Алгоритм Визначення Параметрів L-системи, що Моделює Структуру Геометричного Фрактального Об'єкта

Наталя Кравець
кафедра програмної інженерії
Харківський національний університет радіоелектроніки
Харків, Україна
kravets2017n@gmail.com

Algorithm for Definition of L-System's Parameters that Modeling the Geometric Fractal Object's Structure

Natalia Kravets
Department of Software Engineering
Kharkiv National University of Radio Electronics
Kharkiv, Ukraine
kravets2017n @gmail.com

***Анотація*—Представлено алгоритм визначення параметрів L-системи, яка моделює структуру геометричного фрактального об'єкта.**

***Abstract*—An algorithm for determining the parameters of an L-system that simulates the structure of a geometric fractal object is presented.**

Ключові слова—алгоритм; геометричний фрактал; L-системи

Keywords—algorithm; geometric fractal; L-system

I. Вступ

Багато природних об'єктів мають фрактальні властивості. Фрактальні об'єкти застосовуються для стиснення даних, моделювання процесів та структур у фізиці та біології. У комп'ютерній графіці фрактали використовують для генерації реалістичних зображень природних об'єктів. Метою роботи є побудова алгоритму визначення параметрів L-системи, яка моделює структуру геометричного фрактального об'єкта.

Для цього необхідно вирішити наступні завдання:

- обрати тип L-систем, які є найкращими для моделювання структури геометричних фрактальних об'єктів,
- сформулювати вимоги до вхідних даних для побудови моделі геометричного фрактального об'єкта,
- розробити алгоритм визначення параметрів моделі у вигляді L-системи.

Рекурсивна природа правил L-систем дозволяє успішно моделювати самоподібні структури, до яких належать і геометричні фрактальні об'єкти.

II. Властивості геометричних фракталів

Фракталом зазвичай називають геометричну фігуру, яка має одну чи кілька з наступних властивостей:

- нетривіальна структура, що не спрощується масштабуванням,
- самоподібність,
- дробова розмірність,

- можливість генерації за допомогою рекурсивної процедури.

Існує кілька класифікацій фракталів, за одною з них виділяють детерміновані (геометричні, алгебраїчні) та недетерміновані (стохастичні) фрактали.

A. Геометричні фрактали

Найбільш наочними є геометричні фрактали, вони будуються на основі базової фігури шляхом її дроблення і виконання афінних та логічних перетворень. Алгоритм побудови фрактального об'єкта геометричного типу може бути як ітеративним так і рекурсивним. За один крок алгоритму кожен з відрізків, що становлять базову ламану, замінюється на ламану-генератор, у відповідному масштабі. Прикладами є Сніжинка Коха, Дерево Піфагора, Дракон, Трикутник Серпінського.

B. Алгебраїчні фрактали

Алгебраїчні фрактали отримують за допомогою нелінійних процесів у n-мірних просторах. Найбільш вивчені двомірні процеси. Прикладами є Множина Мандельброта, Множина Жюліа, Фрактал Галлея, Фрактал Ньютона.

C. Стохастичні фрактали

Стохастичні фрактали будують шляхом зміни випадковим чином будь-яких параметрів ітераційного процесу створення фрактального об'єкта. Отримані об'єкти дуже схожі на природні, тому вони часто використовуються для моделювання рельєфу місцевості, рослин та ін.

III. Класифікація L-систем

Угорській біолог Аристид Лінденмайєр розробив L-системи як математичну модель опису процесу зростання рослин [1]. Зараз L-системи використовуються для моделювання складних розгалужених структур. Детерміновану контекстно-вільну L-систему визначають як кортеж:

$$G = (V, \omega, P), \tag{1}$$

де *V* - алфавіт, набір символів (змінних та констант); *ω* - аксіома або ініціатор, рядок символів із *V*, що завдає початковий стан системи; *P* - набір правил, які визначають, як змінні можуть бути замінені комбінаціями констант та інших змінних.

Під час чергової ітерації алгоритму кожна змінна замінюється набором символів у відповідності з правилами. Якщо кожній змінній відповідає лише одно правило — L-система називається детермінованою. Якщо для одної змінної з алфавіту існують кілька правил, та обрання правила для застосування залежить від контексту (оточення змінної, яка оброблюється алгоритмом), L-система зветься контекстно-залежною, якщо правило обирається з певною ймовірністю — стохастичною.

Таким чином детермінована контекстно-вільна L-система, DOL-система є найкращою для моделювання геометричних фрактальних об'єктів.

IV. Алгоритм визначення параметрів L-системи

Вхідними даними для роботи алгоритму є модель геометричного фрактала, яка представлена у вигляді ламаної лінії, завданої послідовністю вершин виду[2]:

$$\left\langle P_1(x_1, y_1), .., P_n(x_n, y_n) \right\rangle, \tag{2}$$

де $n \geq 3,\ n \in Z$.

Щоб отримати послідовність символів, аналогічну L-системі необхідно замінити кожен відрізок P_iP_{i+1} із набору $P_1P_2, P_2P_3, .., P_{n-2}P_{n-1}$ послідовністю

$$\{F\} \ldots \{X\} \ldots, \tag{3}$$

де:

$$X = \begin{cases} "-", (y_i - y_{i+1}) < (y_{i+2} - y_{i+1}), \\ "+", (y_i - y_{i+1}) > (y_{i+2} - y_{i+1}). \end{cases} \tag{4}$$

F - інтервал, що відповідає відрізку ламаної, позначається літерою латинського алфавіту, інтервали із рівною довжиною позначаються однаковими літерами; «+» або «–» відповідають куту повороту ламаної проти годинникової стрілки або за годинниковою стрілкою; розмір кута обчислюється як кут α_i між векторами $\overline{P_iP_{i+1}}$ та $\overline{P_{i+1}P_{i+2}}$; кількість повторень «+» або «–» дорівнює α_i/α_{min} , де $\alpha_1, \alpha_2, .., \alpha_{n-2}$ - кути між відрізками ламаної $P_1P_2..P_n$.

Отримана послідовність *S* (3), що мітить літери та символи «+» , «–» є вхідними даними для роботи алгоритму.

Алфавітом майбутньої L-системи буде набір символів, що входять до послідовності *S*. «+» та «–» — константи, літери латиниці — змінні.

Кількість шуканих правил дорівнює кількості змінних алфавіту V_S.

Треба визначити правила (ліва частина — змінна, права — вираз) та аксіому.

1) *Виділення підрядків:* Виділити всі різні підрядки довжиною більше 1 та менше $Length_S/2$, що починаються та закінчуються літерою.

2) *Пошук набору правил:* Формуємо з виділених на попередньому кроці підрядків всі можливі набори з V_S правил, з них відбираємо ті, в яких хоча б одне з правил починається з першого символу послідовності *S* та хоча б одне з правил закінчується останнім символом послідовності *S*.

3) *Перевірка наборів правил:* Із отриманих на кроці 2 наборів правил відбираємо той, що відповідає обом наступним вимогам:

a) Повнота: Із обраного набору правил-підрядків розташованих у довільному порядку можна сформувати послідовність *S*.

b) Можливість рекурсивного використання: Для обраного набору правил-підрядків можна підібрати набір змінних з алфавіту *V*, такий що рядок, отриманий у результаті заміни можна знову зібрати з правил як у вимозі a).

4) *Пошук аксіоми:* Проводити заміну у відповідності до визначеної системи правил доки не отримаємо рядок, для якого це виконати не можливо, цей рядок і буде аксіомою.

Алгоритм не є універсальним, позитивний результат можна отримати лише для геометричних фрактальних об'єктів, що відносяться до не менш ніж другого покоління.

Розглянемо використання розробленого алгоритму для побудови L-системи кривої Коха рис. 1.

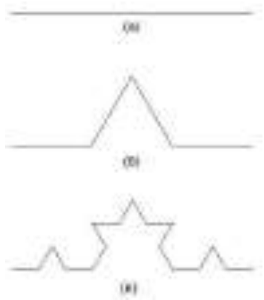

Рис. 4. Крива Коха, (в) - друге покоління

Для другого покоління кривої Коха маємо алфавіт: *F* - одиничний відрізок ламаної; «+» - поворот за годинниковою стрілкою на 60°; «–» - поворот ламаної проти годинникової стрілки на 60°. Кількість шуканих правил дорівнюватиме 1. Послідовність *S* має вигляд (5):

$$F-F++F-F-F-F++F-F++F-F++F-F-\\ F-F++F-F. \tag{5}$$

Виконуючи кроки алгоритму маємо:

1) *Виділення підрядків:* Виділяємо всі різні підрядки довжиною більше 1 та менше *36/2=18*, що починаються та закінчуються літерою. Отримуємо 23 правила (рис. 2).

1) F–F
2) F++F
3) F–F–F
4) F–F++F
5) F++F–F
6) F–F–F–F
7) F–F++F–F
8) F++F–F–F
9) F–F–F++F
10) F–F ++F–F–F
11) F++F–F–F–F
12) F–F–F–F++F
13) F–F–F ++ F–F
14) F–F++ F–F++ F–F
15) F–F–F–F++ F–F
16) F++ F–F–F–F++F
17) F–F–F++F–F++F
18) F–F++F–F–F–F++F
19) F++F–F–F–F++F–F
20) F–F–F–F++F–F++F
21) F–F–F ++F–F++F–F
22) F–F++F–F–F–F++F–F
23) F–F–F–F ++F–F++F–F

Рис. 5. Набори правил

2) *Пошук набору правил:* Формуємо 23 набори лише з одного правила.

3) *Перевірка наборів правил:* Із отриманих на кроці 2 23-х наборів правил відбираємо той, що відповідає вимогам повноти та можливості рекурсивного використання:

a) Повнота: Послідовність *S* можна сформувати із наступних правил-підрядків: p_1,p_7,p_{22}.

b) Можливість рекурсивного використання: Для набору p_1:*F–F* в результаті згортання послідовності *S* отримуємо послідовність: *F++F–F++F++F++F–F++F*, яка не підлягає подальшому згортанню з використанням набору p_1. Для набору p_7:*F–F++F–F* в результаті згортання послідовності *S* отримуємо послідовність: *F–F++F–F*, яка у результаті подальшого згортання з використанням набору p_7 дає послідовність *F*. Для набору p_{22}:*F–F++F–F–F–F++F–F* в результаті згортання послідовності *S* отримуємо послідовність: *F++F*, яка не підлягає подальшому згортанню з використанням набору p_{22}. Таким чином за результатами перевірки наборів правил отримуємо набір, що містить правило *F*$\rightarrow$*F–F++F–F*.

4) *Пошук аксіоми:* Після проведення заміни у відповідності до визначеного правила отримаємо рядок *F*, який і буде аксіомою.

Результатом роботи алгоритму є детермінована контекстно-вільна L-система:

- Алфавіт *V*: змінні: *F*; константи: +, –;
- Аксіома ω: *F* ;
- Правила *P*: p_1: *F*$\rightarrow$*F–F++F–F*.

Для даного алгоритму мається на увазі, що всі змінні і константи вхідної послідовності мають геометричну інтерпретацію. Тобто L-систему для фракталу Дракон:

- Алфавіт *V*: змінні: *X*, *Y*; константи: *F* - одиничний відрізок ламаної; «+» - поворот за годинниковою стрілкою на 90°; «–» - поворот ламаної проти годинникової стрілки на 90°;
- Аксіома ω: *FX* ;
- Правила *P*: p_1: *X*$\rightarrow$*X+YF+*, p_2: *Y*$\rightarrow$*–FX–Y*.

за допомогою цього алгоритму побудувати неможна.

Складність наведеного алгоритму приблизно дорівнює $O(n^2)$ та залежить не тільки від довжини вхідної послідовності, а й від кількості змінних в алфавіті майбутньої системи. Більш складними завданнями є ідентифікація та побудова контекстно-залежної або стохастичної L-системи за вхідною послідовністю.

ЛІТЕРАТУРА REFERENCES

[1] P. Prusinkiewicz, A. Lindenmayer, The algorithmic beauty of plants, – Springer Science & Business Media, 2012.

[2] Н. С. Кравец, Использование метода секвенциального анализа для исследования структуры фрактальных объектов// Радиоэлектроника и информатика, №4 декабрь 2015 С. 61-64.