Interactive Modeling and Authoring of Climbing Plants

Torsten Hadrich et al.
Eurographics 2017

Presented by Qi-Meng Zhang
2017. 04. 20

Computer Graphics @ Korea University
Abstract

• Interactive modeling of developmental climbing plants with an emphasis on efficient control and plausible physics response

• Plant is represented by a set of connected anisotropic particles
  • Each particle stores biological and physical attributes that drive growth and plant adaptation to the environment
    • Light sensitivity, wind interaction, physical obstacles
1 Introduction
1 Introduction

• Dynamic plant model
  ▪ React to the presence of other plants, to varying lighting conditions, and to the scene itself

• A number of methods have been proposed for realizing adaptive plants
  ▪ L-system
    • [PRUSINKIEWICZ P, LINDENMAYER A./“The algorithmic beauty of plants” Springer-Verlag New York 1990]
  ▪ Inverse procedural model
    • [STAVA O et al./“Inverse procedural modelling of trees” Comp.Graph.Forum.2014]
  ▪ Competition for resources
    • [RUNIONS A et al./“Modeling trees with a space colonization algorithm” Eurographics 2007]
  ▪ Simulated adaptation
    • [PIRK S et al./“Plastic trees: interactive self adapting botanical tree models” ACM Trans. Graph. 31, 4 2012]
1 Introduction

• The difficulty with climbing plants
  ▪ Need consider adaptation to the geometry of the supporting object

• Previous approaches simulated climbing plants
  ▪ Environmentally sensitive automata
    • [ARVO J et al./“Modeling plants with environment-sensitive automata” Ausgraph 1988]
  ▪ Competing particles in voxel space
    • [GREENE N./“Voxel space automata: Modeling with stochastic growth processes in voxel space” SIGGRAPH 1989]
  ▪ Represent tendrils as mass-springs
    • [WONG S.-K and CHEN K.-C./“A procedural approach to modelling virtual climbing plants with tendrils” Computer Graphics Forum 2015]

• Control, is a major open problem in plants modeling
  ▪ Most of the existing algorithm focus on standing tree
  ▪ Control by setting input parameters and the initial location of trees
1 Introduction

• Contribution

  ▪ We implemented an interactive method that allows for coherent modeling of climbing plants in changing environments and along the entire developmental process of the plant

  ▪ We model climbing plants as dynamic systems that support biologically- and physically-plausible behavior; plants remain flexible and animation-ready during the modeling session

  ▪ We couple plants with wind simulations and model advanced physical effects
    • Bending and breaking of branches

  ▪ We introduce a number of editing operations
    • Plant seeding, dynamic branch placement, removal, and sketching of attractors on support geometry
2 Related Work
2 Related Work

- Interactive control by positioning attraction

  - Simulate climbing plants by space colonization
  


- Require the entire plant or a set of parameters

  - Simulate plants by either Inverse procedural modeling

  STAVA O et al./ "Inverse procedural modelling of trees" Comp. Graph. Forum. 2014

  - Simulate plants by simulating the effect of wind on tree development

2 Related Work

• Interactive methods focus on user control

  • Example-based sketching system

  OKABE M et al. "Interactive design of botanical trees using freehand sketches and example-based editing" SIGGRAPH Courses. 2007

  Steven Longay et al. "TreeSketch: Interactive procedural modeling of trees on a tablet" SBIM. 2012

• Climbing plants

  • Used L-system to model climbing plants and react to gravity and sunlight


  • Generate climbing plants with a focus on procedural modeling and the behavior of tendrils that grow around objects

2 Related Work

- Particle system

  - Integrate spherical particles that approximate a tree structure within a fluid solver to simulate the interaction between trees and wind


  - Simulate deformable object by a meshless approach

    MÜLLER M et al. "Meshless deformations based on shape matching" ACM Trans. On Graph. 2005

  - Simulate deformable solids using SPH (smoothed particle hydrodynamics)


  - Extend shape matching by incorporating oriented particles

3 Overview

- Update branching structure
  - Particle simulation (Plant dynamic)
    - Geometry-based method
  - Plant growth
    - Directed random walk
      - influenced by environmental conditions
4 Climbing Plants
4 Climbing plants

- **4.1 Plant dynamic**
  - Modifies the existing plant geometry

- **4.2 Plant growth**
  - Add new plant geometry

- **4.3 Species and Material Properties**
4 Climbing plants

- Plant module

- Plant skeleton and Branch thickening

Figure 3
4 Climbing plants

4.1 Plant Dynamics

• Modifies the existing plant geometry
  ▪ User interactions and external forces

• Particle-based representation
  ▪ Particle carry quantities for their current state and rest state
    • Position and orientation, main axis (plant skeleton), velocity, angular velocity

  ▪ Particle attributes
    • Update in each time step

  ▪ Particle group
    • Include current particle, parent particle, successors
4 Climbing plants

4.1 Plant Dynamics

• Our particle-based plant representation is based on the shape matching approach

• Why need using the shape matching approach?
  ▪ The existing plant shape modified by external forces
    - Like pulling the branches to a different location
  ▪ The shape matching algorithm restores the initial plant shape

![Diagram showing the effect of pulling and shape matching on plant representation](image-url)
Shape matching algorithm

• “Meshless deformations based on shape matching”
  [MÜLLER M et al./ ACM Trans. 2014]

- Original shape
- Deformed shape
- Matched shape

- \( x_i^0 \) : initial position
- \( x_i \) : actual position
- \( g_i \) : goal position
4.1 Plant Dynamics

Particle Positions Update

- Particle positions
  - Current position $X$
  - Predicted position $X_p$
  - Target position $X_t$
  - Goal position $X_g$

Figure 4 (a) and (b)
4.1 Plant Dynamics

Predicted Position and Orientation

- Predicted position $X_p$

$$x_p = x + v \Delta t + \frac{(a_g + a_e) \Delta t^2}{2}, \quad (1)$$

$$v = (x_p - x)/\Delta t \quad (17)$$

$$x = x_p \quad (18)$$

- $X$: particle position
- $V$: particle velocity
- $\Delta t$: simulation step
- $a_g$: gravitational acceleration
- $a_e$: external acceleration (caused by fluid particles)
4.1 Plant Dynamics

Predicted Position and Orientation

- Predicted orientation $q_p$

$$q_p = \left[ \hat{\omega} \sin\left(\frac{|\omega| \Delta t}{2}\right), \cos\left(\frac{|\omega| \Delta t}{2}\right) \right] q, \quad (16)$$

- $\omega$: angular velocity of particle
- $q$: current particle orientation

$$\omega = \text{axis}(q_p q^{-1}) \cdot \text{angle}(q_p q^{-1}) / \Delta t \quad (19)$$

$$q = q_p \quad (20)$$
4.1 Plant Dynamics

Optimal Rotation

• Optimal rotation **R**
  - Minimizes the RMSD (root mean squared deviation) between two paired sets of points
  - Matches the rest state to the current state of each particle group

\[
A = RS \quad \text{(polar decomposition)}
\]

\[
S = \sqrt{A^T A}
\]

\[
A = \sum_i \left(A_i + m_i x_i \bar{x}_i^T - m_i c_i \bar{c}_i^T\right).
\] (14)

• **A**: total moment matrix
• **S**: symmetric part
• **m_i**: particle mass
• **x_i** and **\bar{x}_i**: current and rest particle positions
• **c_i** and **\bar{c}_i**: current and rest centers of mass per particle group

\[
R = AS^{-1}
\]
4.1 Plant Dynamics

Optimal Rotation

- The moment matrix depend on mass \( m \)

\[
m = V \rho = \frac{4\pi abc \rho}{3}, \tag{2}
\]

- \( V \): volume
- \( \rho \): density
- \( a, b, c \): the axes of the ellipsoid

\[
A_i = \frac{1}{5} m_i \begin{bmatrix}
a^2 & 0 & 0 \\
0 & b^2 & 0 \\
0 & 0 & c^2
\end{bmatrix} R. \tag{15}
\]
4.1 Plant Dynamics

Target and Goal Position

- Target position

\[
x_t = R(\bar{x} - \bar{c}) + c,
\]

(3)

- Goal position

\[
x_g = \sum_i w_i x_i^i / W,
\]

(4)

- \( w \): individual weight

- When particle attach to objects

\[
x_p' = x_p + \phi(x_{anchor} - x_p),
\]

(5)

\[
0 < \phi < 1
\]
4.2 Plant Growth

- Add new plant geometry
  - By using two way
    - Extending existing branches
    - Adding new lateral branches
  - Reacts to environmental conditions
4.2 Plant Growth

• Within each time step all particles at the end of the plant’s shoots increase their size until a maximal size is reached.

• Growth rate depends on the amount of light at the particle position that can additionally be controlled by the user.

• The two contributions of surface adaption and phototropism are integrated into a new growth position and orientation.
4.2 Plant Growth

Surface Adaption

- Plant particle approaches an object
  - Plant orients itself parallel to the surface

  **Axis:**
  \[ \mathbf{a}_a = \mathbf{\hat{v}}_s \times \mathbf{\hat{v}}_f \]

  **Rotational angle:**
  \[ \alpha_a = (\mathbf{\hat{v}}_s \cdot \mathbf{\hat{v}}_f) \tau \Delta t, \quad (6) \]

  - \( \mathbf{v}_s \): vector pointing to the closest surface
  - \( \mathbf{v}_f \): current forward vector
  - \( 0 \leq \tau \leq 1 \): controls the surface adaption strength
  - Defined by user

Figure 4 (c)
4.2 Plant Growth

Phototropism

- Plant response to light
  - Orients plant organs towards the light direction
  - Help the apices reach areas with more intensive illumination

Axis:

\[ \mathbf{a}_p = \mathbf{v}_l \times \mathbf{v}_f \]

Rotational angle:

\[ \alpha_p = (1 - O)\eta\Delta t, \quad (7) \]

- \( \mathbf{v}_l \): vector to the light source
- \( O \): light occlusion at the particle location
- \( \eta \): controls the phototropism response strength

Figure 4 (d)
4.2 Plant Growth

Growth Integration

- Accumulated rotation matrix
  \[ R_g = R(a_a, \alpha_a) R(a_p, \alpha_p), \]  
  \[ R(a, \alpha) \text{ returns a rotation matrix} \]  

- Update particle orientation
  \[ R = RR_g \]
  \[ \bar{R} = \bar{R}R^{-1}R_g, \]
  \[ R \text{ and } \bar{R} : \text{ current and rest orientations} \]
4.2 Plant Growth

Growth Integration

- Update particle position
  - In its current state and rest state
    \[
    \mathbf{x} = \mathbf{x}_h + \mathbf{\bar{R}} \mathbf{R} \mathbf{u}_f \\
    \mathbf{\bar{x}} = \mathbf{\bar{x}}_h + \mathbf{\bar{R}} \mathbf{u}_f,
    \]
  - \( \mathbf{x}_h \) and \( \mathbf{\bar{x}}_h \) : head position
  - \( \mathbf{u}_f \) : forward vector, \( \mathbf{u}_f = \begin{bmatrix} 0, & 1, & 0 \end{bmatrix}^T \)
4.3 Species and Material Properties

Branches and Leaves

- **Branches**
  - Branching probability
    - Branching variance: [0,1]
      - Direction of lateral branches
        - Orientation MAX=90 degrees
  - Thickness of new branch
    - $t_c = t_p \cdot f$
      - $t_c$ : thickness of child branch
      - $t_p$ : thickness of parent branch
      - $f$ : falloff parameter

- **Leaves**
  - Model each individual leaf with a single particle
4.3 Species and Material Properties

Stiffness and Branch Breaking

- **Stiffness**
  \[
  \mathbf{x}_g' = \mathbf{x}_p + s(\mathbf{x}_g - \mathbf{x}_p)
  \]
  - \(s\) : stiffness parameter range from \([0,1]\)
    - Controls the elasticity of plant
    - \(s = \frac{t_l}{t_m}\) : \(s\) range \([0,1]\)
      - \(t_l\) : life time
      - \(t_m\) : time of particle reaches its maximum stiffness

- **Breaking**
  - Occur from gravity or user interaction
5 Authoring
5 Authoring

5.1 Dynamic Editing

- Interactive editing operations
  - Seeding plant anywhere in the scene
  - Add a new shoot from existing branch
  - Grabbing the branches
  - Coupled with fluid dynamics
  - Cutting branches

![Image](image.png)

Figure 7

(a) (b) (c) (d) (e)
5 Authoring

5.1 Dynamic Editing

• Paint regions on obstacles
  ▪ Attract or repulse the plant’s growth
  ▪ New sketch triggers new branch

Figure 8
5 Authoring

5.2 Collision Response

• Collision of own organs
  ▪ Particle gets closer to the others than their radius \((r_n + R_n)\)

  ▪ Ellipsoid-Ellipsoid Collision
    • “Solid simulation with oriented particles”
      [ MÜLLER M., CHENTANEZ N./ ACM Trans. Graph 2011 ]
  ▪ Compute the contact point of two particles and displace them along their normal until they no longer intersect
    • \(d\) is the scalar that tells us how far to shift ellipsoid
    • \(x\) is the contact point
5 Authoring

5.2 Collision Response

- Collision of other plants, obstacles
  - Assign each shape a signed distance field (SDF)
    - Compute the distance of longest axis of a particle and the surface stored in the SDF
    - If the distance is smaller than the length of the longest axis
    - Move the particle

SDF:

The sign of ‘0’ represent surface

Figure 9
5 Authoring

5.3 Two-way Fluid Coupling

- Couple plants with a fluid simulation
  - Wind field is simulated by Smoothed Particle Hydrodynamics (SPH)

\[
a_i = \frac{dv_i}{dt} = (-\nabla p + \mu \nabla^2 v) / \rho, \tag{11}
\]

- \(-\nabla p\): pressure
- \(\mu \nabla^2 v\): viscosity
- Wood and air density: \(0.3 - 1.0 \cdot 10^3 \text{ kg/m}^3\)
  \(1.3 - 1.4 \text{ kg/m}^3\)

\[
x_p = x + v \Delta t + \frac{(a_g + a_e) \Delta t^2}{2}, \tag{1}
\]

Figure 4 (f)
5 Authoring

5.3 Two-way Fluid Coupling

- Fluid quantities $A(x)$, at a certain location $x$ are computed as a weighted sum of neighboring particles $j$,

$$A(x) = \sum_{j=1}^{N} V_j A_j \, W(\eta), \quad (12)$$

- $V_j$ : volume
- $W$ : smoothing kernel
- $\eta$ : normalized position vector ($\eta = (x - x_j)/h$)
  - Ellipsoidal particles

$$\eta = G(x - x_j), \quad (13)$$
- $G$ : linear transformation
6 Implementation and Result
6 Implementation and Result

- Branch mesh
  - Generate cylinder mesh between two adjacent particles
    - Not explicitly generate a tree graph
- Shadows
  - Computed by using Variance shadow maps
- Obstacle collisions
  - Signed distance filed
- Simulation of fluids and physics response with a time step
  - \( t = 25ms \)
6 Implementation and Result

Performance measurements

<table>
<thead>
<tr>
<th>Fig.</th>
<th>NP (k)</th>
<th>T (ms)</th>
<th>R (%)</th>
<th>P (%)</th>
<th>G (%)</th>
<th>C (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>18.16</td>
<td>27.65</td>
<td>29.38</td>
<td>42.42</td>
<td>0.55</td>
</tr>
<tr>
<td>7</td>
<td>plant: 5 fluid: 1</td>
<td>20.40</td>
<td>26.01</td>
<td>7.83</td>
<td>43.85</td>
<td>22.12</td>
</tr>
<tr>
<td>9</td>
<td>5</td>
<td>9.66</td>
<td>44.42</td>
<td>14.11</td>
<td>41.07</td>
<td>0.40</td>
</tr>
<tr>
<td>11</td>
<td>10</td>
<td>17.05</td>
<td>29.46</td>
<td>29.68</td>
<td>40.37</td>
<td>0.48</td>
</tr>
<tr>
<td>10</td>
<td>7</td>
<td>16.82</td>
<td>26.87</td>
<td>34.17</td>
<td>38.46</td>
<td>0.50</td>
</tr>
<tr>
<td>14, l</td>
<td>10</td>
<td>16.87</td>
<td>29.77</td>
<td>33.55</td>
<td>35.08</td>
<td>1.60</td>
</tr>
<tr>
<td>14, r</td>
<td>25</td>
<td>51.76</td>
<td>11.63</td>
<td>39.51</td>
<td>48.33</td>
<td>0.53</td>
</tr>
</tbody>
</table>
6 Implementation and Result

Parameters used for the results

<table>
<thead>
<tr>
<th>Fig.</th>
<th>N</th>
<th>B</th>
<th>BP</th>
<th>V</th>
<th>Ph</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6</td>
<td>2</td>
<td>0.4</td>
<td>0.2</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td>9</td>
<td>4</td>
<td>1</td>
<td>0.1</td>
<td>0.2</td>
<td>0.5</td>
</tr>
<tr>
<td>11</td>
<td>8</td>
<td>1</td>
<td>0.2</td>
<td>0.1</td>
<td>0.5</td>
</tr>
<tr>
<td>10</td>
<td>5</td>
<td>1</td>
<td>0.2</td>
<td>0.2</td>
<td>0.5</td>
</tr>
<tr>
<td>14</td>
<td>3</td>
<td>1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.7</td>
</tr>
<tr>
<td>14, r</td>
<td>20</td>
<td>1</td>
<td>0.3</td>
<td>0.2</td>
<td>0.5</td>
</tr>
</tbody>
</table>

N: number of plant  
B: maximum number of lateral buds  
BP: branch probability  
V: branching variance  
Ph: phototropism
6 Implementation and Result

6.1 Results

Figure 12
Phototropism

Gravity and high stiffness

Low stiffness
6 Implementation and Result

6.1 Results

Drags

Support structure

Figure 6

Figure 11

Figure 6

Figure 14
6 Implementation and Result

6.2 Evaluation

- Compare our results to photographs of real climbing plants

Real climbing plants

Our system

Figure 13

Wong et al. "A procedural approach to modelling virtual climbing plants with tendrils"
*Computer Graphics Forum* 2015
7 Discussion and Limitations

• Limitation
  ▪ Global control
    • Difficult to predict
  ▪ Species
    • Singleness
  ▪ Biomechanically-plausible simulation
    • Not provide
Our approach

- Provides an efficient means for the control over plant development
  - Allowing the user to affect growth parameters and physical properties of the plant

- Handles efficient modeling of external effects
  - Can be induced at any time without prior analysis of the plant structure

- Provide powerful editing capabilities
  - Allow to modify a plant with respect to its structure and its environment while maintaining a biologically plausible appearance

- Show the efficiency of our approach on a wide variety of interactive examples
8 Conclusion and Future Work

Future Work

• First
  ▪ Explore particle-based method and meshless deformation methods with a stronger focus on biological and physical plausibility

• Second
  ▪ Using particles for the efficient modeling of secondary growth.
    • E.g. development of growth rings, cracking of bark
Interactive Modeling and Authoring of Climbing Plants

Torsten Hädrich\textsuperscript{1} and Bedrich Benes\textsuperscript{2} and Oliver Deussen\textsuperscript{1} and Sören Pirk\textsuperscript{3}

\textsuperscript{1}University of Konstanz, Germany
\textsuperscript{2}Purdue University, USA
\textsuperscript{3}Stanford University, USA