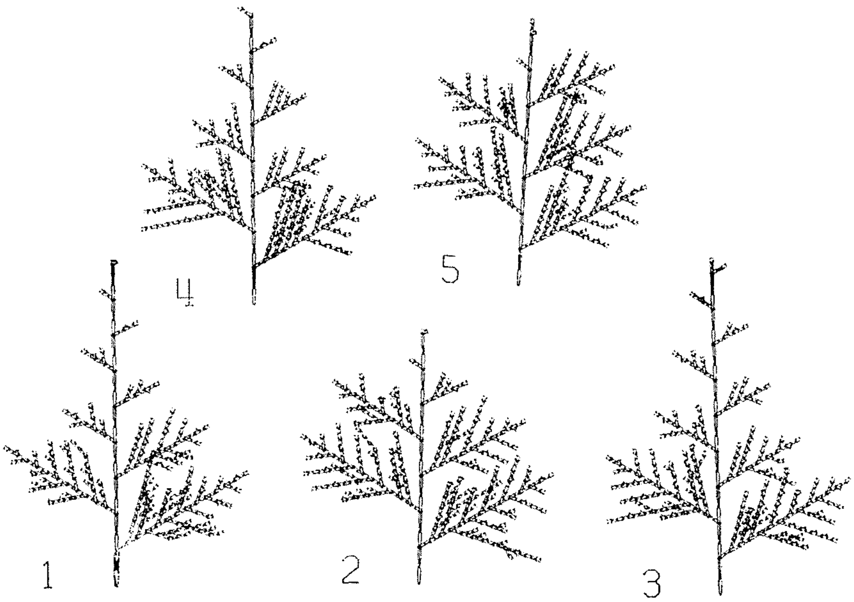

Figure 8a. The observed trees. (observing point number)

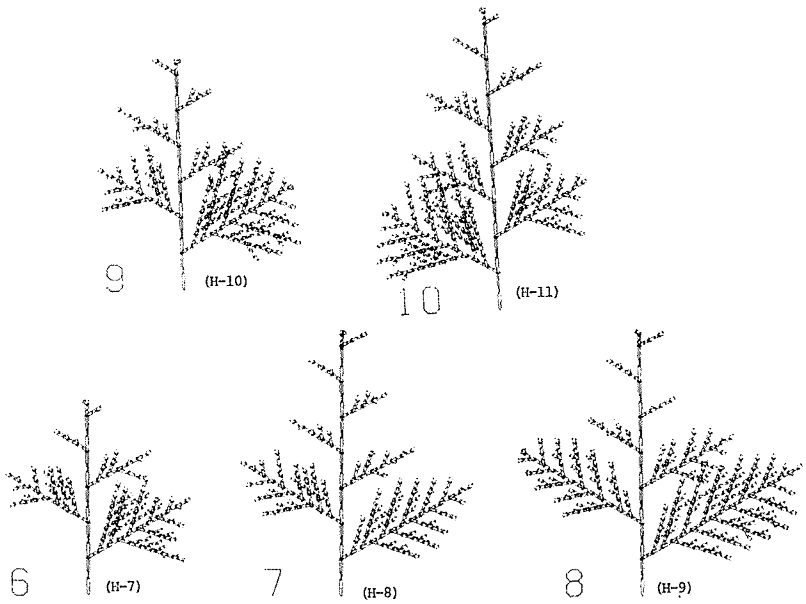

Figure 8b. The observed trees. (obsreving point number)

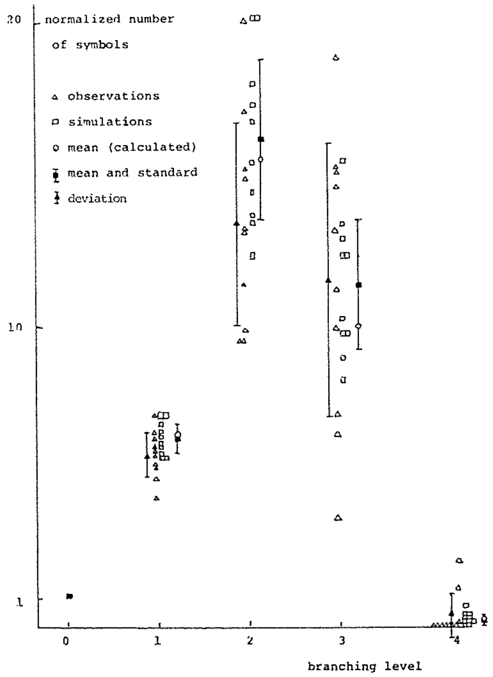

Figure 9. Normarized number of the symbols.

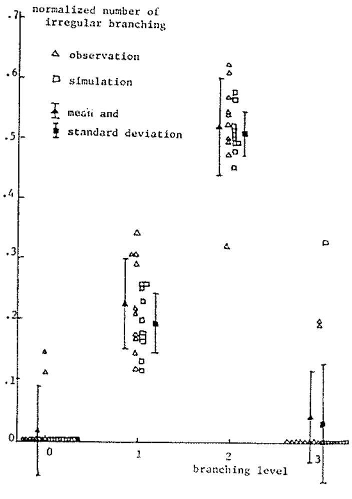

Figure 10. Irregular branching.

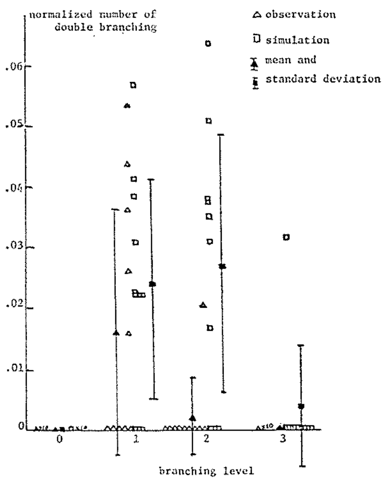

Figure 11. Double branching.

shape of a branch is well-ordered or not. Of course, the less the irregularities, the more the branch is well-ordered. From the Figure 10 we see that the values of simulated trees are in the ranges of the standard deviations of the values of natural trees. The natural trees have larger standard deviations than simulated trees because the number of irregularities is "integrated" value. That is, since at the simulation individual small branches grow independently and the integration of these small branches is the number of irregularities of branchings for the tree, the varieties are canceled. On the other hand the small branches in the natural trees have mutual interactions. Therefore the branches on different trees have larger varieties than simulated trees.

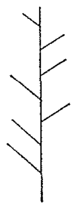

Figure 12. Example of the irregular branching.

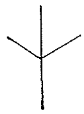

Figure 13. Example of the double branchings.

When a pair of right and left branches are attached to the same branching point as illustrated in Figure 13, we call this phenomenon a double branching. Figure 11 shows the total number of double branchings divided by the total number of branching points at each branching level. These phenomena scarcely happen and affect the shape of branches very little. So we consider this feature as the exceptional phenomena. When we made the rewriting rules, we did not consider double branchings so the simulated trees differ from the natural ones in this respect.

5. Definition and basic results of K0L-system

In this section we give the formal definition of K0L-system and some basic results concerning with the number of symbols in strings generated by it. Thus far, only Jürgensen's work (1975) has appeared as for the theory of probabilistic L-system. He formulated a rather complicated Markovean model. But we need a simpler formulation.

A K0L-system G is a triple $G=<\Sigma, P, \omega>$ where Σ is a finite set of symbols called the alphabet, P is a finite subset of $\Sigma \times \Sigma^* \times (0, 1]$ called the probabilistic rewriting rule and ω is a string in Σ^* called the axiom. (Σ^* is the set of all strings made of the symbols from Σ including the null string λ, i.e. the string of length zero.)

For each σ in Σ there exists at least a string x in Σ^* and a positive number α in (0, 1] such that (σ, x, α) is in P. For every fixed σ, $\sum_{(\sigma,x,\alpha)\in P} \alpha=1$. For the convenience we write $\sigma\to(x, \alpha)$ instead of $(\sigma, x, \alpha)\in P$. Throughout this paper we assume $\Sigma=\{\sigma_1, \sigma_2,\ldots, \sigma_n\}$. For every σ_i in Σ we enumerate the rules such as $\sigma_i\to(x_{i1}, \alpha_{i1})$, $\sigma_i\to(x_{i2}, \alpha_{i2}),\ldots, \sigma_i\to(x_{im_i}, \alpha_{im_i})$ where m_i is the number of rules which starts from σ_i. Let P_{ij} denote the single rule $\sigma_i\to(x_{ij}, \alpha_{ij})$ and let $P_i=\bigcup_{j=1}^{m_i} P_{ij}$.

Similarly to the case of 0L-system, we define the derivation $\overset{*}{\underset{G}{\Longrightarrow}}$ in K0L-system $G=<\Sigma, P, \omega>$. That is, for two strings $x=\sigma_{i_1}\sigma_{i_2}\cdots\sigma_{i_l}$ with $\sigma_{i_k}\in\Sigma$ $(1\le k\le l)$ and $y=y_{i_1j_1}y_{i_2j_2}\cdots y_{i_lj_l}$ with $y_{i_kj_k}\in\Sigma^*$ $(1\le k\le l)$, we write $x\underset{G}{\Longrightarrow}y$ if $\sigma_{i_k}\to(y_{i_kj_k}, \alpha_{i_kj_k})$ is in P_{i_k} $(1\le k\le l)$. For two strings x, $y\in\Sigma^*$ we write $x\overset{*}{\underset{G}{\Longrightarrow}}y$, if there exist $n+1$ strings $x=x_0, x_1, x_2,\ldots, x_n=y$ $(n\ge 0)$ such that $x_{j-1}\underset{G}{\Longrightarrow}x_j$ for $1\le j\le n$. Let $x\underset{G}{\Longrightarrow}y$ be a derivation in K0L-system, there can be several paths from x to y, that is, if $x=\sigma_{i_1}\sigma_{i_2}\cdots\sigma_{i_l}$ and y has k different partitions $y=y^{(1)}_{i_1j_1}y^{(1)}_{i_2j_2}\cdots y^{(1)}_{i_lj_l}=y^{(2)}_{i_1j_1}y^{(2)}_{i_2j_2}\cdots y^{(2)}_{i_lj_l}=\cdots=y^{(k)}_{i_1j_1}y^{(k)}_{i_2j_2}\cdots y^{(k)}_{i_lj_l}$ with $y^{(m)}_{i_rj_r}\in\Sigma^*$ and $\sigma_{i_r}\to(y^{(m)}_{i_rj_r}, \alpha^{(m)}_{i_rj_r})$ in P_{i_r} $(1\le r\le l,\ 1\le m\le k)$, then the derivation $x\underset{G}{\Longrightarrow}y$ has k paths. Let $x\underset{G}{\Longrightarrow}y$ be a derivation described above, the probability of the derivation p_{xy} is defined by,

$$p_{xy}=\sum_{m=1}^{k}\prod_{r=1}^{l}\alpha^{(m)}_{i_rj_r}. \tag{1}$$

It is easily verified that for each string $x\in\Sigma^*$,

$$\sum_{\text{for all } y\ x\underset{G}{\Rightarrow}y} p_{xy}=1. \tag{2}$$

Now we consider sequence of probability spaces $\mathfrak{L}_n(G)\subset\Sigma^*\times[0, 1]$. For a K0L-system $G=<\Sigma, P, \omega>$ $\mathfrak{L}_n(G)$'s are inductively defined as follows,

$$\mathfrak{L}_0(G)=(\omega, 1) \tag{3}$$

$$\mathfrak{L}_n(G)=\{(x, p)\,|\,(x', p')\in\mathfrak{L}_{n-1}(G) \text{ and } x'\underset{G}{\Longrightarrow}x,\ p=\sum_{\text{for all }(x',p')} p'p_{x'x}\}.$$

Since for every (x, p) in $\mathfrak{L}_n(G)$ there are its some ancestors in $\mathfrak{L}_{n-1}(G)$ and for every (x', p') in $\mathfrak{L}_{n-1}(G)$ there are its some descendants in $\mathfrak{L}_n(G)$, the following equations hold,

$$\begin{aligned}\sum_{(x,p)\in\mathfrak{L}_n(G)} p &= \sum_{(x,p)\in\mathfrak{L}_n(G)}\ \sum_{(x',p')\in\mathfrak{L}_{n-1}(G)} p'p_{x'x}\\ &= \sum_{(x',p')\in\mathfrak{L}_{n-1}(G)} p' \sum_{\substack{(x,p)\in\mathfrak{L}_n(G)\\ \text{s.t. } x'\underset{G}{\Rightarrow}x}} p_{x'x}\\ &= \sum_{(x',p')\in\mathfrak{L}_{n-1}(G)} p'.\end{aligned} \tag{4}$$

Therefore, by induction $\mathfrak{L}_n(G)$'s are really probabilistic spaces.

Here let us focus our attention on the number of symbols in strings generated by G. For $x \in \Sigma^*$ and $\sigma_i \in \Sigma$, $\#_{\sigma_i} x$ denotes the number of occurrences of σ_i in x. $(\#_{\sigma_1} x, \#_{\sigma_2} x, \ldots, \#_{\sigma_n} x)$ is called the Parikh vector of x and denoted by $[x]$. For every $P_i \subset P$ $(1 \leq i \leq n)$ let us pick up one rewriting rule P_{ij_i} $\sigma_i \rightarrow (x_{ij_i}, \alpha_{ij_i})$ in P_i, then a transition of Parikh vector space is determined by the selected rules $\{P_{1j_1}, P_{2j_2}, \ldots, P_{nj_n}\}$. This transition is represented by an $n \times n$ matrix $M(P_{1j_1}, P_{2j_2}, \ldots, P_{nj_n})$ and defined as follows,

$$M(P_{1j_1}, P_{2j_2}, \ldots, P_{nj_n}) = \begin{pmatrix} [x_{1j_1}] \\ [x_{2j_2}] \\ \vdots \\ [x_{nj_n}] \end{pmatrix}. \tag{5}$$

An $i-k$ component a_{ik} of $M(P_{1j_1}, P_{2j_2}, \ldots, P_{nj_n})$ represents the number of σ_k's produced by σ_i with a rewriting rule $\sigma_i \rightarrow (x_{ij_i}, \alpha_{ij_i})$ where $\#_{\sigma_k} x_{ij_i} = a_{ik}$. Let $\mathscr{M}$ denotes the set of all such $M(P_{1j_1}, P_{2j_2}, \ldots, P_{nj_n})$'s. An element of $\mathscr{M}$ will often be written as M. Of course, the cardinality of $\mathscr{M}$ is $\prod_{i=1}^{n} m_i$ where m_i is the cardinality of P_i. For a derivation $x \underset{G}{\Longrightarrow} y$ there are some M's in $\mathscr{M}$ such that $[y] = [x]M$. For every M in $\mathscr{M}$, let M be obtained from the set of rules $\sigma_1 \rightarrow (x_{1j_1}, \alpha_{1j_1})$, $\sigma_2 \rightarrow (x_{2j_2}, \alpha_{2j_2})$, $\ldots$, and $\sigma_n \rightarrow (x_{nj_n}, \alpha_{nj_n})$. Then the selection probability $f(M)$ is defined by,

$$f(M) = \prod_{i=1}^{n} \alpha_{ij_i}. \tag{6}$$

Now we consider,

$$f(\mathscr{M}) = \sum_{M \in \mathscr{M}} f(M) = \sum_{\sigma_1 \rightarrow (x_{1j_1}, \alpha_{1j_1}) \text{ in } P_1} \; \sum_{\sigma_2 \rightarrow (x_{2j_2}, \alpha_{2j_2}) \text{ in } P_2} \cdots \sum_{\sigma_n \rightarrow (x_{nj_n}, \alpha_{nj_n}) \text{ in } P_n} \; \prod_{i=1}^{n} \alpha_{ij_i}. \tag{7}$$

Since the summation ranges over all the rules in P_i $(1 \leq i \leq n)$, the summation can be exchanged with the multiplication. Thus we have $f(\mathscr{M}) = 1$ and $(\mathscr{M}, f)$ is a probability space. Now we have the averaged growth matrix $\overline{M} = \sum_{M \in \mathscr{M}} f(M)M$. For a KOL-system $G = <\Sigma, P, \omega>$ π denotes the initial vector that is $\pi = [\omega]$. For a sequence of probability spaces $\mathfrak{L}_n(G)$, $\overline{[x]}_n$'s denote the mean Parikh vector of $\mathfrak{L}_n(G)$.

Theorem 1 Let $G = <\Sigma, P, \omega>$ be a KOL-system and $\overline{M}$ the averaged growth matrix for G, then,

$$\overline{[x]}_n = \pi \overline{M}^n. \tag{8}$$

Proof Let (x, p) be in $\mathfrak{L}_n(G)$, for a fixed derivation s_x from ω to x which has $n+1$

strings $\omega = x_0, x_1, \dots, x_n = x$ with $x_i \in \mathfrak{L}_i(G)$ and $x_{i-1} \underset{G}{\Longrightarrow} x_i$ $(1 \le i \le n)$ $\mathscr{M}_i(s_x)$ denotes the set of all M such that $[x_i] = [x_{i-1}]M$ $(1 \le i \le n)$. Let $M_i \in \mathscr{M}_i(s_x)$, then

$$p = \sum_{\text{all } s_x \text{ of } \omega \overset{*}{\underset{G}{\Longrightarrow}} x} \; \sum_{M_i \in \mathscr{M}_i(s_x)} \prod_{i=1}^{n} f(M_i). \tag{9}$$

The second summation ranges over all sequences $M_1, M_2, \dots, M_n$ with $M_i \in \mathscr{M}_i(s_x)$. Hence,

$$p[x] = \sum_{\text{all } s_x \text{ of } \omega \overset{*}{\underset{G}{\Longrightarrow}} x} \; \sum_{M_i \in \mathscr{M}_i(s_x)} \prod_{i=1}^{n} f(M_i) \pi M_i. \tag{10}$$

Thus we obtain,

$$\begin{aligned} \overline{[x]}_n &= \sum_{(x,p) \in \mathfrak{L}_n(G)} p[x] \\ &= \pi \sum_{(x,p) \in \mathfrak{L}_n(G)} \; \sum_{\text{all } s_x \text{ of } \omega \overset{*}{\underset{G}{\Longrightarrow}} x} \; \sum_{M_i \in \mathscr{M}_i(s_x)} \prod_{i=1}^{n} f(M_i) M_i. \end{aligned} \tag{11}$$

Since the triple summations ranges over all possible sequences of M in $\mathscr{M}$ with length n,

$$\overline{[x]}_n = \pi \Big(\sum_{M \in \mathscr{M}} f(M) M \Big)^n = \pi \overline{M}^n.$$ ▶

Corollary 2 Let η be an n-dimensional column vector with $\eta = (1, 1, \dots, 1)^t$, then the mean length of strings in $\mathfrak{L}_n(G)$ is given by,

$$m(k) = \pi \overline{M}^k \eta. \tag{12}$$ ▶

Since the definition of $\overline{M}$ $(\overline{M} = \sum_{M \in \mathscr{M}} f(M) M)$ is not suitable for the calculation, let us consider another formulation. That is, for every $P_i \subset P$ $(1 \le i \le n)$ the averaged Parikh vector $\overline{[x_i]}$ produced by σ_i is defined by,

$$\overline{[x_i]} = \sum_{\sigma_i \to (x_{ij_i}, \alpha_{ij_i}) \text{ in } P_i} \alpha_{ij_i} [x_{ij_i}]. \tag{13}$$

Then one can easily see $\overline{M} = \begin{pmatrix} \overline{[x_1]} \\ \overline{[x_2]} \\ \vdots \\ \overline{[x_n]} \end{pmatrix}$.

When we use the method of generating function, we can calculate the higher moments, though the calculation is very cumbersome. $p^{(i)}_{<i_1, i_2, \dots, i_n>}$ denotes the probability of production with which $i_1, i_2, \ldots,$ and i_n of $\sigma_1, \sigma_2, \ldots,$ and σ_n are produced from one σ_i respectively. $P_k^{(i)}(s_1, s_2, \ldots, s_n)$ denotes the generating function of the number of $\sigma_1, \sigma_2, \ldots, \sigma_n$ which are produced from one σ_i through k step derivation. Especially we write $P^{(i)}(s_1, s_2, \ldots, s_n)$ instead of $P_1^{(i)}(s_1, s_2, \ldots, s_n)$. Clearly $P_0^{(i)}(s_1, s_2, \ldots, s_n) = s_i$,

$$P^{(i)}(s_1, s_2, \ldots, s_n) = \sum_{i_1=0}^{\infty} \sum_{i_2=0}^{\infty} \cdots \sum_{i_n=0}^{\infty} p^{(i)}_{<i_1, i_2, \ldots, i_n>} s_1^{i_1} s_2^{i_2} \cdots s_n^{i_n}.$$

Let x be a string in Σ^*, $P_k^x(s_1, s_2, \ldots, s_n)$ denotes the generating function of k step starting from x.

Lemma 3 $P_k^x(s_1, s_2, \ldots, s_n)$ is given by

$$P_k^x(s_1, s_2, \ldots, s_n) = \prod_{i=1}^{n} (P_k^{(i)}(s_1, s_2, \ldots, s_n))^{\#_{\sigma_i} x}. \tag{14}$$

Proof Clear. ▶

Theorem 4 The generating functions $P_k^{(i)}(s_1, s_2, \ldots, s_n)$ $(i = 1, 2, \ldots, n,\ k > 0)$ are obtained by the following recurrence formula,

$$P_k^{(i)}(s_1, s_2, \ldots, s_n) = P^{(i)}(P_{k-1}^{(1)}(s_1, s_2, \ldots, s_n), P_{k-1}^{(2)}(s_1, s_2, \ldots, s_n), \ldots, P_{k-1}^{(n)}(s_1, s_2, \ldots, s_n)). \tag{15}$$

Proof Let $i_1, i_2, \ldots, i_n$ of $\sigma_1, \sigma_2, \ldots, \sigma_n$ be produced by the first step of derivation. From lemma 3 the generating function starting from $i_1, i_2, \ldots, i_n$ of $\sigma_1, \sigma_2, \ldots, \sigma_n$ is $\prod_{j=1}^{n} (P_{k-1}^{(j)}(s_1, s_2, \ldots, s_n))^{i_j}$. Multiply $p^{(i)}_{<i_1, i_2, \ldots, i_n>}$ and sum over all first step derivations. Thus the proof has been completed. ▶

Let $P_k^{\omega}(s_1, s_2, \ldots, s_n)$ be the generating function of k step starting from the axiom ω and $E_k^{\omega}(\sigma_i)$, $\mathrm{Var}_k^{\omega}(\sigma_i)$, $\mathrm{Cov}_k^{\omega}(\sigma_i, \sigma_j)$ denote the mean, the variance and the covariance of the numbers of σ_i and σ_j respectively. Then the following equations hold;

$$\begin{aligned} E_k^{\omega}(\sigma_i) &= \frac{\partial}{\partial s_i} P_k^{\omega}(s_1, s_2, \ldots, s_n)|_{s_1, s_2, \ldots, s_n = 1} \\ \mathrm{Var}_k^{\omega}(\sigma_i) &= \frac{\partial^2}{\partial s_i^2} P_k^{\omega}(s_1, s_2, \ldots, s_n)|_{s_1, \ldots, s_n = 1} + E_k^{\omega}(\sigma_i) - (E_k^{\omega}(\sigma_i))^2 \\ \mathrm{Cov}_k^{\omega}(\sigma_i, \sigma_j) &= \frac{\partial^2}{\partial s_i \partial s_j} P_k^{\omega}(s_1, s_2, \ldots, s_n)|_{s_1, s_2, \ldots, s_n = 1} - E_k^{\omega}(\sigma_i) E_k^{\omega}(\sigma_j). \end{aligned} \tag{16}$$

Further the mean $m(k)$ and the variance $\sigma^2(k)$ of the number of all the symbols are given by,

$$m(k)=\sum_{i=1}^{n} E_k^{\omega}(\sigma_i)$$
$$\sigma^2(k)=\sum_{i=1}^{n} \mathrm{Var}_k^{\omega}(\sigma_i)+2\sum_{i<j} \mathrm{Cov}_k^{\omega}(\sigma_i, \sigma_j). \tag{17}$$

The other questions concerning K0L-systems are much more difficult. For example, are there any $(x, p)\in \mathfrak{L}_n(G)$ with $p>\varepsilon$ for fixed $\varepsilon>0$ when $n\to\infty$?

Before the end of this section we show some examples of calculation of the mean and the variance using the method mentioned above.

Example 1 In case of our simulation, we modify the rules of Table 1 and obtain Table 2, in order to discriminate the symbols for different branching levels. That is, X is in place of the symbol V which appears at branching level 2 and never changes, Y is introduced as the symbol which may produce higher branching level than 2, instead of the symbols O and S in the rewriting rules R→OR and S→NS of Table 1. Then we can see the number of symbols of each branching level from the mean Parikh vector.

The averaged growth matrix $\overline{M}$ is shown in Table 3a, in which rows and columns are named by adequate symbols. Since the axiom is A, we only need the first row of $\overline{M}^{15}$. The first row of $\overline{M}^{15}$ obtained by multiplying $\overline{M}$ 15 times is shown in Table 3b.

Example 2 To calculate the growth of length of branches we restrict the rewriting rules as Table 4a–4c. In this case the side branches on an axis have to be neglected. Therefore we can consider each branching level independently. The generating functions of Table 4a (branching level 0) and Table 4b (branching level 1) are easily obtained and are $s_1(ps_2+q)^k$ and $s_1s_2^k$ respectively. From Table 4c, we have

$$\begin{aligned} P^{(1)}&=p_1s_2s_4+q_1s_1s_4\\ P^{(2)}&=p_2s_3s_4+q_2s_2\\ P^{(3)}&=p_3s_3s_4+q_3s_3\\ P^{(4)}&=s_4. \end{aligned} \tag{18}$$

From the last equation $P_k^{(4)}=s_4$ clearly holds for all $k>0$. From theorem 4 the following recurrence equations hold;

$$\begin{aligned} P_k^{(1)}&=p_1s_4P_{k-1}^{(2)}+q_1s_4P_{k-1}^{(1)}\\ P_k^{(2)}&=p_2s_4P_{k-1}^{(3)}+q_2P_{k-1}^{(2)}\\ P_k^{(3)}&=(p_3s_4+q_3)P_{k-1}^{(3)}. \end{aligned} \tag{19}$$

These equations are solved and the generating functions are,

	A	B	C	D	E	F	G	H	I	J	K	V	L	M	N	O	P	Q	T	U	X	R	S	W	Y	+	−	/	$/_1$	$/_2$
A	0.39	0.61	0.61																											
B	0.61	0.39		0.61																										
C					1	1																				1		1		
D					1		1																				1	1		
E					1																									
F						0.3		0.7		1																				
G							0.3		0.7		1																			
H									1	1																				
I								1		1																				
J												1	1	0.01												0.01	1	1	0.01	
K												1	0.01	1												1	0.01	1	0.01	
V												1																		
L													0.4		1		0.6													
M														0.4		1		0.6												
N															0.6						0.4	0.4	0.01			0.4	0.01	0.4		0.01
O																0.6					0.4	0.01	0.4			0.01	0.4	0.4		0.01
P															0.5		0.5		0.5											
Q																0.5		0.5		0.5										
T															0.35			0.65	0.35											
U																0.35		0.35	0.65											
X																					1									
R																						1		0.23	0.02					
S																							1	0.23	0.02					
W																								1						
Y																									1					
+																										1				
−																											1			
/																												1		
$/_1$																													1	
$/_2$																														1

Table 3a. The averaged growth matrix; $\overline{M}$.

A	0.5
B	0.5
C	0.31
D	0.31
E	8.54
F	0.44
G	0.44
H	4.04
I	3.62
J	3.75
K	4.18
V	47.58
L	5.97
M	5.67
N	22.89
O	22.93
P	5.69
Q	5.81
T	12.49
U	12.43
X	62.77
R	33.60
S	30.74
W	33.50
Y	2.91
+	61.56
−	59.37
/	118.89
$/_1$	0.48
$/_2$	1.57

Table 3b.
1st row of $\overline{M}^{15}$.

$$P_k^{(1)}=s_1r_1^k+\frac{p_1s_4\{s_2(q_2-r_3)+p_2s_3s_4\}(r_1^k-q_2^k)}{(r_1-q_2)(q_2-r_3)}-\frac{p_1p_2s_3s_4^2(r_1^k-r_3^k)}{(r_1-r_3)(q_2-r_3)}$$

$$P_k^{(2)}=s_2q_2^k+p_2s_3s_4\frac{q_2^k-r_3^k}{q_2-r_3} \tag{20}$$

$$P_k^{(3)}=s_3r_3^k \qquad \text{where } r_1=q_1s_4,\ r_3=p_3s_4+q_3.$$

Table 4a. Growth of branching level 0.

S_1; A,		S_2; C	
A → CA,	p	C → C,	1
A → A	q,	($p+q=1$)	

Table 4b. Growth of branching level 1.

S_1; H,		S_2; K
H → KH,	1	
K → K,	1	

Table 4c. Growth of branching level 2.

S_1; M, S_2; Q,		S_3; U, S_4; O	
M → OQ,	p_1	U → OU,	p_3
M → OM,	q_1	U → U,	q_3
Q → OU,	p_2	O → O,	1
Q → Q,	q_2	($p_i+q_i=1$, $i=1, 2, 3$)	

Since the starting symbol is M (corresponds to s_1), the mean and the variance are obtained from the partial derivatives of $P_k^{(1)}$. The mean length of branching level 2 at k step $m(k)$ and the variance of the branching level 2 at k step $\sigma^2(k)$ are,

$$m(k)=1+kp_3+(q_1^k-1)\frac{p_3p_1-p_1p_2+p_2p_3-p_2}{p_1p_2}+\frac{p_1(p_3-p_2)(q_1^k-q_2^k)}{p_2(q_1-q_2)}$$

$$\begin{aligned}\sigma^2(k)=&\,4-4(k+1)q_1^k+4kp_3-kp_3^2\\&+\frac{p_1(3p_2-2p_3)\{p_3(q_1-q_2)+p_2q_1\}(q_1^k-q_2^k)}{p_2^2(q_1-q_2)^2}\\&+\frac{p_1(p_3-p_2)\{(2k+1)q_1^k-q_2^k\}}{p_2(q_1-q_2)}\\&+\frac{p_1p_3-p_2(q_1-p_3)}{p_1p_2}\{(2kq_3+2)q_1^k-2\}\\&+p_1\frac{kq_1^{k+1}-(k+1)q_2q_1^k+q_2^{k+1}}{(q_1-q_2)^2}-p_1\frac{(k+2)q_1^k-2q_2^k}{q_1-q_2}\\&-\frac{2}{p_1^2p_2^2}[p_1p_2p_3(q_1-p_3)\end{aligned}$$

$$+\{p_1p_3-p_2(q_1-p_3)\}^2](q_1^k-1)-(q_1^k-1)^2\frac{\{p_1p_3-p_2(q_1-p_3)\}^2}{p_1^2p_2^2}$$
$$-\frac{p_1^2(p_3-p_2)^2(q_1^k-q_2^k)^2}{p_2^2(q_1-q_2)^2}-\frac{2kp_1p_3(p_3-p_2)}{p_2}\frac{q_1^k-q_2^k}{q_1-q_2}$$
$$+2\frac{(p_3-p_2)\{p_1p_3-p_2(q_1-p_3)\}}{p_2^2}(q_1^k-1)\frac{q_1^k-q_2^k}{q_1-q_2}. \tag{21}$$

6. Discussions

Our KOL-system simulates well the development of tree-like shapes including the feature "almost but not exactly the same". Moreover, continuous quantities such as growth speed can be discussed with discrete system.

However, we find a few problems of the simulation. First, there are limitations in length and width of branches simulated by the KOL-system. Branches of only 15 centimeters long and wide could be produced. Simulated trees much differ from the natural trees at later simulation steps. We may resolve these problems by using appropriate special rewriting rules for large branches.

Secondly, when we compare simulated trees with observed ones carefully, we see that the simulated trees are "rugged". Especially, the length of the branches at branching level 2 are very uneven. This problem is due to the interactionless L-system. When we introduce interactions among branches, we might solve this problem.

Acknowledgements

The author thanks Prof. H. Nishio, Dr. H. Okabe and other members of his laboratory for helpful discussions and advice of programming. The author is also indebted to Forest, Faculty of Agriculture, Kyoto University for permission to utilize the cypress trees.

References

1) Frijters, D., Lindenmayer, A. (1974) A model of the growth and flowering of *Aster novae-angliae* on the basis of table <1, 0> L-system, In: L Systems, edited by G. Rozenberg and A. Salomaa, Lecture Notes in Computer Science 15, pp 24–52, Springer Verlag, Heidelberg.
2) Herman, G. T., Rozenberg, G. (1975) Developmental systems and Languages. North-Holland Publ. Comp., Amsterdam.
3) Hogeweg, P., Hesper, B. (1974) A model study on biomorphological description. Pattern Recognition **6**, pp 165–179.
4) Honda, H. (1971) Description of the form of trees by the parameters of the tree-like body:

Effects of the branching angle and the branch length in the shape of the tree-like body. Journal of Theoretical Biology **31**, pp 331–338.

5) Jürgensen, H. (1975) Probabilistic L system, In: Automata, Languages, Development, edited by A. Lindenmayer and G. Rozenberg. North-holland Publ. Comp., Amsterdam.