

Windy Trees: Computing Stress Response for Developmental Tree Models

Sören Pirk^{1,2} Till Niese² Torsten Hädrich² Bedrich Benes³ Oliver Deussen²

¹Esri R&D Center Zurich, Switzerland

²University of Konstanz, Germany

³Purdue University, USA

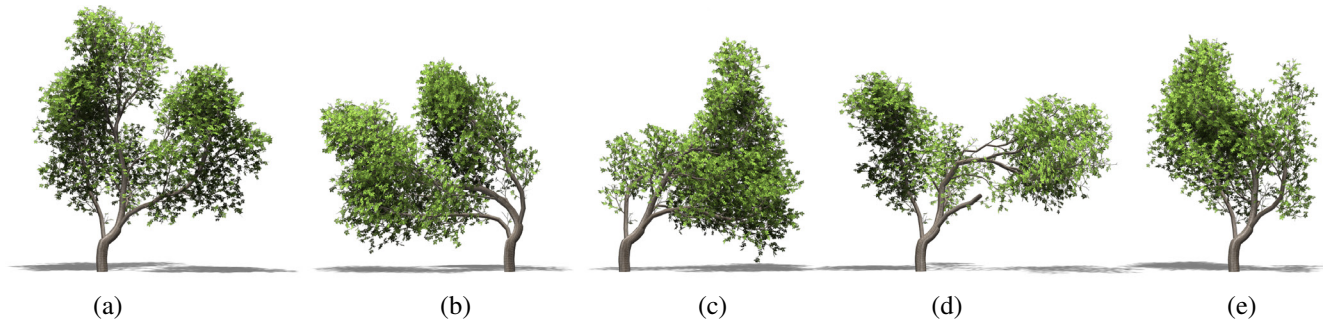


Figure 1: A 3D tree model is imported to our framework (a) and develops according to a prevailing wind direction (b) and (c). Besides considering wind as developmental factor our system also handles the breaking of branches (d) and the abrasion and drying of buds (e).

Abstract

We present a novel method for combining developmental tree models with turbulent wind fields. The tree geometry is created from internal growth functions of the developmental model and its response to external stress is induced by a physically-plausible wind field that is simulated by Smoothed Particle Hydrodynamics (SPH). Our tree models are dynamically evolving complex systems that (1) react in real-time to high-frequent changes of the wind simulation; and (2) adapt to long-term wind stress. We extend this process by wind-related effects such as branch breaking as well as bud abrasion and drying. In our interactive system the user can adjust the parameters of the growth model, modify wind properties and resulting forces, and define the tree’s long-term response to wind. By using graphics hardware, our implementation runs at interactive rates for moderately large scenes composed of up to 20 tree models.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling; I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction Techniques I.6.8 [Simulation and Modeling]: Types of Simulation—Visual

Keywords: Generative Tree Modeling, Interactive Procedural Modeling, Visual Models of Trees, Plant Growth, Simulation, Animation

Links: [DL](#) [PDF](#) [WEB](#)

1 Introduction

Trees are complex systems that heavily interact with their environment. The actual tree shape results from internal plant characteristics and external factors such as light, temperature, or available nutrients. The interaction of trees with light has been investigated frequently over the years [Honda 1971; Měch and Prusinkiewicz 1996; Palubicki et al. 2009]. Several methods concentrate on modeling the sway motions of developed trees and plants in wind fields [Diner et al. 2006; Habel et al. 2009; Shinya and Fournier 1992]. However, no method exists that models the influence of dynamic wind fields on the developmental process of a tree.

We close this gap by introducing a method for modeling tree development within a realistic wind field. Wind affects trees in many ways; destructive winds can rip off single branches or blow down entire trees, strong winds can also stretch the tree roots and separate them from the soil thus decreasing water absorption, bend their branches (sculpting influence), while moderate and low winds can significantly affect the heat exchange within a tree [Chaney 2001] or cause bud abrasion and drying [Putz et al. 1984]. The latter effects are caused by swaying branches that mechanically destroy buds. In this paper, we focus on the effects that are common and can be found in all trees. We do not focus on catastrophic events and limit ourselves to branch bending and breaking as well as bud abrasion and drying.

All effects depend on wind duration and intensity as well as the actual exposure of the tree. A tree at the edge of the forest will be affected much more than a tree that is surrounded by other trees or obstacles. By using Smoothed Particle Hydrodynamics (SPH) for the wind simulation we are able to model wind fields realistically in real-time and also account for the aforementioned local effects in the developmental process.

The acting forces are used to move the tree realistically but are also integrated over time to change the tree morphology during growth. While wood is elastic at a young age, it becomes increasingly rigid over time and plastic deformations occur if wind constantly acts on the tree skeleton. We use the mechanical property of wood which prevents it from recuperating completely from the applied force as a way of expressing tree response to the long-term stress caused by wind.

Figure 1 shows a 3D tree model (consisting of 15,600 graph vertices and 33,176 leaves) processed with our system. The younger developmental stages of the input model (a) develop according to a prevailing wind direction (b) and (c). Our framework also handles environmental effects, such as the breaking of branches (d) and the abrasion and drying of buds (e).

2 Related Work

Modeling trees and plants has been an important topic in computer graphics for almost forty years. While early approaches were inspired by fractals and repetitive patterns [Aono and Kunii 1984; Kawaguchi 1982; Oppenheimer 1986; Smith 1984], more recent approaches focus on interactive modeling [Ijiri et al. 2006; Lintermann and Deussen 1999], sketching [Longay et al. 2012; Okabe et al. 2007], and on expressing environmental interaction [Greene 1989; Měch and Prusinkiewicz 1996; Benes and Millán 2002; Palubicki et al. 2009]. L-systems [Lindenmayer 1968] are the most developed formal approach for plant modeling and allow for the expression of either simple repetitions or, as in recent extensions, are able to capture and express the environmental interaction [Měch and Prusinkiewicz 1996; Palubicki et al. 2009]. Other approaches either capture real-world data, e.g., from images [Reche-Martinez et al. 2004; Neubert et al. 2007] and laser-scanned point sets [Livny et al. 2011; Xu et al. 2007], or utilize user-defined sketches to model the branching structure [Quan et al. 2006].

While modeling the branching structure is important when defining tree and plant models, recent approaches concentrate on dynamically adapting tree models and more complex animations. Pirk et al. [2012b] introduced a modeling approach that adds environmental sensitivity to static input exemplars and enables tree models to adapt to changing environmental conditions. More recently, Zhao and Barbič [2013] proposed a method for the interactive authoring and simulation of adult plant and tree models. They discretize tree models based on an FEM model and show that this reduction can be utilized to model a variety of effects, also including wind animations. However, despite recent advances in synthesizing continuous animations of biologically- and physically-plausible tree models, real-time synthesis still remains a challenge.

Several approaches exist to animate the reaction of trees and plants to wind. One class uses spectral approximation which captures the characteristic motions of wind based on noise. One of the early approaches utilizes a field of random velocities in the frequency domain to simulate the power spectrum of wind passing through trees [Shinya and Fournier 1992]. Stam [1997] investigates the motion of branches based on filtered white noise in the Fourier domain. Other more recent approaches focus on retrieving the power spectrum from real world data [Diener et al. 2006]. An advantage of noise-based methods is their generality and consistency. However, it is difficult to create a specific spectra and they often require pre-processing to be used efficiently.

Coupling wind models with tree skeletons and modeling the underlying biomechanics are important aspects when animating and simulating tree models [Wang et al. 2013]. Ota et al. [2003] combine $1/f^\beta$ noise power spectra for approximating wind with a spring model to describe branch dynamics. Diener et al. [2008] propose more evolved mechanical means to model a tree’s response to varying wind-fields. The approach of Habel et al. [2009] couples a noise power spectrum with Euler-Bernoulli beams to efficiently compute branch motions. Bertails et al. [2008] provide a thorough overview of strand dynamics also related to tree dynamics. While the existing approaches already expose a high level of complexity, it is not possible to capture the full variety of parameters involved in nature at interactive rates.

Physical models for wind animations are commonly based on computational fluid dynamics. Several methods exist to integrate the flow field, however only a small number of previous work employs them to animate tree and plant models. Akagi and Kitajima [2006] proposed a method for animating the two-way dynamics of tree-wind interaction with a particle-based fluid simulation. Oliapuram and Kumar [2010] as well as Yang et al. [2011] follow this approach and propose acceleration methods for solving computationally expensive equations directly on graphics hardware. A more recent work by Selino and Jones [2013] concentrates on meshfree simulations of particles using *Smoothed-Particle Hydrodynamics (SPH)* [Lucy 1977]. Their work allows to animate grown tree models while we focus on developmental tree models.

Research in forestry and botany predominantly concentrates on the tree’s responses to weather conditions and the resulting stress on its branching structure [Marshall 1998]. Petola et al. [1996] investigate a tree’s reaction to wind and the consequences for groups of trees. Fourcaud et al. [2003] go even further and propose numerical models for shape regulation and growth stress caused by different wind conditions. Recently, the aerodynamic properties of trees have been investigated [Sellier and Fourcaud 2009]. James et al. [2006] as well as Ye [2013] focus on the mechanical stability of branching structures under dynamic loads. Moore et al. [2004] provided a more general overview of tree-wind interaction.

In a recent work Derzaph and Hamilton [2013] introduced a model for capturing growth characteristics of simple tree models in relation to wind. Since it is intended for games, their method does not integrate a physically plausible wind model and their trees are modeled in a schematic and very limited way. In contrast to their method, we focus on the efficient processing of biologically-plausible plant development in real-time and full complexity.

3 Overview

The input to our method is a set of parameters of a developmental tree model, the scene geometry, and the set of parameters describing the wind emitters, their geometry, and their behavior. In the simplest configuration it is a single tree with one wind emitter, but more complex scenarios with obstacles and multiple trees and emitters are used. Our system supports developmental models that provide branch competition for resources and tropism such as [Greene 1989; Měch and Prusinkiewicz 1996; Palubicki et al. 2009; Benes and Millán 2002]. In our framework, we use the tree model by Pirk et al. [2012a].

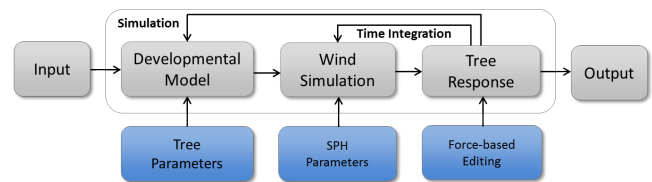


Figure 2: Overview of our method. The input is a set of parameters of a developmental model and the wind simulation. The developmental model provides a 3D geometry of the tree that interacts with the wind. The changes and behavior are integrated in a time integration loop. and plant shape.

We simulate wind with a Lagrangian fluid model that uses particles as discrete quantities (SPH). The SPH are adaptive, allow for quick integration as well as interaction with the tree geometry, and provide complex fluid phenomena such as turbulence, fusion, and separation. The wind affects the tree by employing forces that cause dynamic behavior of the branches.

The tree’s response is calculated by adding particle integrators to the tree geometry, which allow for time integration of the wind effect. By using this method we can capture the effect of the forces over user-defined periods of time. After each integration step the resulting changes in the geometry are calculated and transferred to the developmental model that changes the growth directions accordingly and grows the tree.

Our framework allows for various levels of *user control* as shown in the lower row of Figure 2. A user can directly manipulate the tree parameters, such as its mechanical response to the applied forces, the width of branches, or the number of leaves. The wind properties can be manipulated by changing its distribution, strength, direction, or there may be multiple wind emitters in the scene.

Similar to other approaches, our system represents a tree model as a skeletal graph $G = \{V, E\}$ [Palubicki et al. 2009; Livny et al. 2011], where V are the vertices and E the edges. Given an edge e_i and its corresponding vertices (v_s, v_t) a hierarchical relationship is produced, with v_s being the ancestor of v_t ; one of the vertices is the root node: v_{root} . Each branch is a chain of a varying number of edges $C = \{e_1, e_2, \dots, e_n\}$ within the same branching level; n denotes the length of a chain. These chains not only represent the connection of vertices but also the biological properties of the branching structure such as wood characteristics. Processing a tree graph instead of the full mesh geometry lowers the computational costs for simulation and structural adaptation. At rendering time the mesh geometry is produced on the fly with a modern graphics API shader pipeline.

4 Wind Simulation

A wind field can be described by the Navier-Stokes equations, which can be solved by a variety of algorithms [Bridson 2008]. Our method could make use of any wind simulation that allows for calculating the forces exerted on the tree geometry. However, in our framework we use Smoothed Particle Hydrodynamics (SPH), which gained importance in recent years for a number of reasons. In contrast to Eulerian approaches that require spatial subdivision, particles are trackable objects in 3D space and thus allow easier simulation. Moreover, they are implicitly adaptive since they move in the areas where calculation is needed. The imprecision caused by the varying density of particles in different regions can be solved by adaptive particle splitting.

SPH is a flexible approach that accounts for objects that move quickly in the wind. The interaction of trees and wind fields is a two-way process. When trees are affected by wind they create turbulence and eddies in the wind field that can easily be integrated by the SPH approach. This is another big advantage of SPH over other methods. Finally, SPH can be computed on the GPU which allows simulating even complex tree models at interactive rates.

4.1 Smoothed Particle Hydrodynamics

Each SPH particle has its position \mathbf{x} and represents local physical quantities such as density or pressure. The Navier-Stokes equations describe the acceleration a_i of the i -th particle as the material derivative of the velocity

$$\mathbf{a}_i = \frac{d\mathbf{v}_i}{dt} = \frac{-\nabla p + \mu \nabla^2 \mathbf{v} + \rho \mathbf{g}}{\rho_i}, \quad (1)$$

which depends on the fluid’s density ρ , pressure $(-\nabla p)$, viscosity $(\mu \nabla^2 \mathbf{v})$ and external forces $(\rho \mathbf{g})$. The quantities at a certain location are computed by summing the relevant quantities from contributing particles that lie within a certain radial distance. The con-

tribution of each j -th particle is weighted by a distance-based kernel smoothing function

$$A(\mathbf{x}) = \sum_{j=1}^N \frac{m_j}{\rho_j} A_j W(\mathbf{x} - \mathbf{x}_j, h), \quad (2)$$

where A is the calculated quantity (such as density or pressure), \mathbf{x} the particle position, m its mass, ρ the particle density, and W the smoothing kernel with kernel radius h . The fluid development over time is computed by using Euler integration. These computations yield a 3D velocity field representing the wind direction [Liu and Liu 2003].

4.2 Wind Collision and Response

Another feature of particle-based wind dynamics that makes them suitable for our purpose is that it is easy to calculate collisions of particles with other objects. We use signed distance fields to represent distance to obstacles in our system. During the computation of the particle movement we check collisions by looking-up the distance to the closest obstacle similar to Guendelman et al. [2003]. The collision of fluid particles and trees is described in Section 5. To perform fast distance tests we approximate complex (non-tree) objects in our framework with bounding volumes. When a collision is detected the distance field provides us with the distance of the particle to the closest point on the surface and the corresponding direction. As a reaction to the collision, the velocity and position of the particle is updated.

5 Tree Dynamics

We compute the dynamics of swaying tree motions with an approximate general force model for branching structures [Sakaguchi and Ohya 1999]. If forces act for a longer time in the same direction, their results are transferred to the developmental model and affect the growth direction of the tree (see Section 6). Additionally, we compute more complex effects such as the breaking of branches (see below) and the abrasion and drying of buds (Section 6.4).

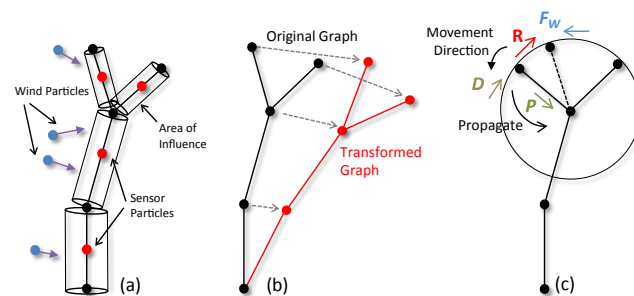


Figure 3: Force-based transformations of a tree graph: a branching structure is influenced by a force field (a). Internodes are affected by the forces propagated through the branching structure and yield a transformed tree graph (b). An external wind force (F_W) is compensated with restoration (R), damping (D), and propagation (P) forces.

To measure the forces of fluid particles we use sensor particles distributed along the tree structure. The sensor particles measure and integrate the forces for passing fluid particles and thereby allow for a two-way coupling of the tree graph and the wind field. Figure 3(a) shows a branching structure with associated sensor particles. To account for a different resistance of branches and leaves, we differentiate between sensor particles distributed along the edges and

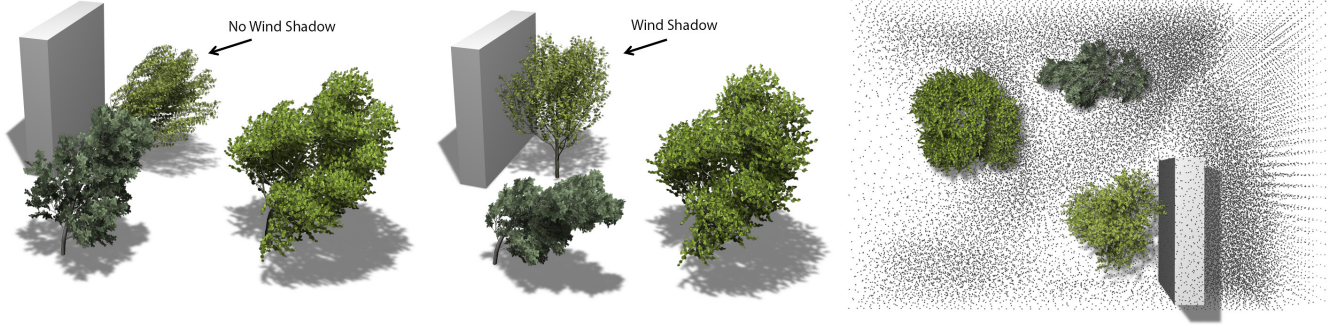


Figure 4: In contrast to vector fields, SPH provide plausible fluid effects and efficient collision handling. Left: All trees in the scene are affected by the same vector field without handling collisions. Middle + Right: Objects in the scene interact with the particles and vice versa. The box causes a wind shadow that protects the tree behind it. Each of the trees influences the particles and thereby influences their density.

cluster particles placed in the foliage. Both particle types are surrounded by sensor volumes that allow us to quantify the passing fluid particles [Akagi and Kitajima 2006; Selino and Jones 2013]. For edge particles we simply perform cylinder point tests to instantaneously resolve the collision. Fluid particles within the volumes of cluster particles receive a penalty force, simulating the interaction of wind and leaves.

In order to achieve precise results, we would need to cover each triangle of the input mesh, including leaves and branches, with particles distributed with equal density. However, this would be impractical, because detailed tree models often consist of tens of thousands of triangles. Instead, we approximate the tree structure by particles distributed along the edges e_i of the tree graph and use a dynamic clustering of smaller branches and leaves which are then represented by cluster particles. Similar to the methodology described by Müller et al. [2011], each branch sensor particle has a cylindrical volume (Figure 3(a)) whereas cluster particles are represented by spheres. This allows for a significant reduction in the number of particles to a couple of hundreds for each tree.

Figure 3 shows schematically the way the tree is modified and how forces act on it. The tree is affected by the wind field that creates a force on the tree nodes and its connecting edges. Integration of the forces causes the tree graph to move to a new position (Figure 3 (b)). Trees have a large damping due to internal wood characteristics and the air resistance of the leaves. This is modeled by damping forces \mathbf{D} that are computed in every simulation step. After the wind has gone, restoration forces \mathbf{R} move the tree back into its original position.

5.1 Tree Forces

As described in Oliapuram and Kumar [2010] a branch is modeled as a chain of edges, where each edge e is a rigid link with the corresponding vertices $E = \{v_s, v_t\}$; a vertex vt rotates around its parent v_s . When an external force hits a tree graph, the edge movement is determined by computing the temporal derivative of the angular velocity ω of the edge, which depends on the torque \mathbf{N} and the moment of inertia \mathbf{I} (rod):

$$\mathbf{N} = \mathbf{I} \frac{d\omega}{dt}; \quad \mathbf{I} = \frac{mr^2}{3}, \quad (3)$$

where m is the branch mass and r is the length of the edge e . The actual values of the mass and length are calculated from the tree geometry obtained from the developmental model in Section 6.3.

The momentum \mathbf{N} is calculated as the cross product of the forces \mathbf{F} acting on the edge and the edge vector \mathbf{e} : $\mathbf{N} = \mathbf{F} \times \mathbf{e}$. Similar

to Sakaguchi and Ohya [1999] we compute the force \mathbf{F} as

$$\mathbf{F} = \mathbf{F}_W + \mathbf{R} + \mathbf{D} + \mathbf{P} + \mathbf{L}, \quad (4)$$

where \mathbf{F}_W is the external wind force, \mathbf{R} is the restoration force of the branch to its resting position, \mathbf{D} is an axial damping force, \mathbf{P} is the back propagation force that propagates the forces acting upon the child branch to the parent branch and \mathbf{L} is a force term representing the drag of leaves. To correctly transfer the forces back, the dynamics computation is performed recursively from the outer edges to the root edge. Figure 3 (c) illustrates the acting forces on a branching structure.

The wind model provides a velocity vector affecting the tree graph. For the computation of the wind force we employ the aerodynamic drag equation

$$\mathbf{F}_W = S_b \sigma \nu, \quad (5)$$

where S_b is the normal projected area of the surface facing towards the wind, σ is a drag coefficient (here approximated with a value of 0.6), and ν is the external wind velocity.

The *restoration force* \mathbf{R} moves a deflected branch towards its rest position, relative to the parent edge. The force direction \mathbf{d}_r is the spherical direction to the rest position. Its strength depends on the branch rigidity k and its angular displacement α :

$$\mathbf{R} = \mathbf{d}_r k \alpha. \quad (6)$$

Due to strong binding forces among branch segments the motion of a branch is suppressed by the *axial damping force*. We model the damping as acting against the edge angular velocity, proportional to the edge thickness coefficient μ :

$$\mathbf{D} = -(\hat{\omega} \times \hat{\mathbf{e}}) \mu \omega |\omega|. \quad (7)$$

When a branch is deflected from its initial position, forces acting on a child branch are propagated to the parent branch. To correctly transfer the forces back, the dynamics computation is performed recursively from the outer edges to the root edge. The *back propagation force* \mathbf{P}_{i-1} is propagated to a branch from its child branches

$$\mathbf{P}_{i-1} = - \sum k_i \mathbf{F}_{Ri}, \quad (8)$$

where k_i is the propagation coefficient of the force. The ratio of child and parent branch thickness is

$$k_i = k_c \frac{T_{h_i}}{T_{h_{i-1}}}, \quad (9)$$

where k_c is the fixed propagation coefficient and T_{h_i} and $T_{h_{i-1}}$ are the thickness of the parent and the child vertex respectively. If the

restoration force of child branches is zero the propagation force is zero as well.

To represent the influence of leaves in the force model we add the *leaf force* \mathbf{L} for each of the leaf clusters and compute it similar to the *wind force* \mathbf{F}_w :

$$\mathbf{L} = \mathbf{S}_1 \sigma \nu c, \quad (10)$$

where \mathbf{S}_1 represents the approximate projected area of the leaf surface, ν the external wind velocity and c a constant leaf coefficient for representing leaf characteristics of different species.

5.2 Breaking of Branches

A branch breaks when the acting forces exceed a certain level of stress. As our branching structures do not provide the required physical properties of wood, such as rigidity, stiffness, and hardness, we determine the required parameters by approximating Young's modulus and Hook's law similar to Ennos and van Carsteren [2010]. This is due to the fact that these properties heavily depend not only on the kind of wood [Laboratory 2013] but also on the local growth structure. Wood is a highly anisotropic material that incorporates drastic local changes that are not modeled by our approach. Thus we compute the current stress σ acting upon an edge of our branching structure by

$$\sigma = \frac{4cM}{3\pi r^2}, \quad (11)$$

with c being the coefficient of Young's modulus E and the radius of the curvature R : $c = E/R$. The radius of the branch is denoted by r and M is the bending moment: $M = d \times F$, with d being the direction vector of the branch and F the force of the wind model. Similar to Cannell and Morgan [1989] we compute the maximum stress σ_{max} a branch is able to compensate as being proportional to the branch thickness d and a material property p as

$$\sigma_{max} = d^3 p. \quad (12)$$

p is commonly determined by measuring the properties of wood, in our system we let the user define it. A branch breaks when the current stress σ exceeds the maximum stress σ_{max} . In contrast to vector fields, particle-based collisions allow modeling the breaking of individual and upwind-sided branches (Figure 12).

6 Tree Response

In contrast to previous methods our approach allows capturing wind influence at two different time scales. On one hand we animate tree motions induced by wind forces, on the other hand we compute long-term variations of tree growth due to wind fields and allow the user to shape trees this way. Our approach thus is a unified solution for tree animation and modeling.

We achieve this by coupling the wind field with the tree graph. As mentioned above, we use sensor particles attached to the tree skeleton to measure the influence of SPH particles. Each of the particles exerts a force to the tree model and lets the tree move. If integrated over time, the average influence of particles is measured and the sensor is able to deliver a stable force signal if the wind field is stable over time. If coupled with a developmental tree model this signal can then be used to reshape the tree during its growth.

The same two trees growing at the same speed with two different wind fields show a significant difference. If the wind field changes its direction quite suddenly (unstable wind field), the model has a less clear direction than with a slowly changing wind field. In Figure 5 we demonstrate the effect for a rotating wind field. For

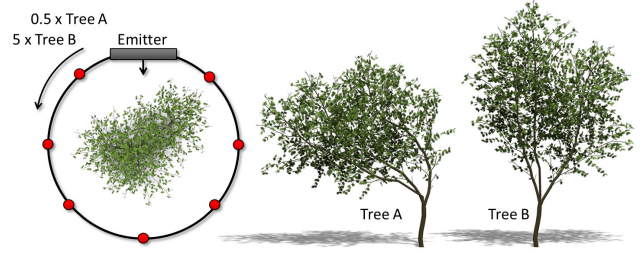


Figure 5: Two trees growing within a rotating wind field: Tree A grew within a slowly changing wind field and adapted its shape heavily due to the prevailing wind direction. Tree B grew in a fast rotating field for which the effects balance out.

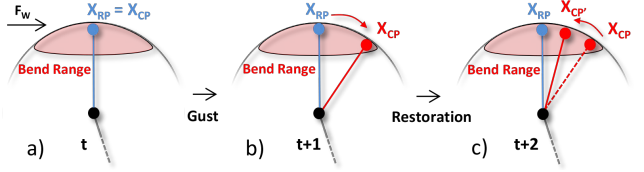


Figure 6: Immediate tree response: a vertex in its rest position (x_{RP}) is hit by an external force (a). This yields the current position (x_{CP}) that oscillates around its rest position (b). (c): as soon as the external force stops, it moves back to its rest position (elasticity).

Tree A the wind field rotated very slowly, only about 180 degrees during the growth of the model, whereas Tree B was affected by a field that rotated five times around the model during its growth. Here the influences of the wind balance out due to the unstable wind direction.

Even though we use SPH for the wind simulation and a particular growth model for our implementation, our approach can be used with any wind model and with any developmental model for trees. The main advantage of a particle-based wind model, however, lies in its ability to handle collisions (used for bud abrasion and drying) as well as lee effects (wind shadow) which frequently occur in most scenes (Figure 4).

6.1 Immediate Tree Response

Let us assume that a grown tree is swaying in a dynamic wind field. The forces applied to the tree result in a dynamic motion of its branches that we call *immediate tree response*. A vertex in its resting position x_{RP} is pushed by wind in time $t + \Delta t$ to its current position x_{CP} (cf. Figure 6, (a),(b)). As soon as the force stops the vertex moves back to its rest position x_{RP} (elastic restoration, Figure 6 (c)). The end vertex of each edge moves on a spherical surface that we call the *bend range*. It is implicitly defined by Eq. (4) in Section 5.1. Wind oscillations force the branch to move within this range.

The time integration of a position can be computed similar to Oliapuram and Kumar [2010]. The direction of an edge is given by

$$\begin{aligned} \theta' &= \theta + \omega(\Delta t) + \frac{1}{2}\alpha(\Delta t)^2, \\ \omega' &= \omega + \alpha(\Delta t), \end{aligned} \quad (13)$$

where θ and θ' represent the original and the new angular orientation, ω and ω' the current and new velocity and α the angular acceleration $\alpha = d\omega/dt$. Δt represents the time step of the inte-

gration. The actual transformation of a vertex is computed by

$$\begin{aligned} \mathbf{x}' &= R_\theta \mathbf{x} \\ \mathbf{x}'_{\mathbf{L}} &= \mathbf{x}' R_\theta^{-1}, \end{aligned} \quad (14)$$

where \mathbf{x}' represents the updated vertex position that is computed by transforming \mathbf{x} by the rotation matrix R_θ . The vector $\mathbf{x}'_{\mathbf{L}}$ represents the updated position in the local coordinate system of the vertex. From Eq. (14) we compute the current position (x_{CP}) and the rest position (x_{RP}) (cf. Figure 6).

6.2 Long-Term Tree Response

So far we only described the immediate reaction of a tree to a wind field. However, in nature a tree is subject to plastic deformation when it is exposed to stress of a wind field for a long time. The key idea behind our approach is that a branch will not recuperate perfectly to its rest position x_{RP} when exposed to wind for some time [Neild and Wood 1999]. To capture the corresponding long-term effect of a wind field on a tree model we sample and integrate the occurring forces for the sensor particles on the tree graph (Figure 7). We determine the alteration of the current rest position by integrating the wind field over time and by applying the resulting forces to the rest position.

The following paragraphs introduce two stabilization modes for capturing the long-term adaptations of branching structures caused by stress from wind fields. Although described separately they are both applied in parallel.

Dynamic stabilization allows structural changes to be modeled within a single developmental state (DS), by changing the rest position x_{RP} . The developmental position x_{DP} is the position to which the node would grow without the effect of wind. It is received from the developmental model and represents the neutral structure of the tree without any environmental influences. The developmental position serves as a reference for the interpolation between different stages under the influence of wind. As shown in Figure 7 the *rest range* defines the possible locations of rest positions for a branch.

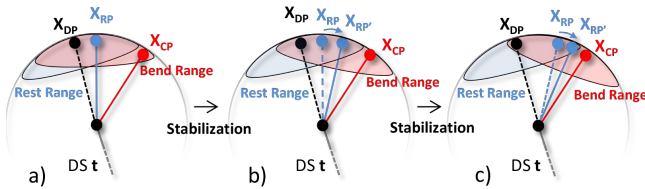


Figure 7: Dynamics stabilization: when the tree model is affected by an external force for a longer period of time, its rest position x_{RP} is slowly adjusted towards the current position x_{CP} (a), (b). To maintain the structural properties of the input model, the rest position is only allowed to move within the rest range (c). The developmental position x_{DP} is constant for a given developmental state (DS) t .

We update the structure of our tree model by moving the rest position towards a temporal average of its position. This transformation occurs over a period of time Δt and thus slowly adjusts the tree structure according to the dominant direction in the wind field. When the external force stops, the tree recuperates to this modified rest position (x_{RP}') and thus shows plasticity. However, this new rest position is not fixed; when an external force affects the tree from another direction, the rest position is adjusted according to the new force.

Growth stabilization accounts for the growth that affects the plasticity of branches. Since elasticity gradually decreases as a branch becomes older and thicker, plastic deformation increases. This is illustrated in Figure 8; the size of the rest range decreases as the branch becomes more rigid. At the same time, the developmental position (x_{DP}) slowly moves towards the rest position (x_{RP}). In fact our neutral tree model adapts towards the current model under the influence of wind.

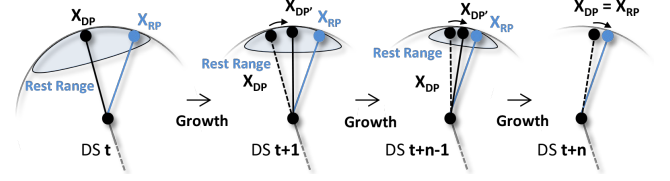


Figure 8: Growth stabilization: As the tree grows the developmental position (x_{DP}) is adjusted towards the rest position (x_{RP}). This slowly fixates the tree according to the direction of the wind field. To account for the increasing rigidity of branches, the size of the rest range decreases.

This fixates the orientation of the branch and stabilizes the tree model according to the average long-term wind direction. If we change the wind towards the opposite direction, the tree would not transform back to the neutral tree model but to a deformed shape since the plastic deformation was already integrated in the model. The only way to create a different deformation would be to move backwards in time and to let the tree grow again under the influence of the new wind direction.

The developmental position x_{DP} and the rest position x_{RP} are computed based on the description in Section 6.3. The transformations of x_{DP} and x_{RP} are performed as described in Eq. (14). While the *bend range* represents the current variations of the tree model, we need to explicitly compute the *rest range* to maintain the structural properties of the tree model during development. This range is computed based on a logistic function [Prusinkiewicz et al. 1993], the normalized thickness T_n and the constant scaling factor r :

$$E(T_n, r) = \left(1 + e^{(T_n - 1)2r + r}\right)^{-1}. \quad (15)$$

As stated above, both stabilization modes are applied at the same time and invoked when the tree starts to grow. This allows us to combine realistic tree simulation with developmental stabilization of the tree under the influence of long-term wind.

6.3 Developmental Models

The above mechanism can be applied to different developmental models. Typically, such models provide a developmental position (x_{DP}) and a rest position (x_{RP}) for branches as part of their modeling process. When it is possible to add sensor particles to their branching structure, both immediate and long-term response can be integrated easily.

To show that our model is suited for real-time applications we adapted the approach of Pirk et al. [2012a], which supports interactive computations while at the same time maintaining biologically plausible branching structures. However, this model focuses on computing a growth animation from a small tree towards the input geometry and ignores environmental effects. We extend the approach by integrating the growth adaptation from the previous section and show that it can be used to model structural adjustments of trees in dynamic wind fields.

An input tree model is analyzed and a skeletal graph is determined. This graph is then used to interpolate branching angles and branch



Figure 9: A set of pine trees processed with our system. The original tree model (a) suffers from bud drying and bud abrasion (b). A younger developmental stage is affected by wind and stabilizes its structure (c). Later in its growth process the wind direction changes and causes the tree to grow towards another direction (d). The models (b)-(d) were made from the input model (a).

radii during the growth process. The position of a vertex within a developmental state can be computed by

$$\mathbf{x}_m = R_{Prop}(\Phi)R_{Local}(\Phi)\mathbf{v}_y. \quad (16)$$

where $R_{Local}(\Phi)$ represents the local rotation matrix and $R_{Prop}(\Phi)$ the propagated rotation matrix from the previous branch segments. Φ is the rotation angle. \mathbf{v}_y represents the unit vector in the direction of the y-axis in a standard coordinate system and \mathbf{x}_m denotes the rest position and the current position in our model. The developmental position is given by computing the inverse tropism as shown by Pirk et al. [2012a]. By inserting the local transformations for x_{CP} and x_{RP} (represented as matrix R_{Local}) to Eq. (16), we are able to compute the current position and the rest position of the vertex accordingly.

Taking into consideration that young branches of a bent tree grow towards light, we additionally apply a simple form of phototropism. We compute the *bend axis* for the tropism by $\mathbf{b} = \mathbf{L} \times \mathbf{d}_e$, where \mathbf{L} represents the light direction and \mathbf{d}_e the direction of the edge. We compute the rotation matrix R_L and apply the tropism to the position of the edge vertex from the developmental model \mathbf{x}_m . We only modify the youngest branches and apply the tropism before the wind transformation. This provides visually plausible results; however, more complex light models (e.g. as described in Palubicki et al. [2009]) potentially yield more appealing results.

6.4 Bud Abrasion and Drying

Wind may also hurt newly developed buds. They may dry out due to heavy wind speeds or be abraded due to contact between branches. Both phenomena are well known to botanists since they influence the development of trees. Putz and Parker [1984] for example studied the formation of openings in forest canopies. In our simulation, buds are located on the youngest branches of the tree model. We account for the drying of buds by sampling the wind field at each bud and evaluate if the wind intensity exceeds a predefined user threshold. If this is the case the bud dries out and after some time is marked as dead and eliminated from the simulation. Similarly, bud abrasion is computed by counting the number of collisions of a bud with neighboring branches. We define a bounding sphere around each active bud and use it to detect the number of collisions. A bud is killed when the collisions reach a user-specified threshold. Figures 9 (b) and 12 (e) illustrate the bud drying and abrasion due to wind.

7 Implementation and Results

We have implemented our system in C++ and OpenGL on a desktop computer with an Intel i7 processor at 3.4 Ghz and 16 GB RAM. All results shown in the paper were rendered with an Nvidia Geforce 780 GPU in our framework. We employed the geometry processing capabilities of an OpenGL 4.2 shader pipeline to render our models. The meshes were produced on the fly on a frame to frame basis.

Table 1 shows computation times for the results shown in the paper using our non-optimized code. Depending on the model complexity we reach between 5 and 50 fps, for a large scene with dozens of trees one second per frame is needed.

Table 1: Modeling and rendering times for the Figures shown in the paper.

| Fig. | Tree Verts. (k) | Particles | | Updates | | | |
|-------|-----------------------|----------------|------------|-------------|----------------|----------------|-----------------|
| | | Cluster (k) | SPH (k) | SPH (ms) | Forces (ms) | Growth (ms) | Render. (ms) |
| 1 | 15,6 | 1,8 | 50 | 45 | 28 | 19 | 8 |
| 4 | 6,2 | 4,7 | 50 | 49 | 14 | 10 | 7 |
| 9 (a) | 34,1 | 1,6 | 20 | 38 | 63 | 57 | 9 |
| 10 | 12,7 | 0,8 | 10 | 33 | 13 | 12 | 6 |
| 11(a) | 7,3 | 1,2 | 10 | 32 | 14 | 12 | 6 |
| 11(e) | 2,5 | 0,1 | 5 | 12 | 4 | 3 | 6 |
| 12 | 4,3 | 1,4 | 50 | 41 | 8 | 7 | 6 |
| 15 | 63,8 | 14,5 | 100 | 256 | 119 | 104 | 13 |

7.1 User Interaction

Although the introduced system runs completely automatic, it provides several levels of user interaction. Our implementation of SPH depends on a number of parameters such as the number of particles in total, emitted particles per second, and physical properties of particles such as viscosity, mass, rest density, and gas constant. Moreover, the user can directly change the position of wind emitters as shown in the accompanying video. All objects interact with the fluid and therefore can be used to create complex interactions and wind flows.

The stabilization modes introduced in Section 6.2 enable realistic tree modeling and also support artistic needs. By directly placing the emitter, the *Dynamic Stabilization* enables users to reshape a given tree in its current developmental state. The tree will slowly lean away from the wind source and the rest position x_{RP} will move towards the average current position x_{CP} of the branches over a period of time Δt . Combined with *Growth Stabilization* a user can integrate stress response of a tree model for a given developmental state. This process can be applied repeatedly for different developmental stages and is then - what we call - *stress-based authoring* of tree models. We used this approach for generating the models shown in in Figure 11 (bottom).

7.2 Results

The examples in Figure 4 illustrate the ability of our particle-based fluid dynamics to model and track collisions. The particles interfere with the trees and create unique flow fields for each individual exemplar; this is much more precise than just using vector fields. The pine trees in Figure 9 were generated from the same input model. Based on different environmental conditions very different shapes can be generated.

Figure 10 shows three developmental stages of a tree model and how it developed under the influence of wind. The younger developmental stages receive wind from a certain direction and stabilize their structure. The models shown in Figure 11 show two fully developed trees modeled according to dynamically changing wind fields.

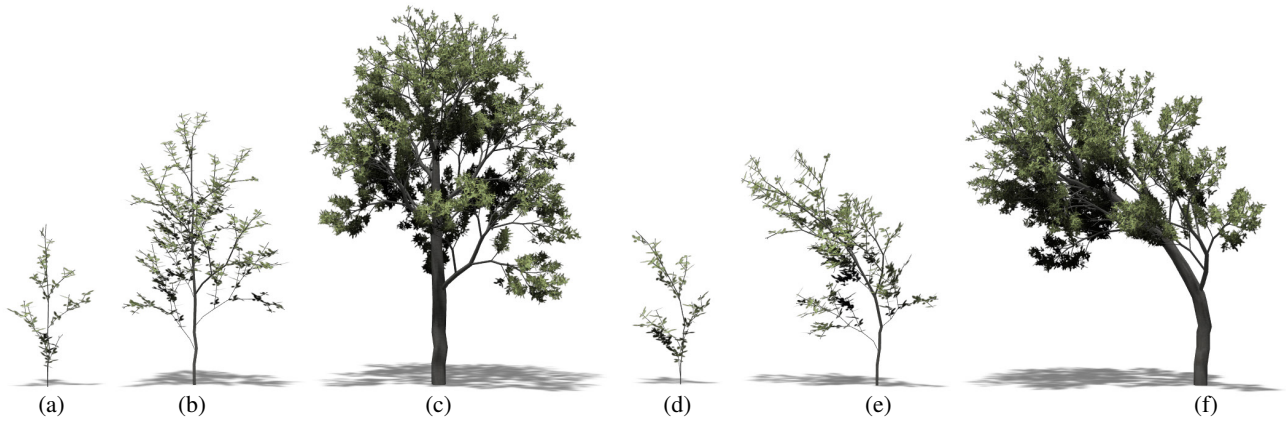


Figure 10: Comparison of three developmental stages of a tree grown with ((a)-(c)) and without ((d)-(f)) the influence of wind. The younger developmental stages ((d), (e)) receive wind from a certain direction and stabilizes their structure. All the models are processed from the input model shown in (c).

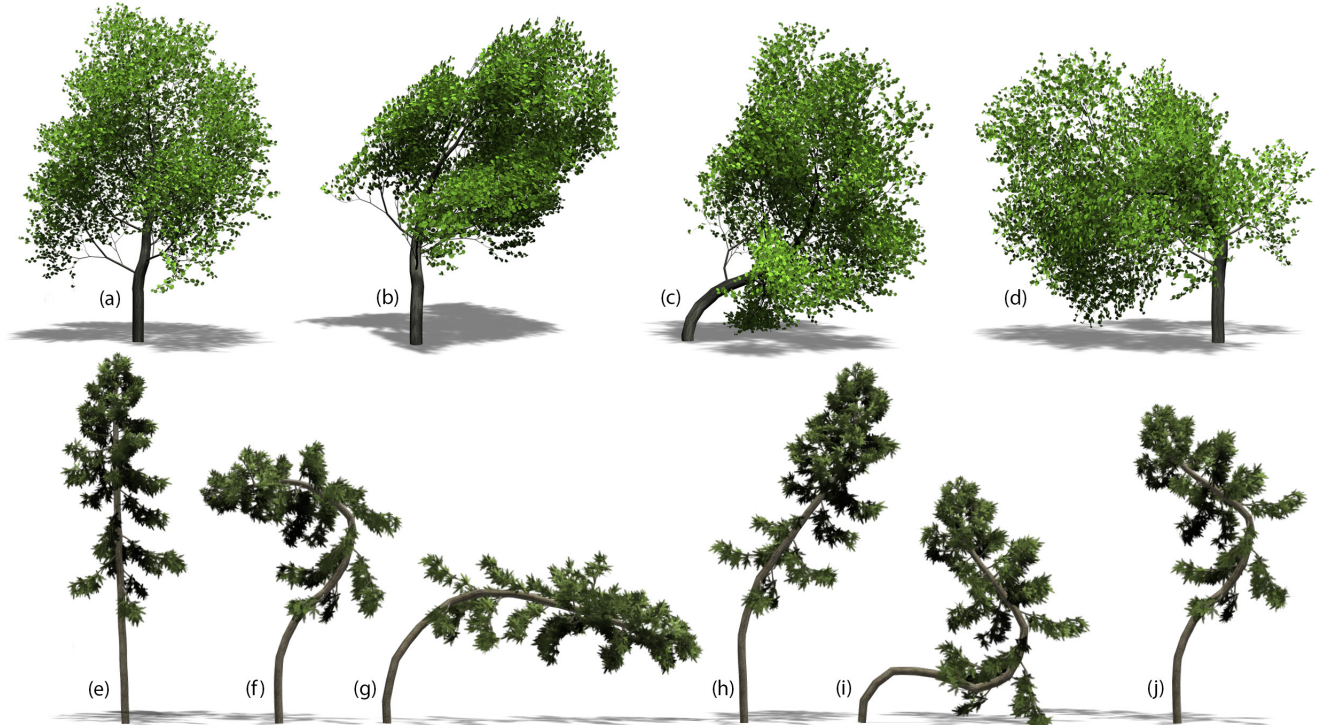


Figure 11: Two sets of models generated with our framework. A given tree model (a) and (e) is exposed to wind while it grows. The model stabilizes its structure (dynamics stabilization) and compensates the stress. The deformations are fixated when the model reaches a new developmental state. The models (b)-(d) and (f)-(j) are the result of a developmental process under the influence of a different wind field.

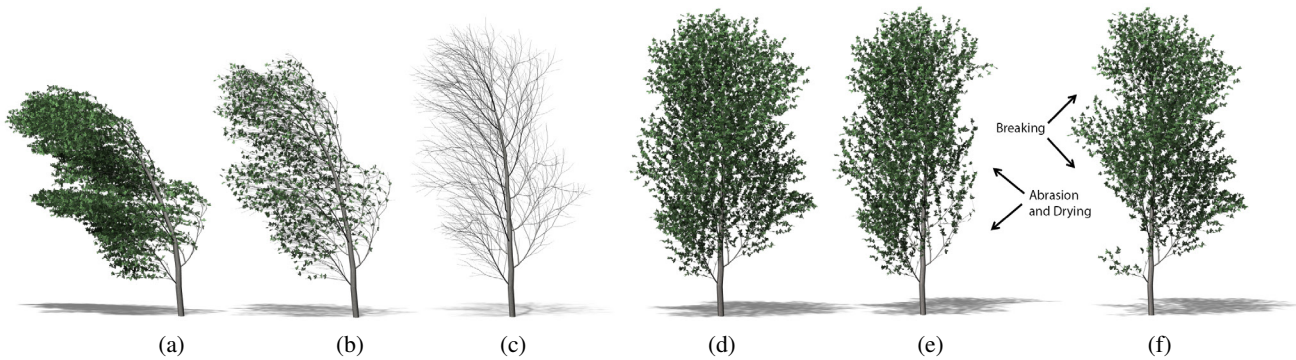


Figure 12: A tree model exposed to the same wind field gradually loses its leaves (a)-(c). As can be seen, the bending of the branching structure is dynamically affected the more leaves are attached. The input tree model (d) suffers from environmental influences: wind causes the abrasion and the drying of buds as the tree develops (e) and causes the breaking of branches (f).

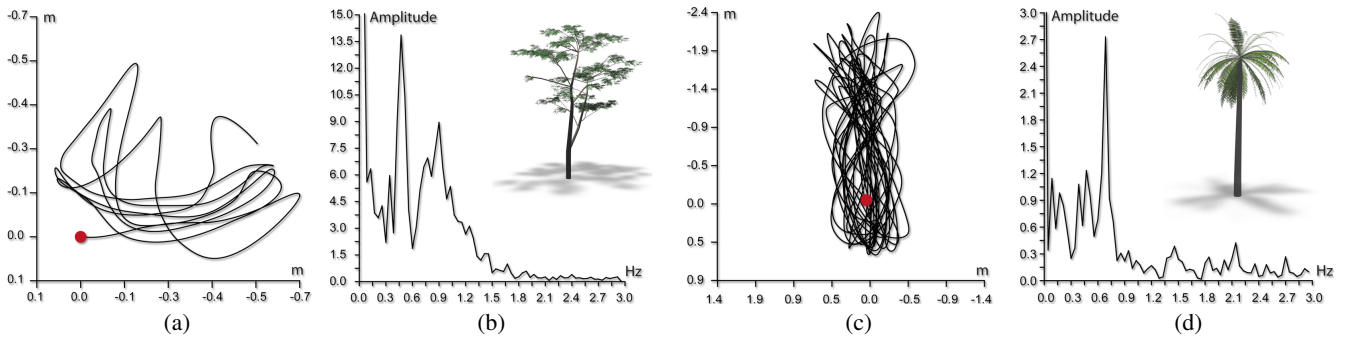


Figure 13: Sway motions and frequency spectra of a leafy tree and a palm tree captured in our system. The sway motion diagrams (a), (c) show the tracked position of a particle over a period of two minutes. The frequency diagrams indicate the dominant sway frequency of the tree models at about 0.45 Hz (b) and 0.7 Hz (d). Compared to the real data discussed in James et al. [2006] our measurements show similar characteristics.

In Figure 12 (a)-(c) we demonstrate one of the advantages of our particle-based modeling approach. A tree model gradually loses its leaves and as a result it is less affected by the wind field. The sub-figures (d) and (e) illustrate the abrasion and the drying of buds, (f) shows the effect of branch breaking. A large scene with 20 tree models is shown in Figure 15. The models were exposed to strong wind fields during their growth process. Each tree adapts its branching structure according to the prevailing wind direction at its location.

7.3 Evaluation

Most trees are exposed to wind during their entire life and compensate this stress with structural adaptations. Although the impact of stress on branching structures is an active subject of research in botany and forestry, we are not aware of measurements that represent the growth process of a tree with respect to wind over its entire life-time.

We can, however, evaluate our framework by measuring sway motions and frequency spectra of tree graphs exposed to our wind model. This is similar to James et al. [2006] who investigated sway motions and frequency spectra of real trees swaying in wind. They attached motion sensors to the branches to measure their stability during storms. To acquire similar data for our tree models we selected one of our sensor particles and tracked its movement over a period of time. Additionally, we computed the angular displacement between the current x_{CP} and rest position x_{RP} to determine the sway frequency of the tracked particle. We used a Fourier Transformation to analyze the dominant natural frequencies in this measured signal. To reproduce the randomness of real-world wind fields we added noise to the output velocity of the wind emitter.

The sway motion diagram (Figure 13, (a), (c)) shows sway motions of a sensor particle attached to the tree trunk at about two meters of height (in our case the fourth internode within the root chain). This particle represents the tracked displacement of the current position (x_{CP}). We measured the particle movement over a period of two minutes. The spectrum diagram shows the most dominant frequencies for two tree models (Figure 13, (b), (d)) at 0.45 Hz and 0.7 Hz. By adjusting the force terms discussed in Section 5.1 we are not only able to match the shape of the spectrum, but also the dominant frequency of 0.3 Hz as discussed in James et al. [2006]. This confirms that the immediate response to the wind of trees in our framework corresponds to real-world data.

Figure 14 shows our attempt to recreate models observed in nature. The first column shows the tree developed without any wind, the second column shows the tree affected by a prevailing wind, and the right image is the actual photograph. Two tree models were



Figure 14: Two of our models compared to photographs of real trees. The left shows the unaffected model, and in the middle shows the two authored models. Photographs courtesy of Alex Bamford, Lake District, UK (top) and Federica Gentile, Eastbourne, UK (bottom).

grown with our developmental model while exposed to different wind conditions. Our system allows growth characteristics to be reproduced similar to observations in nature. The overall shape of the simulated trees reflects the competition between phototropism (tendency to grow against the light) and the long-term prevailing stress caused by the wind.

8 Conclusion

We have introduced a method that combines tree developmental models with SPH for wind simulation into a single framework. The simulated wind exerts stress on a tree model which alters the growth direction based on elasticity and mechanical properties of the wood. Our framework generates tree shapes similar to those observed in nature, our implementation runs in real-time for single tree models and at interactive frame rates for moderately large groups. In addition we have shown that the introduced method provides novel possibilities for editing tree models by using wind as a bio-morphological mechanism.

However, the proposed system is not without limitations. Our simulation is a complex system and as such it is sensitive to initial conditions. For example, the sensor particles on each tree are associated with the vertices of the tree graph. Moving these sensors to different locations could alter the results; the effect of their location would be worthy of investigating as it could affect the computed forces. Similarly, the size and position of the emitter also effects the shape of the tree.

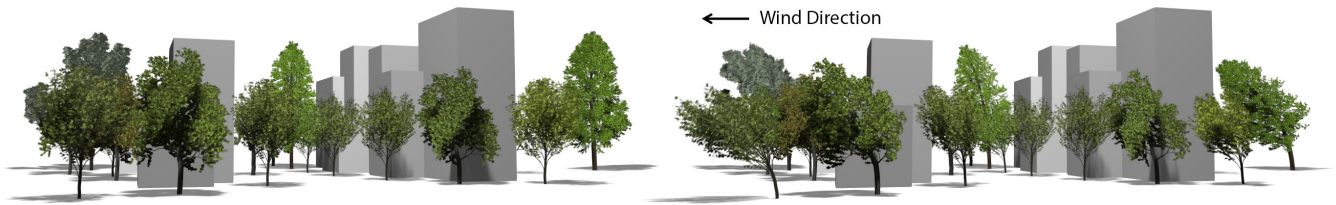


Figure 15: A large scene including 20 tree models rendered with our system. Left: The scene with tree models grown without the influence of wind. Right: the same scene with all models exposed to a wind field from one direction (as indicated by the arrow). Tree models and obstacles in the scene interact with the wind field and thereby cause individual growth conditions for each tree.

One open issue and an interesting avenue for future work would be to increase the level of control. Although the current implementation generates plausible results, it is sometimes difficult to achieve a desired shape for a tree model. It could help to combine our method with editing techniques as proposed by Longay et al. [2012] or Pirk et al. [2012b]. Currently, our system does not support secondary branch motions, i.e., forces are not propagated from the root to the outer branches. Leveraging more complex models for branch and leaf dynamics, e.g., such as the one proposed by Derzaph and Hamilton [2013], would allow generating more realistic behavior for non-interactive applications. Finally, we do not provide a level of detail approach for our system. However, for large scenes it would be desirable to vary the number of sensor particles dynamically for every tree model.

Acknowledgements

We thank the anonymous reviewers. This work was supported by the DFG Research Training Group GK-1042 "Explorative Analysis and Visualization of Large Information Spaces", University of Konstanz.

References

- AKAGI, Y., AND KITAJIMA, K. 2006. A study on the animations of swaying and breaking trees based on a particle-based simulation. *Journal of WSCG* 20, 1, 21–28.
- AONO, M., AND KUNII, T. 1984. Botanical tree image generation. *IEEE Computer Graphics and Applications* 4(5), 10–34.
- BENES, B., AND MILLÁN, E. 2002. Virtual climbing plants competing for space. In *IEEE Proc. of the Computer Animation 2002*, IEEE Computer Society, 33–42.
- BERTAILS, F., HADAP, S., CANI, M.-P., LIN, M., KIM, T.-Y., MARSCHNER, S., WARD, K., AND KAČIĆ-ALESIĆ, Z. 2008. Realistic hair simulation: Animation and rendering. In *ACM SIGGRAPH 2008 Classes*, ACM, New York, NY, USA, SIGGRAPH '08, 89:1–89:154.
- BRIDSON, R. 2008. *Fluid Simulation for Computer Graphics*. A K Peters, CRC Press.
- CANNELL, M. G., AND MORGAN, J. 1989. Branch breakage under snow and ice loads. *Tree Physiol* 5, 3, 307–17.
- CHANEY, W. R. 2001. How wind affects trees. *Indiana Woodland Stewart* 10(1).
- DERZAPH, T. K. M., AND HAMILTON, H. J. 2013. Effects of wind on virtual plants in animation. *International Journal of Computer Games Technology* vol. 2013.
- DIENER, J., REVERET, L., AND FIUME, E. 2006. Hierarchical re-targeting of 2D motion fields to the animation of 3D plant models. In *ACM SIGGRAPH / Eurographics Symposium on Computer Animation, SCA'06*.
- DIENER, J., RODRIGUEZ, M., BABOUD, L., AND REVERET, L. 2008. Wind projection basis for real-time animation of trees. Rapport de recherche RR-6674, INRIA.
- ENNOS, A. R., AND VAN CASTEREN, A. 2010. Transverse stresses and modes of failure in tree branches and other beams. *Proc Biol Sci* 277, 1685, 1253–8.
- FOURCAUD, T., BLAISE, F., LAC, P., CASTRA, P., AND DE REFFYE, P. 2003. Numerical modelling of shape regulation and growth stresses in trees. *Trees* 17, 1, 31–39.
- GREENE, N. 1989. Voxel space automata: Modeling with stochastic growth processes in voxel space. *SIGGRAPH Comput. Graph.* 23, 3, 175–184.
- GUENDELMAN, E., BRIDSON, R., AND FEDKIW, R. 2003. Non-convex rigid bodies with stacking. In *ACM SIGGRAPH 2003 Papers*, ACM, New York, NY, USA, SIGGRAPH '03, 871–878.
- HABEL, R., KUSTERNIG, A., AND WIMMER, M. 2009. Physically guided animation of trees. *Comp. Graph. Forum* 28, 2, 523–532.
- 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 on the shape of the tree-like body. *Journal of Theoretical Biology* 31, 331–338.
- IJIRI, T., OWADA, S., AND IGARASHI, T. 2006. Seamless integration of initial sketching and subsequent detail editing in flower modeling. *Comp. Graph. Forum* 25, 3, 617–624.
- JAMES, K. R., HARITOS, N., AND ADES, P. K. 2006. Mechanical stability of trees under dynamic loads. *American Journal of Botany* 93, 10, 1522–1530.
- KAWAGUCHI, Y. 1982. A morphological study of the form of nature. *SIGGRAPH Comput. Graph.* 16, 3, 223–232.
- LABORATORY, F. P. 2013. *Wood Handbook: Wood as an Engineering Material*. CreateSpace Independent Publishing Platform.
- LINDENMAYER, A. 1968. Mathematical models for cellular interaction in development. *Journal of Theoretical Biology Parts I and II*, 18, 280–315.
- LINTERMANN, B., AND DEUSSEN, O. 1999. Interactive modeling of plants. *IEEE Comput. Graph. Appl.* 19, 1, 56–65.
- LIU, G., AND LIU, M. 2003. *Smoothed Particle Hydrodynamics : A Meshfree Particle Method*. World Scientific Pub Co.
- LIVNY, Y., PIRK, S., CHENG, Z., YAN, F., DEUSSEN, O., COHEN-OR, D., AND CHEN, B. 2011. Texture-lobes for tree modelling. *ACM Trans. Graph.* 30, 4, 53:1–53:10.

- LONGAY, S., RUNIONS, A., BOUDON, F., AND PRUSINKIEWICZ, P. 2012. Treesketch: interactive procedural modeling of trees on a tablet. In *Proc. of the Intl. Symp. on Sketch-Based Interfaces and Modeling*, SBIM '12, 107–120.
- LUCY, L. B. 1977. A numerical approach to the testing of the fission hypothesis. *Astron.J.* 82, 1013–1024.
- MARSHALL, B. J. 1998. *Wind flow structures and wind forces in forests*. PhD thesis, University of Oxford, UK.
- MOORE, J., AND MAGUIRE, D. 2004. Natural sway frequencies and damping ratios of trees: concepts, review and synthesis of previous studies. *Trees* 18, 2, 195–203.
- MÜLLER, M., AND CHENTANEZ, N. 2011. Solid simulation with oriented particles. *ACM Trans. Graph.* 30, 4, 92:1–92:10.
- MĚCH, R., AND PRUSINKIEWICZ, P. 1996. Visual models of plants interacting with their environment. In *SIGGRAPH '96: Proc. of the 23rd annual Conf. on Comp, graphics and interactive techniques*, ACM, 397–410.
- NEILD, S. A., AND WOOD, C. J. 1999. Estimating stem and root-anchorage flexibility in trees. *Tree Physiol* 19, 3, 141–151.
- NEUBERT, B., FRANKEN, T., AND DEUSSEN, O. 2007. Approximate image-based tree-modeling using particle flows. *ACM Trans. Gr.* 26, 3, Article 71, 8 pages.
- OKABE, M., OWADA, S., AND IGARASHI, T. 2007. Interactive design of botanical trees using freehand sketches and example-based editing. In *ACM SIGGRAPH 2007 Courses*, ACM, SIGGRAPH '07.
- OLIAPURAM, N. J., AND KUMAR, S. 2010. Realtime forest animation in wind. In *Proceedings of the Seventh Indian Conference on Computer Vision, Graphics and Image Processing*, ACM, ICVGIP '10, 197–204.
- OPPENHEIMER, P. E. 1986. Real time design and animation of fractal plants and trees. *SIGGRAPH Comput. Graph.* 20, 4, 55–64.
- OTA, S., TAMURA, M., FUJITA, K., FUJIMOTO, T., MURAOKA, K., AND CHIBA, N. 2003. $1/f$ beta; noise-based real-time animation of trees swaying in wind fields. In *Computer Graphics International, 2003. Proceedings*, 52–59.
- PALUBICKI, W., HOREL, K., LONGAY, S., RUNIONS, A., LANE, B., MĚCH, R., AND PRUSINKIEWICZ, P. 2009. Self-organizing tree models for image synthesis. *ACM Trans. Graph.* 28, 3, 58:1–58:10.
- PELTOLA, H. 1996. Swaying of trees in response to wind and thinning in a stand of scots pine. *Boundary-Layer Meteorology* 77, 3-4, 285–304.
- PIRK, S., NIESE, T., DEUSSEN, O., AND NEUBERT, B. 2012. Capturing and animating the morphogenesis of polygonal tree models. *ACM Transactions on Graphics* 31, 6, 169:1–169:10.
- PIRK, S., STAVA, O., KRATT, J., SAID, M. A. M., NEUBERT, B., MĚCH, R., BENES, B., AND DEUSSEN, O. 2012. Plastic trees: interactive self-adapting botanical tree models. *ACM Transactions on Graphics* 31, 4, 50:1–50:10.
- PRUSINKIEWICZ, P., HAMMEL, M. S., AND MJOLSNESS, E. 1993. Animation of plant development. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, ACM, New York, NY, USA, SIGGRAPH '93, 351–360.
- PUTZ, F. E., PARKER, G. G., AND ARCHIBALD, R. M. 1984. Mechanical abrasion and intercorwn spacing. vol. 112, 24–28.
- QUAN, L., TAN, P., ZENG, G., YUAN, L., WANG, J., AND KANG, S. B. 2006. Image-based plant modeling. *ACM Trans. Graph.* 25, 3, 599–604.
- RECHE-MARTINEZ, A., MARTIN, I., AND DRETTAKIS, G. 2004. Volumetric reconstruction and interactive rendering of trees from photographs. *ACM Trans. Gr.* 23, 3, 720–727.
- SAKAGUCHI, T., AND OHYA, J. 1999. Modeling and animation of botanical trees for interactive virtual environments. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, ACM, 139–146.
- SELINO, A., AND JONES, M. D. 2013. Large and small eddies matter: Animating trees in wind using coarse fluid simulation and synthetic turbulence. *Comp. Graph. Forum* 32, 1, 75–84.
- SELLIER, D., AND FOURCAUD, T. 2009. Crown structure and wood properties: Influence on tree sway and response to high winds. *American Journal of Botany* 96, 5, 885–896.
- SHINYA, M., AND FOURNIER, A. 1992. Stochastic motion: Motion under the influence of wind. *Comp. Graph. Forum* 11, 3, 119–128.
- SMITH, A. R. 1984. Plants, fractals, and formal languages. In *SIGGRAPH '84: Proc. of the 11th annual Conf. on Comp. graph. and interactive techniques*, ACM Press, 1–10.
- STAM, J. 1997. Stochastic dynamics: Simulating the effects of turbulence on flexible structures. *Comp. Graph. Forum* 16, 3, C159–C164.
- WANG, H. Y., KANG, M. Z., HUA, J., AND WANG, X. J. 2013. Modeling plant plasticity from a biophysical model: Biomechanics. In *Proceedings of the 12th ACM SIGGRAPH Intl. Conf. on VRCAI*, ACM, 115–122.
- XU, H., GOSSETT, N., AND CHEN, B. 2007. Knowledge and heuristic-based modeling of laser-scanned trees. *ACM Trans. Gr.* 26, 4, Article 19, 13 pages.
- YANG, M., HUANG, M.-C., AND WU, E.-H. 2011. Transactions on edutainment vi. Springer-Verlag, Z. Pan, A. D. Cheok, and W. Müller, Eds., 27–39.
- YE, F. 2013. Branch dynamics: A theoretical interpretation of natural phenomena. *International Journal of Modern Nonlinear Theory and Application* Vol. 2, 1A, 74–77.
- ZHAO, Y., AND BARBIČ, J. 2013. Interactive authoring of simulation-ready plants. *ACM Trans. Graph.* 32, 4, 84:1–84:12.