A Study on the Animations of Swaying and Breaking Trees based on a Particle-based Simulation

Yasuhiro AKAGI
Tokyo University of Agriculture and Technology
Naka-cho 2-24-16
184-8588, Koganei, Tokyo
scout@cc.tuat.ac.jp

Katsuhiro KITAJIMA
Tokyo University of Agriculture and Technology
Naka-cho 2-24-16
184-8588, Koganei, Tokyo
Kitajima@cc.tuat.ac.jp

ABSTRACT

In this paper, we propose a particle-based simulation method to create the animations of swaying and breaking trees. Since the shapes of trees and their realistic motions are usually complex, it is difficult for us to create an animation of a swaying tree manually. Therefore, to produce films and image contents which contain a natural scene of swaying and breaking trees, it takes a lot of work to create the animation. To solve this problem, it is important that how to calculate interactions between a tree and wind to automatically generate the swaying and breaking motions of a tree by using a physical simulation. We model both a tree and wind as particles to simulate the interactions. The advantage of the particle-based method is that the method is robust for changing of the topology of wind and the branching structure of a tree. Our results show that the proposed method can naturally represent the breaking behavior of a tree and the wind flow around the tree by using the particle-based simulation of the wind.

Keywords

Tree animation, Breaking branches, Fallen leaves, Particle-based simulation, SPH method, Elastic deformation

1. INTRODUCTION

The 3D modeling and creation of animations of natural scenes is one of the most time-consuming works of producing films and videos[Chn11]. It is difficult to manually create an animation of swaying trees that has realistic movements of branches and leaves in an environment having a complex wind flow. Therefore, there are many studies on the animation of swaying trees on the basis of tree dynamics and wind simulations. We propose a practical method of generating tree animations that simulate the swaying motions and breaking behaviors of trees by using a particle-based model. We represent both the trees and wind by particles and links to be able to simulate the breaking behaviors of trees. The most advantage of the method using the particles is that the simulation method can freely divide the tree structure into broken branches and fallen leaves. This method also represents the change of the movement which is caused by the breaking and changing the shape of the tree.

Previous work

Most of these studies propose a method of simulating the dynamics of branches by using beam structures[AWZ09]. These methods simulate the bending motions of the beam structures (branches) on the basis of external forces (wind). For instance, Habel [HKW09] and Yang[YHY+10] proposed a method of automatically animating swaying trees using beam structures. Diener [DRB+09] proposed a simpler method of simulating bending branches. This method divides a tree into the sections and simulates the movements of branches only by the rotation of these sections. The method enables us to generate tree animations that contain thousands of trees in real time. Then, our method simulates the deformation of the beam structure of a branch by using the particles and connection of the particles.

In recent years, there are several studies on the simulation of the breaking behaviors of branches and falling leaves for generating tree animations [BCN+10]. Saleem[SCZ+07] proposed a method that simulates the breaking behaviors of branches. In this method, the breaking positions of a branch are decided on the basis of the threshold velocity, which is the velocity of the wind around the branch. When the breaking position of a branch is detected, the method detaches the broken branch from the tree and simulates the falling motion of the branch. Since the threshold velocities are set by the users beforehand, it takes a considerable amount of time to create the tree animations[Chn11]. Moreover, this method cannot detect the broken positions of branches and cannot generate the cross-sectional shape of a broken branch automatically.

On the other hand, to generate fluid animations, many studies on the simulation of complex movements such as broken waves, splashing, and pouring have been conducted [AT11][CBP05]. These studies are based on the SPH method[Luc77], which is a particle-based fluid dynamics method. One of the features of the SPH method is that it represents a fluid as a set of particles that makes it easy to simulate the separation and fusion of the fluid. Another feature of the SPH method is that it is easy to parallelize. With the use of this method, Goswami[GSS+10] succeeded in accelerating the generation of fluid animations by using GPUs. Stava et.al.[SBB+08] proposed "Surface particles" to simulate the erosion of a terrain by defining the relationship between particles and terrain.

Moreover, there are several studies based on the particle method for simulating the deformations of elastic objects. Muller[MC11] proposed a particle-based simulation method for elastic objects. One of the features of this method is that it effectively represents elastic objects by using ellipsoid particles. Gerszewski[GBB09] also succeeded in deforming elastic objects robustly by using particle-based models. Since the particle-based approach has the advantage of geometric changes, there are several studies on the destruction simulation of solid objects. Pauly[PKA+05] succeeded in simulating the destruction of solid objects by replacing the finite element meshes of an object with particles. [WRK+10] proposed a method for simulating deformation and fracturing of elastic objects. This method efficiently simulates the elastic objects with remeshing technique. Since the trees have complexly branched structures and intersections, we use the particle-based method to simulate the deformation and fracturing of trees to prevent the miss generation of meshes.

2. PHYSICAL SIMULATION OF SWAYING AND BREAKING TREES

In this section, we propose a simulation method to create an animation of swaying and breaking trees. This method is based on a particle-based simulation of wind and elastic deformations. We model both wind and trees as particles that have the same physical parameters in order to handle the interactions between the wind and the trees independently. The advantages of using the particle-based method are as follows:

1. If the physical parameters of the particles, such as position and velocity, are discrete, the interactions between the particles can be simply simulated as a particle-to-particle problem.
2. The particle-based structure has an advantage with respect to a breaking behavior that contains a topology change.
3. As the particles are discrete, it is easy to speed-up the simulation by using a parallel processing technique.

	Parameter	Symbol
Particle	Position	x_i
	Velocity	u_i
Prt_i	Acceleration	a_i
	Interaction Radius	r_i
	Mass	m_i

Table 1. Physical parameters of a particle

In the following sections, we describe the equations of our simulation by using the symbols given in Table 1.

Types of interaction models

When we perform a particle-based physical simulation, it is important to define the interactions between particles. In order to distinguish between the physical characteristics and interactions of the wind and the trees, we define the following four types of particles:

Wind : wind particle

Branch: particle for a part of a branch

Leaf : leaf particle

Bone : particle for the bone structure of a tree

We define three of the abovementioned types of particles for representing a tree. Two of these three are the types that help distinguish between the branches and the leaves. The fourth one represents the structure of a tree, such as the bone structure for a CG character. This bone particle is used for calculating the restoring force. We will describe the effect of this type of particles in Section 2.

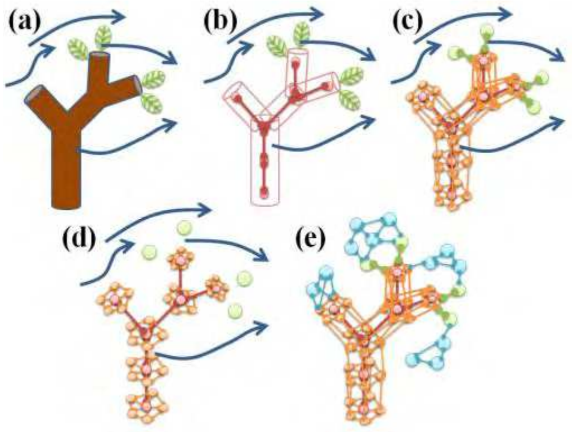

Figure 1. Generation process of the particle and link-structure based tree model form a polygon-based tree model

Particle-based tree modeling

In this section, we describe a method of modeling trees on the basis of the abovementioned four types of particles. To simulate the swaying motions and breaking behaviors of a tree, we define a link structure of the particles that are used for replacing the 3D tree model. The link structure represents the relationships between the particles for detecting an interaction force between two particles. In the proposed method, we require two types of information for the 3D tree model. One is the information regarding the differences between branches and leaves, and the other is the information regarding the tree structure and the positions of the branching points. By using the information of the tree, the proposed method generates the particles and link structures by following steps:

		Destination			
		Wind	Branch	Leaf	Bone
Source	Wind	f_{wind}	f_{wind}	f_{wind}	f_{wind}
	Branch	f_{wind}	f_{spr}	f_{spr}	f_{spr}
	Leaf	f_{wind}	f_{spr}	-	-
	Bone	f_{wind}	$f_{spr} + f_{shr}$	-	f_{bnd}

Table 2. Interaction model

(a) Define central axis

We use the links that are named the "central axis" [TMW02]. The central axis connects two particles to form the bone structure of a tree. We generate these links on the basis of the information regarding the positions of the branching points of the 3D tree model. Then, the proposed method generates subdivision particles at regular intervals in order to calculate the bending of a branch.

(b) Replace branches and leaves with particles

The particles for representing branches are generated on the circumference of a circle whose center is formed by the particles of the central axis generated in step (a). The radius of the circle is the same as that of the branch, and the normal vector of the circle is the same as the direction of the central axis. The particles for representing the branches are generated on the central position of the leaf.

(c) Generate link structure

We generate the link structure of a tree on the basis of the particles that are generated in the previous step. Particles for various parts of a branch are linked side by side on the circumference of the circle and are linked to the particles of the central axis, which are the center particles of the circle. These particles are also linked to particles that are linked to the neighboring circles. The leaf particles are linked to the nearest particles of the parent branch.

(d) Define particles and links for wind

We define the cubic area for simulating the wind, and the wind particles are generated in this area at regular intervals. The links from a wind particle are generated on the basis of the influence radius of the particle, and the links are updated in each time step of the simulation.

Next, we define the four types of interaction forces as follows:

f_{wind}	: Force of the wind from the fluid simulation
f_{spr}	: Internal force of a tree based on a mass-spring system
f_{bnd}	: Restoring force from the central axis
f_{shr}	: Shear stress from the structure of a branch

Table 2 shows the relationships between the four types of particles and the four types of interaction forces. The names of the types in the first column are the source of the interaction force, and those in the second row from the top are the destination.

The wind particles affect all types of interactions. We calculate the interaction force f_{wind} using the Navier-Stokes equations. Further, we simulate the elastic deformations of a tree by using three types of interaction forces. We will describe the method for simulating the elastic deformations in fifth Section.

Wind simulations

The interactions from the wind particles are calculated using the Navier-Stokes equations, which are one of the fundamental equations in the field of fluid dynamics. We apply the SPH method, which is a numerical solution of fluid dynamics, to simulate the airflow around a tree. Since the SPH method handles both fluids and objects as particles, it can be used for simulating the interactions between the wind and the trees. Equation (1) gives the interaction force $f_{wind\,i}$, which is the force exerted by all types of particles on particle Prt_i.

$$f_{wind\,i} = \mu \nabla^2 u_i - \nabla p_i + \rho_i f \qquad (1)$$

μ: viscosity, p_i: pressure,

ρ_i: density, f: external force

The first factor on the right side of equation 1 is the viscosity factor and the second factor is the pressure factor. The following equations are the differential equations for each factor.

$$\mu \nabla^2 u_i = \mu \sum_j m_j \frac{u_j - u_i}{\rho_j} \nabla W(x_j - x_i) \qquad (2)$$

$$\nabla p_i = \sum_j m_j \frac{p_j + p_i}{2\rho_j} \nabla W(x_j - x_i) \qquad (3)$$

The variable W denotes the kernel function of the SPH method[MC11] between Prt_i and Prt_j.

By using these equations, we simulate the interactions between the wind and the trees on the basis of the particle definitions. Then all of the relationships between wind and trees are detected based on the radius of the particles. In this process, since we deal with the particles of trees as same as the particles of wind, the two-way effect are calculated by the SPH method. When we implement the SPH method, we set the natural density of the air for the parameter ρ_i and the mass of the tree m_j is calculated with $0.5 g/cm^3$.

Interaction models for branches and leaves

We described the link structures of branches and leaves for simulating elastic deformations of a tree in the second section. In this section, we describe the three types of interaction forces that affect the particles for representing a tree. Figure 2 shows the link structure and the interaction forces of a branch.

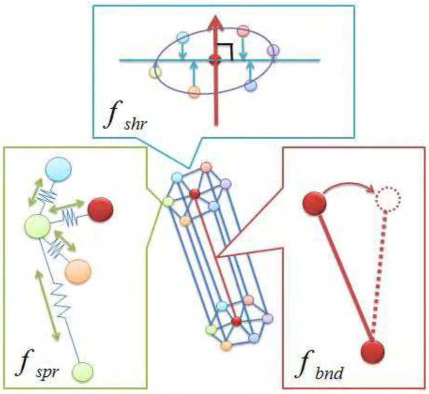

Figure 2. Three types of interaction forces for simulation of the elastic deformation of a branch

The first interaction force $f_{spr\,i}$ is the internal force of a tree based on a mass-spring system. The value of $f_{spr\,i}$ is given by equation 4.

$$f_{spr\,i} = \sum_j k_{i.j}\left(\left(x_j - x_i\right) - L_{i,j}\right) \quad (4)$$

$k_{i.j}$: Spring constant between Prt_i and Prt_j

$L_{i,j}$: Initial length between Prt_i and Prt_j

The second interaction force $f_{bnd\,i}$ is the restoring force from the central axis. Now, there are two particles for representing the central axis, Prt_i (top) and Prt_j (bottom). Then, we define the direction vector of the central axis $V_{center\,i}(= x_i - x_j)$. When we determine the restoring force, we require the relative rotation angle of the central axis to be against the initial rotation. To determine the rotation angle, we calculate the quaternion Q_j, which represents the rotation from $V_{center\,j}$ to $V_{center\,i}$ on the basis of the initial position of the tree beforehand.

$$Q_j(V_{center\,j}) = V_{center\,i} \quad (5)$$

Then, we define the interaction force $f_{bnd\,i}$ as follows:

$$f_{bnd\,i} = G_{b\,i}\left(Q_j\left(L_{i,j}\left|V_{center\,j}\right|\right) - V_{center\,i}\right) \quad (6)$$

$G_{b\,i}$: Elastic modulus

The third interaction force f_{shr} is the shear stress from the structure of a branch. This force moves Prt_i for representing branches on the circle whose center is formed by the particles of the central axis Prt_j. The normal vector of the circle is given by equation 7:

$$V_{normal\,i} = \frac{V_{center\,i}}{|V_{center\,i}|} \quad (7)$$

By using the normal vector, we can determine the distance and the direction from Prt_i to the circle. Equation 8 gives the relationship between both the distance and the direction and the interaction force f_{shr}.

$$f_{shr\,i} = G_{s\,i}\left(V_{normal\,i} \cdot x_j - V_{normal\,i} \cdot x_i\right)V_{normal\,i} \quad (8)$$

$G_{s\,i}$: Modulus of rigidity

We simulate the elastic deformations of trees by using these three types of interaction forces.

Breaking conditions of a tree

In this section, we describe the determination of the broken locations of a tree. We represent the breaking behavior of a tree by erasing the links between the particles that compose the particle-based model of the tree. We assign the limitation length of a stretch $e_{i,j}$ and a compress $c_{i,j}$ to each link of the tree model. Usually the breaking conditions are detected based on the stress of an object. In our method, the stress of a link can be calculated with $f_{spr\,i}$. Since the value of $f_{spr\,i}$ simply proportion the length between particles, we use the length to detect the broken position to be easy to set the threshold parameters by users. To determine the broken link between Prt_i and Prt_j, we use the strain measure of the link $d_{i,j}$.

$$d_{i,j} = \left|\frac{(x_j - x_i)}{L_{i,j}}\right| \quad (9)$$

If the strain rate of the link $d_{i,j}$ is greater than $e_{i,j}$ or lesser than $c_{i,j}$, the link is erased from the tree model.

$$IsBroken = \begin{cases} true & d_{i,j} \geq e_{i,j} \ \ or \ \ d_{i,j} \leq c_{i,j} \\ false & e_{i,j} > d_{i,j} > c_{i,j} \end{cases} \quad (10)$$

The threshold parameters $e_{i,j}$ and $c_{i,j}$ are given by the users. Then, if half of the links that are generated along the central axis are erased, the link of the central axis is erased.

3. GENERATION OF BROKEN BRANCH

In this section, we describe a method of modeling a broken shape of a branch. This method automatically generates a polygon model that represents a cross-sectional shape of a broken branch on the basis of the link structure of a tree described in Section 2.

Internal model of a branch

A branch is constructed by a bundle of fibers, and the cross-sectional shape of a broken branch is uneven. We define the internal model of a branch that represents the difference in the density, radius, and color of fibers in order to generate the unevenness. This internal model consists of three layers: core, wood, and bark. By controlling the thickness of each layer, this model can handle a branch that has an internal cavity, such as a bamboo.

Polygon model for broken branch

In this section, we describe the generation of a polygon model for a broken branch on the basis of the internal model of a branch (Figure 3). The generation procedure is as follows:

(a) The layered structure of a branch.

(b) The vertices that represent the fibers are placed randomly into the area of each layer according to its density.

(c) By using the Delaunay triangulation, we connect the vertices to form triangles.

(d) The triangles generated in step (c) are expanded to form triangle poles.

In our implementation, the broken shape is dynamically generated when the branch is broken. In our preliminary experiment, the computation time of the generation process is match faster than other simulations (< 0.005sec.).

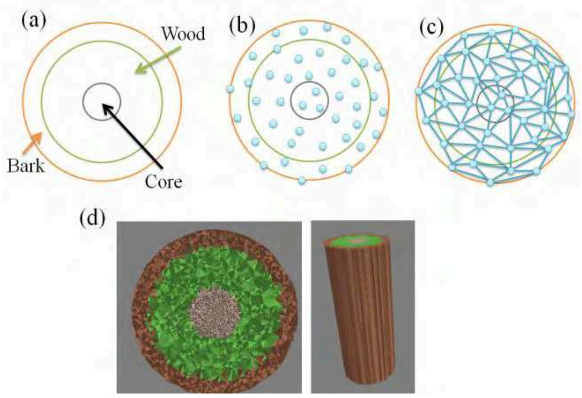

Figure 3. Generating process of a polygon model for broken branch.

Cross-section of a broken branch

In this section, we describe the division of the polygon model generated in Section 3. When a natural branch is broken at an arbitrary position on the surface of the branch, the breaking process proceeds to the other side. Then, the cross-sectional shape of the branch forms a curve that goes down to the bottom gradually. Therefore, we define the side curve of the cross-sectional shape on the basis of equation 11. This equation gives the drop rate F of the side curve (Figure 4 upper-side).

$$F(x) = L * \alpha \left(\frac{(C-x_0)\cdot(x-x_0)+1}{2r^2}\right)^2 \qquad (11)$$

L: Length of central axis, α: Brittleness,

C: Position of central axis, x_0: Starting position of the breaking.

The bottom side of the figure 4 shows the cross-sectional shape of the branch.

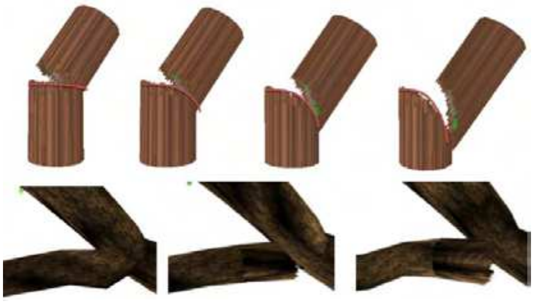

Figure 4. Cross-sectional shapes of a branch.

Upper-side: Braking goes down to the bottom.

Bottom-side: Braking behavior in a scene.

4. Results and Discussions

Particle-based tree modeling

We generate a particle- and link-based tree model by using the method described in Section 2. Figure 5 shows the polygon-based model and the particle-based model of a tree. One of the advantages of this method is that since the intervals between the particles can be changed, the users can freely change the details of the simulation of the tree.

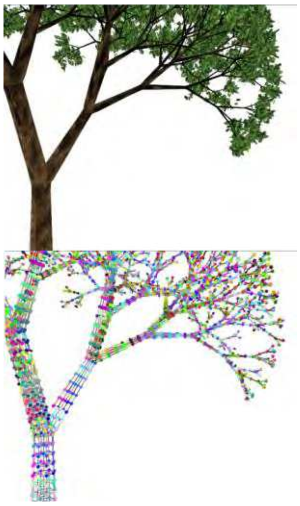

Figure 5. Result of generating particle-based model from a 3D model of a tree.

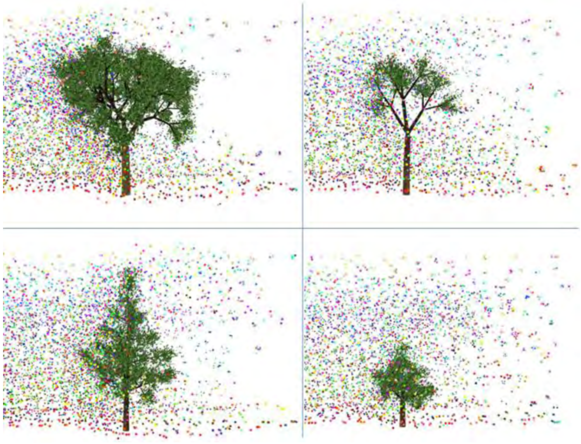

Figure 6. Comparison of the interaction between a tree and wind flow according to the differences in the shapes of trees.

Interactions between wind and tree

In this section, we present the results of the tree animations that contain the breaking behaviors of branches and falling leaves.

4.1.1 Swaying branches and leaves

Figure 6 shows the comparison of the wind flow according to the differences in the shapes of trees. The color dots on the images are the particles used for representing the wind. Since the branches and leaves of the trees block the wind particles, the larger and denser tree the more wind flows around the tree. These results show that the proposed method can represent the natural interactions between the tree and the wind.

4.1.2 Breaking branches and falling leaves

Figure 7 shows the Comparison of the broken conditions of branches according to the differences in the velocity of the wind. The red circles are the signs of the broken branches. In the previous study, Saleem et.al. [SCZ+07] enable to visualize the flying effect of the branches. However, this method calculates the broken conditions by using the angle of the branch and velocity of wind. Therefore the Salem's method could not visualize the difference of the broken position of a tree according with the velocity of wind. In our result, the image on the top of figure 7 is the result of the lowest velocity of the wind. This results show that there are any broken branches and a few fallen leaves. The faster the velocity of the wind causes the more number of the broken branches and fallen leaves. This result shows that the proposed method can visualize the difference of the broken position according with the velocity of the wind and the structure of the tree.

4.1.3 Implementation and performance

In this section we discuss the simulation performance of our method. We measured the processing time on a 2.66Ghz Xeon PC with 8GB RAM. We set the radius of the particles for wind 0.05m and place the particles for the trees every 0.1m. The time-step of our simulation is 0.01sec. Table3 shows the results to compare the time scaling according to the number of the particles for trees. This result shows that the increase of the number of the particles causes the increase of both the time for simulating wind and the tree. Since our method calculates the relationship between a tree and wind in the process of SPH method, the processing time for SPH method also increase according to the particles of a tree. The table 4 shows the results to compare the time scaling according to the number of the particles for wind. This result shows that since the relationship between the tree and wind are considered in the process of SPH method, the processing time for computing the tree deformation is not changed according to the number of the particle of wind. Since the processing times are not real-time,

Number of particles		Calculation time per fame (sec.)		
Wind	Tree	SPH	Tree Deformation	Total
7680	16877	0.225	0.134	0.360
	22319	0.356	0.178	0.535
	31812	0.565	0.264	0.829
	44222	0.958	0.367	1.325
	72759	2.412	0.617	3.028
	102282	4.350	0.869	5.219

Table 3. The time scaling according to the number of the particles for trees.

Number of particles		Calculation time per fame (sec.)		
Wind	Tree	SPH	Tree Deformation	Total
2560	21900	0.262	0.177	0.439
5120		0.310	0.174	0.484
9984		0.395	0.174	0.569
17664		0.610	0.176	0.787
25088		1.096	0.175	1.270
35072		2.312	0.181	2.492

Table 4. The time scaling according to the number of the particles for wind.

5. CONCLUSIONS

We proposed the particle-based simulation method to create the animations of swaying and breaking trees. We obtained the following results:

(1) We defined the four types of particles (wind, branch, leaf and bone) and the four types of interaction forces (wind, internal force, restoring

force and shear stress) between the particles. By using the models, we can simulate the swaying and breaking behaviors of trees.

(2) We proposed the generation method of a polygon model for a broken branch. This method is able to generate the cross-sectional shape of a broken branch on the basis of the link structure of a tree.

(3) Our results show that the animation of swaying and breaking trees is automatically generated by the proposed method. By using the particle-based simulation of the wind, the proposed method can represent that the wind flow naturally around the tree.

In a future work, we will accelerate our simulation method to be able to generate animations in real-time by using parallel processing on GPU.

6. ACKNOWLEDGMENTS

This research was supported by Adaptable and Seamless Technology Transfer Program through Target-driven R&D (A-Step), Japan Science and Technology Agency.

7. REFERENCES

[AT11]Ando, R., Tsuruno, R., A particle-based method for preserving fluid sheets, In Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA '11), pp.7-16, 2011.

[AWZ09]Ao, X., Wu, Y., Zhou, M., Real time animation of trees based on BBSC in computer games, International Journal of Computer Games Technology, Article 5, 2009.

[BCN+10]Baxter, R., Crumley, Z., Neeser, R., Gain, J., Automatic addition of physics components to procedural content, In Proceedings of the 7th International Conference on Computer Graphics, Virtual Reality, Visualisation and Interaction in Africa (AFRIGRAPH '10), pp.101-110, 2010.

[CBP05]Clavet, S., Beaudoin, P., Poulin, P., Particle-based viscoelastic fluid simulation, In Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation (SCA '05), pp.219-228, 2005.

[Chn11]Chng. E., Realistic Placement of Plants for Virtual Environments. IEEE Comput. Graph. Appl. 31, 4, pp.66-77, 2011.

[DRB+09]Diener, J., Rodriguez, M., Baboud, L., Reveret, L., Wind projection basis for real-time animation of trees, Computer Graphics Forum (Proceedings EUROGRAPHICS 2009), 28, 2, pp. 533-540, 2009.

[GSS+10]Goswami, P., Schlegel, P., Solenthaler, B., Pajarola, R., Interactive SPH simulation and rendering on the GPU, In Proceedings of the 2010 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA '10), pp.55-64, 2010.

[GBB09]Gerszewski, D., Bhattacharya, H., Bargteil, W.A., A point-based method for animating elastoplastic solids, In Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA '09), pp.133-138, 2009.

[HKW09]Habel, R., Kusternig, A., Wimmer, M., Physically Guided Animation of Trees, Computer Graphics Forum (Proceedings EUROGRAPHICS 2009), 28, 2, pp.523-532, 2009.

[Luc77]Lucy, L., A numerical approach to the testing of the fission hypothesis, 82, pp.1013-1024, 1977.

[MC11]Muller, M., Chentanez, N., Solid simulation with oriented particles, ACM Trans. Graph., 30, 4, Article 92, 2011.

[PKA+05]Pauly, M., Keiser, R., Adams, B., Dutr, P., Gross, M., Guibas, J.L., Meshless animation of fracturing solids, ACM Trans. Graph. 24, 3, pp.957-964, 2005.

[SBB+08]Stava, O., Benes, B., Brisbin, M., Krivánek, J., Interactive Terrain Modeling Using Hydraulic Erosion. In Proceedings of Symposium on Computer Animation. pp.201-210, 2008.

[SCZ+07]Saleem, K., Chen, S., Zhang, K., Animating Tree Branch Breaking and Flying Effects for a 3D Interactive Visualization System for Hurricanes and Storm Surge Flooding, In Proceedings of the Ninth IEEE International Symposium on Multimedia Workshops (ISMW '07), pp.335-341, 2007.

[TMW02]Tobler, F. R., Maierhofer, S., Wilkie, A., A multiresolution mesh generation approach for procedural definition of complex geometry. In Shape Modeling International 2002, 11, 15, pp.35-43, 2002.

[WRK+10]Wicke, M., Ritchie, D., Klingner,M. B., Burke, S., Shewchuk, R. J., O'Brien, F. J, Dynamic Local Remeshing for Elastoplastic Simulation, In Proceedings of ACM SIGGRAPH 2010, pp. 49:1–11, 2010.

[YHY+10]Yang, M., Huang, M., Yang, G., Wu, E., Physically-based animation for realistic interactions between tree branches and raindrops, In Proceedings of the 17th ACM Symposium on Virtual Reality Software and Technology (VRST '10), pp.83-86, 2010.

Figure 7. Comparison of the broken conditions of branches according to the differences in the velocity of the wind.