Automatic Modeling of Plant Development by Lindenmayer Systems

Martin Čermák, Jiří Koutný and Alexander Meduna

Deparment of Information Systems
Faculty of Informatin Technology
Brno University of Technology, Faculty of Information Technology
Božetěchova 2, Brno 612 00, Czech Republic

Advanced Topics of Theoretical Computer Science

FRVŠ MŠMT FR2581/2010/G1

Branching Structures

Introduction

- Turtle interprets character string as a sequence of line segments.
- Output is just a single line.
- Plant kingdom is dominated by branching structures.
- Mathematical description of tree–like shapes and method for generating them are needed for modeling purposes.
- An axial tree complements the graph-theoretic notion of a rooted tree with the botanically motivated notion of branch axis.

Part I

Axial Trees

Basic Definitions

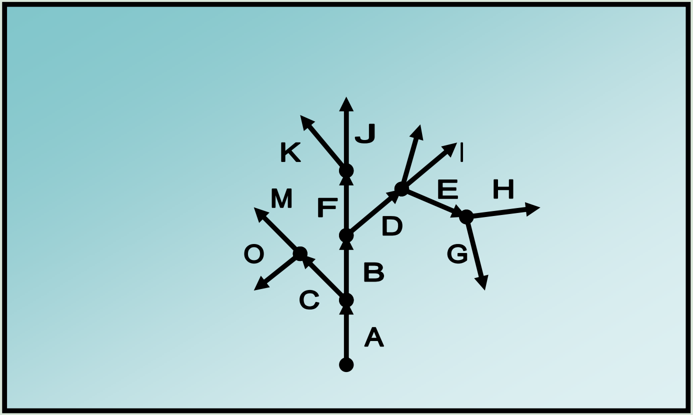
Rooted Tree

A rooted tree is a tree with edges that are labeled and directed.

Rooted Tree

A rooted tree is a tree with edges that are labeled and directed.

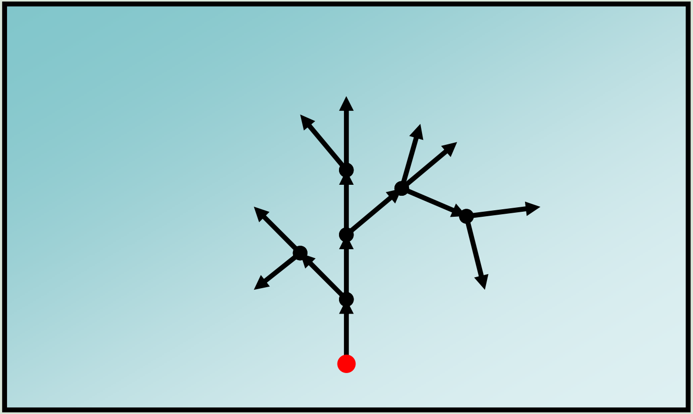
Example



Root

A root (base) is a distinguished node.

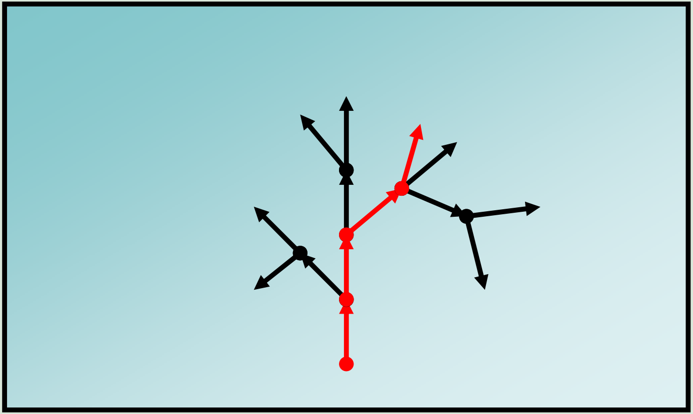
Example



Branch Segments

Edge sequences (paths) from the root to the terminal nodes.

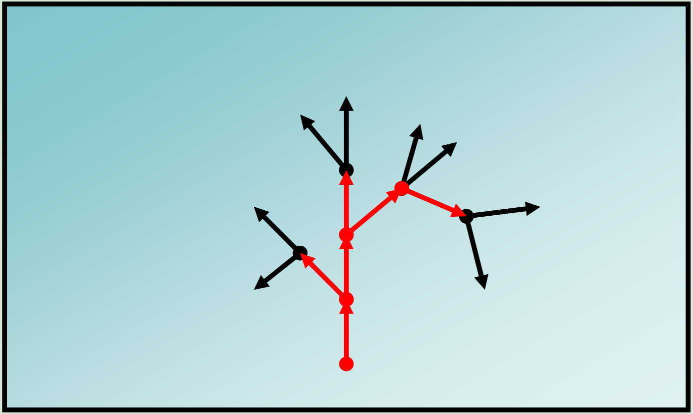
Example



Internode

A segment followed by at least one more segment in some path.

Example

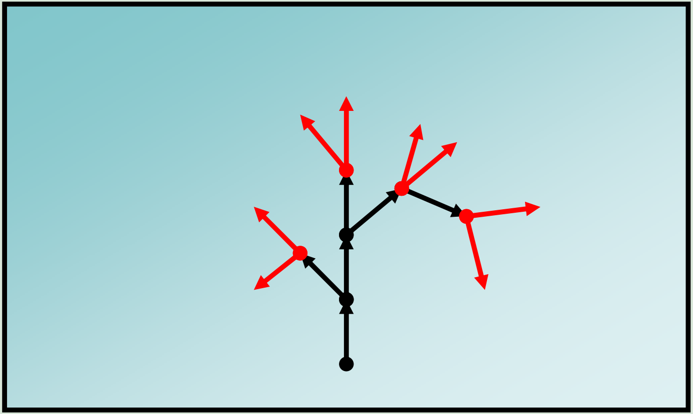


Basic Definitions

Apex

An apex is a terminal segment (with no succeding edges).

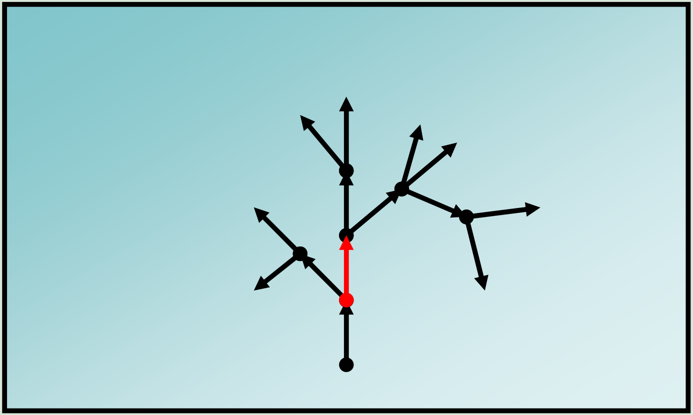
Example



Axial Tree

An axial tree is a root tree, where at each of its nodes, at most one outgoing straight segment is distinguished.

Example

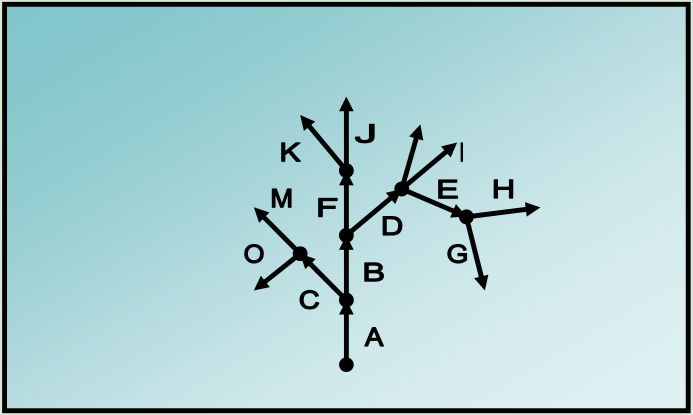


Axis

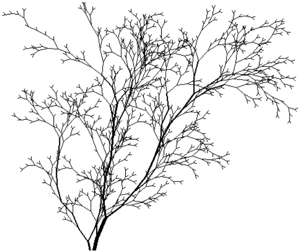
Sequence of segments where:

- the first segment in the sequence originates at the root of the tree or as a lateral segment at some node,
- each subsequent segment is a straight segment, and
- the last segment is not followed by any straight segment in the tree.

Example



- Axial trees are purely topological objects.
- Geometric connotation should be viewed at this point as an intuitive link between the graph–theoretic formalism and real plant structures.

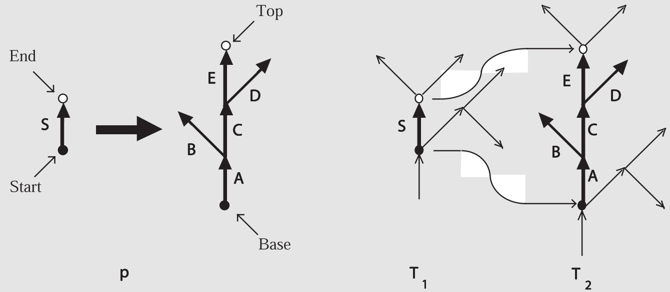


Part II

Tree 0L–Systems

Introduction

- Rewriting mechanism can operates directly on axial trees.
- A rewriting rule replaces a predecessor edge by a successor axial tree.
- The starting node of the predecessor is identified with the successors's base.
- The ending node is identified with the successor's top.



0L–System

A Tree 0L–System is a triple

$$G = (T, P, w)$$

where:

- T is an set of edge labels,
- P is a set of productions,
- w is the initial tree with labels from T.

Derivation

Let $G = (T, P, w)$ be a tree 0L–system and let R_1 and R_2 be two axial trees. R_2 is directly derived from R_1, $R_1 \Rightarrow R_2$, if R_2 is obtained from R_1 by simultaneously replacing each edge in R_1 by its successor according to the production set P.

An axial tree of 0L–system

Let $G = (T, P, w)$ be a tree 0L–system and let $R_0, R_1, \ldots, R_n$ be $n + 1$ axial trees for $n \geq 0$. An axial tree R is generated by G in a derivation of lenght n if there exists a sequence of defivation
$R_0 \Rightarrow R_1 \Rightarrow \ldots \Rightarrow R_n$, where $R_0 = w$ and $R_n = R$.

Bracketed 0L–System

A Bracketed 0L–System is an 0L–system which generates some strings for the turtle graphics, but some parts of these strings can be in parentheses.

interpretation

Consider parentheses (and) and any word $w = a_1 a_2 \ldots a_n$ over $\{F, +, -, (,)\}^*$.

1. set position *pos* and orientation *or* of the turtle
2. set $i = 1$
3. if a_i is (then push *pos* and *or* on the stack
4. else if a_i is) then pop *or* and *pos* and set position and orientation of the turtle
5. else work by the standard way
6. if $i = n$ then finish
7. else $i = i + 1$ and go to step 3

Example

Consider 0L–system given by following rules:

- axiom $w = F$,
- $F \to F(+F)F(-F)F$

with angle $\delta = 25.5^\circ$ for turtle interpreting.

Example

Consider 0L–system given by following rules:

- axiom $w = F$,
- $F \to F(+F)F(-F)F$

with angle $\delta = 25.5^\circ$ for turtle interpreting. After 5 iterations:

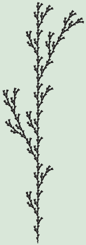


Fig: 0L System from [Chap. 1.6.2 in The Algorithmic Beauty of Plants]

Example

Consider 0L–system given by following rules:

- axiom $w = F$,
- $F \rightarrow F(+F)F(-F)(F)$

with angle $\delta = 20^\circ$ for turtle interpreting.

Example

Consider 0L–system given by following rules:

- axiom $w = F$,
- $F \rightarrow F(+F)F(-F)(F)$

with angle $\delta = 20^\circ$ for turtle interpreting. After 5 iterations:

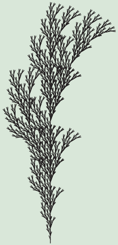


Fig: 0L System from [Chap. 1.6.2 in The Algorithmic Beauty of Plants]

Example

Consider 0L–system given by following rules:

- axiom $w = F$,
- $F \rightarrow FF - (-F + F + F) + (+F - F - F)$

with angle $\delta = 22.5^\circ$ for turtle interpreting.

Example

Consider 0L–system given by following rules:

- axiom $w = F$,
- $F \rightarrow FF-(-F+F+F)+(+F-F-F)$

with angle $\delta = 22.5^{\circ}$ for turtle interpreting. After 4 iterations:

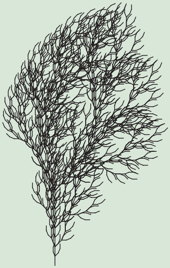


Fig: 0L System from [Chap. 1.6.2 in The Algorithmic Beauty of Plants]

Example

Consider 0L–system given by following rules:

- axiom $w = X$,
- $X \to F(+X)F(-X) + X$
- $F \to FF$

with angle $\delta = 20^\circ$ for turtle interpreting.

Example

Consider 0L–system given by following rules:

- axiom $w = X$,
- $X \to F(+X)F(-X) + X$
- $F \to FF$

with angle $\delta = 20^\circ$ for turtle interpreting. After 7 iterations:

Fig: 0L System from [Chap. 1.6.2 in The Algorithmic Beauty of Plants]

Example

Consider 0L–system given by following rules:

- axiom $w = X$,
- $X \rightarrow F(+X)(-X)FX$
- $F \rightarrow FF$

with angle $\delta = 25.7^\circ$ for turtle interpreting.

Example

Consider 0L–system given by following rules:

- axiom $w = X$,
- $X \rightarrow F(+X)(-X)FX$
- $F \rightarrow FF$

with angle $\delta = 25.7^\circ$ for turtle interpreting. After 7 iterations:

Fig: 0L System from [Chap. 1.6.2 in The Algorithmic Beauty of Plants]

Example

Consider 0L–system given by following rules:

- axiom $w = X$,
- $X \rightarrow F - ((X) + X) + F + (+FX) - X$
- $F \rightarrow FF$

with angle $\delta = 22.5^\circ$ for turtle interpreting.

Bracketed 0L–Systems

Example

Consider 0L–system given by following rules:

- axiom $w = X$,
- $X \to F - ((X) + X) + F + (+FX) - X$
- $F \to FF$

with angle $\delta = 22.5^\circ$ for turtle interpreting. After 5 iterations:

Fig: 0L System from [Chap. 1.6.2 in The Algorithmic Beauty of Plants]

Example

Consider 0L–system given by following rules:

- axiom $w = A$,
- $A \rightarrow (\&FL; A)/////'(\&FL; A)///////'(\&FL; A)$,
- $F \rightarrow SF$
- $S \rightarrow FL$
- $L \rightarrow ('''\wedge\wedge\{-f+f+f-|-f+f+f\})$

with angle $\delta = 22.5^\circ$ for turtle interpreting. After 7 iterations:

Fig: 0L System from [Chap. 1.6.2 in The Algorithmic Beauty of Plants]

Bracketed 0L–Systems

- 0L–system used by this way, generate only one object.
- An attempt to combine them in the same picture would produce a striking, artificial regularity.
- It is necessary to introduce specimen–to–specimen variations that will preserve the general aspects of a plant but will modify its details.
- Solution:
  - randomizing turtle interpretation,
  - randomizing L-system,
  - randomizing turtle interpretation and L-system

Part III

Stochastic 0L–systems

Stochastic 0L–System

A stochastic 0L–System is a quadruplet

$$G = (T, P, w, \pi)$$

where:

- $G = (T, P, w)$ is an 0L–System,
- $\pi : P \rightarrow (0, 1]$ is a probability distribution and maps the set of productions into the set of probabilities.

- It is assumed that for any letter $a \in T$, the sum of probabilitions with the predecessor a is equal to 1.
- By this way, different productions with the same predecessor can be applied to various occurences of the same latter in one derivation step.

Stochastic 0L–Systems

Example

Consider 0L–system given by following:

w F

p_1 $F \stackrel{.33}{\rightarrow} F(+F)F(-F)F$

p_2 $F \stackrel{.33}{\rightarrow} F(+F)F$

p_3 $F \stackrel{.34}{\rightarrow} F(-F)F$

Fig: 0L System from [Chap. 1.7 in The Algorithmic Beauty of Plants]

Stochastic 0L–Systems

Fig: 0L System from [Chap. 1.7 in The Algorithmic Beauty of Plants]

Part IV

Context–sensitive L–systems

2L–System

A 2L–System is a triplet $G = (T, P, w)$, where:

- T is an alphabet,
- w is the start string (axiom) and
- P is a set of rules of the form $a_l < a > a_r \rightarrow x$ or $a \rightarrow x$ with $a_l, a, a_r \in T$ and $x \in T^*$.

Derivation Step

- Rules of the form $a_l < a > a_r \rightarrow x$ can be used only if the first symbol on the letf is a_l and a_r is the first symbol on the left,
- If there is a collision between rules $a \rightarrow x$ and $a_l < a > a_r \rightarrow y$, the context one is used.

1L–System

Contains rules of the form:

- $a_l < a \rightarrow x$ or $a > a_r \rightarrow x$ or $a \rightarrow x$ with $a_l, a, a_r \in T$ and $x \in T^*$.

Problem with tree

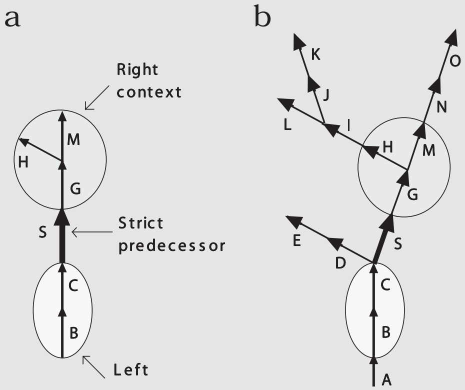


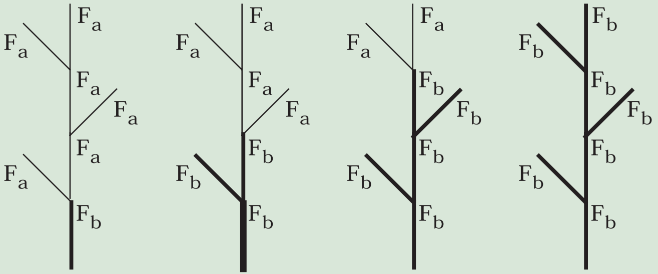
Fig: [Chap. 1.8 in The Algorithmic Beauty of Plants]

Example

Consider 1L–system G given by following:

- $w = F_b(+F_a)F_a(-F_a)F_a(+F_a)F_a$
- $p_1 : F_b < F_a \rightarrow F_b$

Consider $+$ and $-$ are ignored in context.



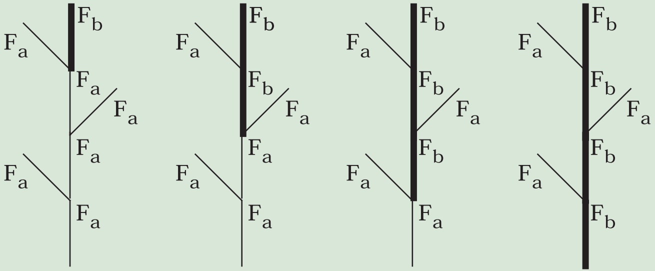
L System from [Chap. 1.8 in The Algorithmic Beauty of Plants]

Example

Consider 1L–system G given by following:

- $w = F_b(+F_a)F_a(-F_a)F_a(+F_a)F_a$
- $p_1 : F_a > F_b \rightarrow F_b$

Consider $+$ and $-$ are ignored in context.



L System from [Chap. 1.8 in The Algorithmic Beauty of Plants]

Example

Consider 1L–system G given by following:

- ignere: $F, +$ and $-$
- $w = F1F1F1$
- $0 < 0 > 0 \rightarrow 0$
- $0 < 0 > 1 \rightarrow 1(+F1F1)$
- $0 < 1 > 0 \rightarrow 1$
- $0 < 1 > 1 \rightarrow 1$
- $1 < 0 > 0 \rightarrow 0$
- $1 < 0 > 1 \rightarrow 1F1$
- $1 < 1 > 0 \rightarrow 0$
- $1 < 1 > 1 \rightarrow 0$
- $* < + > * \rightarrow -$
- $* < - > * \rightarrow +$

Example

Consider $\delta = 20^\circ$ and $n = 30$

Fig: L System from [Chap. 1.8 in The Algorithmic Beauty of Plants]

Example

Consider 1L–system G given by following:

- ignere: $F, +$ and $-$
- $w = F1F1F1$
- $0 < 0 > 0 \to 0$
- $0 < 0 > 1 \to 1(-F1F1)$
- $0 < 1 > 0 \to 1$
- $0 < 1 > 1 \to 1$
- $1 < 0 > 0 \to 0$
- $1 < 0 > 1 \to 1F1$
- $1 < 1 > 0 \to 0$
- $1 < 1 > 1 \to 0$
- $* < + > * \to -$
- $* < - > * \to +$

Example

Consider $\delta = 20^\circ$ and $n = 30$.

Fig: L System from [Chap. 1.8 in The Algorithmic Beauty of Plants]

Part V

Parametric L–systems

Parametric 0L–System

A Parametric 0L–System is a quadruplet

$$G = (T, P, w, \Sigma)$$

where:

- T is an alphabet,
- Σ is the set of formal parameters,
- $w \in (T \times R^*)+$ is a nonempty start parametric string (axiom) and
- $P \subset (T \times \Sigma^*) \times C(\Sigma) \times (T \times E(\Sigma))^*$ is a finit set of productions.

Example

- $w = B(2)A(4,4)$
- $A(x,y) : \; x \leq 3 \; \rightarrow \; A(x*2, x+2)$
- $A(x,y) : \; x > 3 \; \rightarrow \; B(x)A(x/y, 0)$
- $B(x) : \; x < 1 \; \rightarrow \; C$
- $B(x) : \; x \geq 1 \; \rightarrow \; B(x-1)$

Example

- $R = 1.456$
- $w = A$
- $A \rightarrow F(1)(+A)(-A)$
- $F(s) \rightarrow F(s * R)$

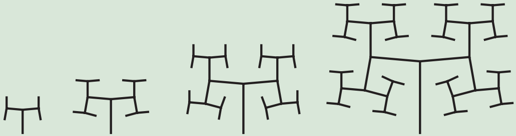


Fig: L System from [Chap. 1.10 in The Algorithmic Beauty of Plants]

References

- Meduna Alexander.
  Automata and Languages: Theory and Applications, London, Springer, 2000.
  Spring, 2000.
- Meduna Alexander, Švec Martin.
  Grammars with Context Conditions and Their Applications.
  Wiley, 2005.
- Przemyslaw Prusinkiewicz, Aristid Lindenmayer
  The Algorithmic Beauty of Plants.
  From URL:
  `http://algorithmicbotany.org/papers/#webdocs`
- Rozenberg Grzegorz, Salomaa Arto.
  Lindenmayer systems.
  Springer-Verlag, 1992.

Thank you for your attention!

End