

Fire in Paradise¹: Mesoscale Simulation of Wildfires

TORSTEN HÄDRICH, KAUST
DANIEL T. BANUTI, UNM
WOJTEK PAŁUBICKI, UAM
SÖREN PIRK, Google AI
DOMINIK L. MICHELS, KAUST

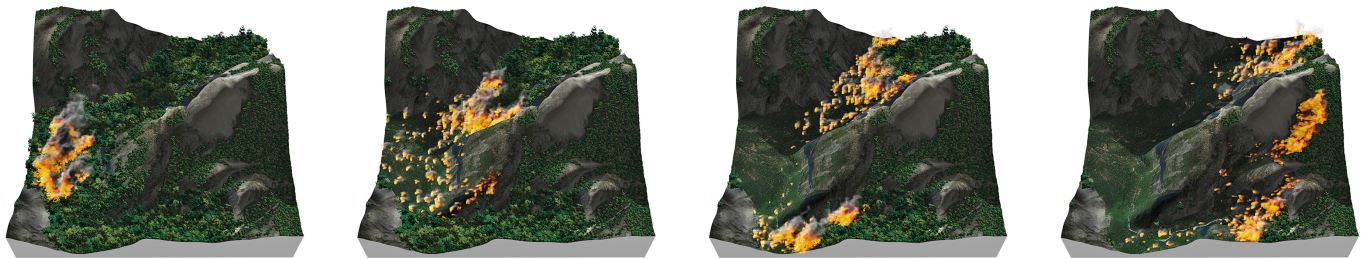


Fig. 1. Simulation of a wildfire spreading in the valley around Half Dome in California’s Yosemite National Park. Using our framework, this scene can be simulated at interactive rates allowing the user to conveniently explore the wildfire.

Resulting from changing climatic conditions, wildfires have become an existential threat across various countries around the world. The complex dynamics paired with their often rapid progression renders wildfires an often disastrous natural phenomenon that is difficult to predict and to counteract. In this paper we present a novel method for simulating wildfires with the goal to realistically capture the combustion process of individual trees and the resulting propagation of fires at the scale of forests. We rely on a state-of-the-art modeling approach for large-scale ecosystems that enables us to represent each plant as a detailed 3D geometric model. We introduce a novel mathematical formulation for the combustion process of plants – also considering effects such as heat transfer, char insulation, and mass loss – as well as for the propagation of fire through the entire ecosystem. Compared to other wildfire simulations which employ geometric representations of plants such as cones or cylinders, our detailed 3D tree models enable us to simulate the interplay of geometric variations of branching structures and the dynamics of fire and wood combustion. Our simulation runs at interactive rates and thereby provides a convenient way to explore different conditions that affect wildfires, ranging from terrain elevation profiles and ecosystem compositions to various measures against wildfires, such as cutting down trees as firebreaks, the application of fire retardant, or the simulation of rain.

¹The title ‘Fire in Paradise’ is chosen in memory of the Camp Fire that devastated the town Paradise in Northern California’s Butte County, November 8–25, 2018, resulting in more than 80 fatalities. A 2019 documentary film directed by Zackary Canepari and Drea Cooper carries a similar title.

Authors’ addresses: Torsten Hädrich, KAUST, Visual Computing Center, Thuwal 23955, KSA; Daniel T. Banuti, UNM, Mechanical Engineering, Building #122, Albuquerque, NM 87131, USA; Wojtek Pałubicki, UAM, Umultowska 87, 61-614 Poznań, Poland; Sören Pirk, Google AI, 1600 Amphitheatre Parkway, Mountain View, CA, 94043, USA; Dominik L. Michels, KAUST, Visual Computing Center, Thuwal 23955, KSA.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

© 2021 Copyright held by the owner/author(s).
0730-0301/2021/8-ART163

<https://doi.org/10.1145/3450626.3459954>

CCS Concepts: • **Computing methodologies** → **Physical simulation**.

Additional Key Words and Phrases: Combustion, Fire, Fluid Dynamics, Level of Detail, Numerical Simulation, Physics-based Modeling, Wildfires.

ACM Reference Format:

Torsten Hädrich, Daniel T. Banuti, Wojtek Pałubicki, Sören Pirk, and Dominik L. Michels. 2021. Fire in Paradise¹: Mesoscale Simulation of Wildfires. *ACM Trans. Graph.* 40, 4, Article 163 (August 2021), 15 pages. <https://doi.org/10.1145/3450626.3459954>

1 INTRODUCTION

In recent years, climate change has facilitated an inexorable increase of bigger and more intense wildfires across the globe. Understanding the complex interplay of fires and large-scale ecosystems plays a key role in preventing wildfires and in containing them. To this end, we argue that simulating wildfires with detailed geometric models of terrain and vegetation along with physically plausible fluid dynamics can serve as an essential tool for understanding wildfires and for predicting their outcome. However, realistically simulating wildfires in different ecosystems, also considering the wide range of geometric compositions of trees and plants, their inhomogeneous material properties, as well as the interaction of a fire and the atmosphere, is a challenging and open problem.

While a wide range of methods exists to plausibly model branching structures [Měch and Prusinkiewicz 1996; Pałubicki et al. 2009; Pirk et al. 2012b; Stava et al. 2014], only very recently methods also focus on the realistic simulation of dynamic behavior and physics response of plant models, including the simulation of growth [Longay et al. 2012], surface adaptation [Hädrich et al. 2017], the interaction with wind [Pirk et al. 2014], or based on realistic material properties [Wang et al. 2013; Zhao and Barbič 2013]. Previous work has combined ecosystem and terrain erosion simulation for authoring landscapes [Cordonnier et al. 2017]. This avenue of research has

been expanded modeling large-scale ecosystems [Kapp et al. 2020; Makowski et al. 2019], and terrain features, such as avalanches [Cordonnier et al. 2018] or glaciers [Argudo et al. 2020]. Together, these methods provide a testament that efforts trend toward physically plausible and specialized approaches to simulate natural phenomena.

Most of the current methods for simulating combustion processes do not specifically focus on tree or wood combustion and therefore cannot be easily applied to models of trees and plants [Melek and Keyser 2002]. Methods in other research disciplines, such as material sciences or forestry, specifically focus on wildfires or the resistance of trees to fires. However, these methods are often computationally demanding and only focus on the combustion of wood samples in laboratory setups [Thi et al. 2016] or employ severely simplified geometric representations of trees and plants [Seidl et al. 2012], such as a suspended cloud of spherical Lagrangian particles that represent either foliage or wood [Mendoza et al. 2019]. Closest to our work is the method of Pirk et al. [2017], who discretize branches as triangular surface meshes that enable the simulation of tree combustion with an astounding degree of detail for complex branching structures at interactive rates. However, while their work focuses on the combustion of individual tree models, we aim to simulate wildfires at forest scale, which cannot be realized with their representation.

In this paper, we advance the field of wildfire simulations by introducing a novel mathematical formulation that allows us to simulate the combustion of trees at an intermediate scale using detailed geometric models. We employ the method of Makowski et al. [2019] to simulate ecosystems. Each tree model is composed of a number of self-organizing branch templates that define its 3D branching structure. Collections of trees can grow together, which results in diverse and realistic branching structures for individual tree models in the ecosystem, while each module is reused across the same tree and for all other trees, which enables efficient modeling and rendering. An advantage of a module-based tree representation is that it provides a convenient way to control the level of detail for representing trees. A tree can either be represented by a large number of very detailed modules, which allows us to generate complex and highly realistic branching structures, or – to the opposite effect – by only a few coarser modules to represent each tree in a lightweight and thus more efficient manner.

To simulate tree combustion we use this module-based representation for trees in two ways: (1) we simulate the combustion at the branch level for each module. This allows us to capture various effects necessary to realistically simulate the combustion of individual branches, including char insulation, mass loss, and heat transfer; (2) we compute the combustion of wood – also known as pyrolysis – across the entire tree at module-scale. A collection of modules that represent a tree model is defined as a directed graph. Once the combustion of a single module progresses toward an adjacent module, the combustion is propagated to this module and continued for this module’s branching structure.

Our goal is to jointly simulate fire and the combustion of large collections of plants – a computationally demanding undertaking. To capture the spread of fire across the entire ecosystem we use a volumetric grid-based fluid solver that enables us to transfer heat

from the environment to individual plants. A burning plant – in turn – releases heat to its environment, which triggers a feedback loop that maintains the combustion and that may cause the spread of fire from one plant to another. A key advantage of our wildfire model is that the combustion of plant tissue and the simulation of fire are decoupled: Trees can be represented with a varying number of modules, while fire can be computed with more or less detailed volumetric grids. This allows us to manage the complexity required for wildfire simulations, while maintaining the realistic and physically plausible interaction of trees and fire. An example of a complex wildfire simulation is shown in Figure 1. In this contribution, we do not address fire spread on the ground facilitated by grass, branch litter, and undergrowth vegetation. Moreover, the role of leaves is ignored and the modeling of sparks flying through the air is left for future work.

In summary, our contributions are as follows: (1) we introduce a novel combustion model for individual trees based on branch modules that allows us to realistically simulate wood pyrolysis; (2) we propose a hybrid model capturing heat transfer between individual branch modules and the environment allowing to appropriately capture fire spread; (3) we capture cloud and rain phenomena within our wildfire simulator by extending the Kessler model; (4) we simulate wildfires of more than 100K individual plants represented by complex and detailed geometry; (5) we show that our interactive framework enables us to explore the emergence of wildfires in ecosystems of different composition and ways to counteract the spread of fire.

2 RELATED WORK

With our goal to simulate wildfires for individual and detailed models of trees our work is related to methods that aim at generating complex and realistic models of terrains, the modeling of vegetation, as well as the simulation of fire or – more generally – fluid dynamics. While this spans a breath of work that we cannot conclusively discuss, our goal is to provide an overview for these research directions with a focus on tree and terrain modeling.

Modeling Trees and Plants. Many of the early approaches for modeling trees and plants have focused on defining the internal properties of trees, such as branching angles and internode lengths to model branching structures [Aono and Kunii 1984; Kawaguchi 1982; Oppenheimer 1986; Smith 1984]. Later, biologically plausible methods were introduced that allow us to model the many variants of tree form in more nuanced and principled ways [Bloomenthal 1985; Weber and Penn 1995] and based on defining the developmental process of plants [de Reffye et al. 1988]. Furthermore, L-systems [Prusinkiewicz 1986] and rule-based techniques [Lintermann and Deussen 1999] have been recognized as powerful modeling approaches for diverse shapes of trees and plants.

To further increase the realism, a few methods also aim at modeling the environmental response of plants during their development [Měch and Prusinkiewicz 1996; Palubicki et al. 2009; Pirk et al. 2012b; Stava et al. 2014]. Besides the forward modeling of branching structures, reconstructing trees and plants based on images [Argudo et al. 2016; Neubert et al. 2007; Quan et al. 2006; Reche-Martinez et al. 2004; Tan et al. 2008] or point clouds [Livny et al. 2011; Xu et al.

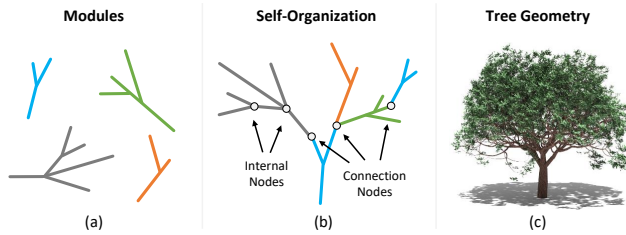


Fig. 2. We use a module-based representation for plants. Each plant is defined as a combination of modules (a). Modules are adapted through self-organization during ecosystem development and are reused across the same plant and the entire ecosystem (b). Once the branch graph has been defined we generate the final plant geometry (c).

2007] also provides a convenient alternative to capture complex plant form. Sketch-based approaches on the other hand enable the refined generation of tree models, while also supporting artistic requirements toward content creation [Chen et al. 2008; Longay et al. 2012; Okabe et al. 2007; Wither et al. 2009]. More recently, a number of methods simulate the physics-response and the dynamics of tree models, including the swaying of trees in wind fields [Habel et al. 2009; Pirk et al. 2014], the interactive modeling of growth [Hädrich et al. 2017; Pirk et al. 2012a], or the simulation of tree dynamics based on physically plausible materials [Wang et al. 2013; Zhao and Barbič 2013] or through machine learning-assisted iterative solvers [Shao et al. 2021].

Terrain Models and Plant Ecosystems. Generating detailed models of complex terrain has been extensively studied in computer graphics [Fournier et al. 1982; Kelley et al. 1988]. Early approaches for modeling photo-realistic terrains mostly focus on generating complex natural landscapes by employing fractals [van Lawick van Pabst and Jense 1996], noise functions [Perlin 1985], or procedural models [Ebert et al. 2002]. For plant ecosystems, existing methods not only aim at finding ways to compute realistic distributions of various species [Deussen et al. 2002, 1998; Lane and Prusinkiewicz 2002], but also to identify representations for ecosystems that enable modeling and rendering at scale; methods range from voxels [Jaeger and Teng 2003] and volumetric textures [Bruneton and Neyret 2012] to layers [Argudo et al. 2017] and branch templates [Makowski et al. 2019]. To support the design and content creation of terrain and ecosystems a number of methods also explore sketch-based interfaces in conjunction with biological priors [Beneš et al. 2009]. We refer to the recent survey by Galin et al. [2019] for a more detailed overview on terrain modeling. It is worth pointing out that real-world data and machine learning has been leveraged using generative adversarial networks trained by real-world terrains and their sketched counterparts [Guérin et al. 2017], or by deriving a canopy height model combined with an understory layer resulting in realistic ecosystems [Kapp et al. 2020].

Due to the enormous amount of geometry required to realistically generate plant ecosystems a number of methods focus on level of detail strategies. Prominent examples include point and line representations [Deussen et al. 2002; Stamminger and Dretakis 2001], billboard clouds [Behrendt et al. 2005], or stochastic

simplification [Cook et al. 2007; Neubert et al. 2011]. To efficiently model large-scale ecosystems, we employ the method of Makowski et al. [2019] that represents trees as collections of branch modules that can be efficiently instantiated to model and render large collections of plants, while the full branch geometry of individual tree models is retained.

Simulating Fire and Combustion. The computation of fluid dynamics as required for simulating fire has a long tradition in computer graphics research [Bridson 2008]. Most approaches rely on grid-based fluid solvers to capture turbulence as one of the predominant features of fire [Hong et al. 2010; Nguyen et al. 2002; Stam 1999] or smoke [Fedkiw et al. 2001; Pan and Manocha 2017; Rasmussen et al. 2003]. Furthermore, a number of methods explicitly focus on rendering fires either based on physically-accurate models [Nguyen et al. 2002; Pegoraro and Parker 2006] also with respect to specific flame properties [Nguyen et al. 2001], with an emphasis on artistic control [Lamorlette and Foster 2002], or based on combined representations that also use particles to simulate turbulence [Horvath and Geiger 2009].

Similar to simulating fire, the process of combustion is often modeled based on planar or volumetric grids that enable to not only model the distribution of heat [Melek and Keyser 2002] on surfaces [Chiba et al. 1994] or in volumes [Zhao et al. 2003], but also to simulate fire across disconnected propagating fronts [Liu et al. 2012]. Simulating combustion and heat diffusion for articulated and continuously defined surface geometry remains a challenging problem, which is only addressed by a few methods [Hong et al. 2010]. Material point methods, on the other hand, have recently gained popularity capturing thermodynamic properties to simulate phenomena such as the melting or solidifying of materials [Stomakhin et al. 2014]. However, most of these methods are not defined to simulate wood combustion at ecosystem scale.

Wood Combustion and Wildfires. In forestry, botany, and material science a substantial amount of work focuses on the combustion of wood and plants. Existing methods range from simulating heat transfer [Encinas et al. 2007], charring [Lizhong et al. 2002], or the pyrolysis process of entire trees and plants [Bohren and Thorud 1973]. A key factor for understanding the propagation of fire in forests is the fire resistance of plants. Hence, a number of approaches aim at modeling the resistance of individual species [Lawes et al. 2011], the impact of canopy architecture on flammability [Schwilk 2003], or the moisture content of plant material [Masinda et al. 2020]. A large body of work focuses on simulating wildfires, often with the goal to establish predictive models [Monedero et al. 2017; Pastor et al. 2003; Richards 1990], to simulate fires for different biomes [Cheney et al. 1993; Dupuy and Larini 2000], to understand smoke properties and the ignition of wildfires [Anand et al. 2017; Gustenyov et al. 2018], to predict high-fidelity flows around strongly simplified trees [Mendoza et al. 2019], or by specifically focusing on the coupling of wildfires and the atmosphere [Coen 2005; Sun et al. 2009]. Finally, researchers also investigate the long-term growth response of vegetation to wildfires [Chileen et al. 2020]. Similar to our work, many of these approaches aim at defining accurate models for wood combustion or physically-accurate solvers for wildfires. However, unlike these methods, we simulate wood combustion for

detailed geometric models of trees, which enables us to explore the impact of vegetation geometry on the propagation of wildfires.

3 OVERVIEW

The main motivation for our approach is to realistically model and simulate wildfires at forest scale. This objective is challenging because of two main reasons. First, modeling realistic large-scale scenes of vegetation commonly requires the generation of an enormous amount of geometric detail. Second, computing the pyrolysis of wood at branch-level, while also coupling the combustion process with a solver for fluid dynamics to model the spread of fire is computationally demanding and commonly not performed jointly.

We address these challenges by employing a multi-scale representation for vegetation that uses branch modules to represent plants [Makowski et al. 2019]. Each plant is defined as a collection of branch modules that locally adapt as a result of the plant's development and its interaction with neighboring plants – a process that results in individual and highly detailed branching structures. A key advantage of this representation is that modules can be instantiated and reused across the same plant and for other plants in the ecosystem (Figure 2).

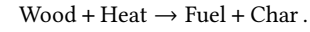
We use this module-based representation to define a novel combustion model for plants. Unlike other approaches that define pyrolysis at the scale of mesh elements [Pirk et al. 2017], we define the combustion of wood for individual branch modules. This level of abstraction has two major advantages. For one, the modules allow us to define the level of detail at which we want to model wildfires. Trees can be represented with varying degrees of detail, ranging from only a few coarse modules – where each module only represents a few branch segments – to a large number of detailed modules that result in branching structures with a high degree of visual fidelity. Second, computing the combustion at the level of modules allows us to retain the geometric structure of individual plants, while we can simultaneously process large collections of trees. A detailed geometric representation is important to realistically simulate the propagation of fire within a single tree as well as across an entire forest.

Finally, we employ two models for fluid dynamics to simulate fire, smoke, and clouds. We employ an Eulerian fluid solver to simulate fire and to model its propagation through the ecosystem. This way, fire can be transferred from module to module and – in turn – from tree to tree. Second, we use a state-of-the-art model for cloud dynamics to simulate so called *flammagenitus clouds* that emerge from large-scale wildfires as shown in Figure 19. For real wildfires, fire clouds often play a critical role as they occlude the fire and thereby hinder taking measures against fire spread. Therefore, simulating fire clouds greatly adds to the realism of wildfire simulations.

In summary, we model wildfires by coupling a module-based representation for vegetation with a novel combustion model that operates at the scale of individual branch modules. This is combined with state-of-the-art fluid solvers for fire, smoke, and cloud dynamics in which we incorporate novel formulations capturing heat transfer between tree modules and the environment. An overview of our framework is provided in Figure 3.

4 METHODOLOGY

Combustion of solid fuel starts when it is exposed to heat as soon as its ignition temperature is reached. Wood is decomposed into char and flammable gases (fuel), i.e.,



As explained by Pirk et al. [2017], the rate of mass change dM/dt can be described by

$$\frac{dM}{dt} + k(T_M) c A = 0, \quad (1)$$

in which k denotes the reaction rate which is dependent on the temperature T_M . The dimensionless char insulation parameter is denoted by c and the pyrolyzing front area by A . Both, c and A depend on the tree geometry and vary during the combustion process. In Pirk et al. [2017], a reaction rate

$$k(T_M) = \eta \cdot \begin{cases} 0 & T_M < T_0, \\ S((T_M - T_0)/(T_1 - T_0)) & T_0 \leq T_M \leq T_1, \\ 1 & T_M > T_1, \end{cases} \quad (2)$$

is applied with constant η . The function $S : x \mapsto 3x^2 - 2x^3$ describes a sigmoid-like function interpolating smoothly from zero to one for temperatures between $T_0 = 150^\circ\text{C}$ and $T_1 = 450^\circ\text{C}$. We include an extension to the reaction rate to take into account wind speed u described by a function

$$\eta(u) = (\eta_{\max} - 1) S(u/u_{\text{ref}}) + 1. \quad (3)$$

The function output corresponds to $\eta = 1$ in cases without wind and to $\eta = \eta_{\max} \geq 1$ for a threshold velocity u_{ref} at which a maximum boost is reached. Consequently, blowing wind increases the reaction rate and heat release, so that firestorms potentially emerge. This covers the common observation that blowing into the fire increases its temperature and accelerates the combustion process as the oxygen concentration becomes higher.

4.1 Tree Representation

Unlike Pirk et al. [2017], who discretized branches as triangular surface meshes and defined pyrolysis as a propagating front that transforms virgin wood into char coal by moving towards the branch axis, we introduce a higher level of abstraction by representing branches as truncated cones.

Given a branch of length l represented as a truncated cone with radii $r := r^{(0)}$ and $r' := r^{(1)}$, its lateral surface area can be computed as

$$A_{\text{branch}} = \pi (r + r') \sqrt{(r - r')^2 + l^2}, \quad (4)$$

its volume by

$$V_{\text{branch}} = \frac{\pi}{3} l (r^2 + rr' + (r')^2), \quad (5)$$

and its mass by

$$M_{\text{branch}} = \rho_{\text{wood}} V_{\text{branch}}, \quad (6)$$

in which ρ_{wood} denotes the wood's density.

We simulate combustion on a module-level, which reduces the complexity and – in turn – enables us to simulate wildfires. A module \mathcal{M} is composed of a set of adjacent truncated cones. A tree \mathcal{T} is defined as a set of connected modules and can consequently be decomposed into different numbers of modules depending on the

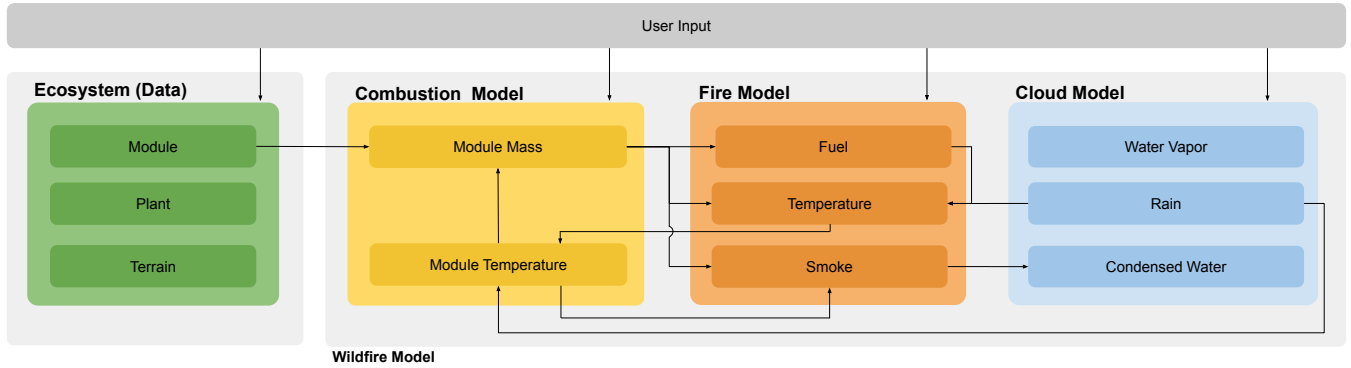


Fig. 3. Our wildfire model expresses plant shape using a multi-scale geometric representation of branch modules and plants; terrain is represented by a surface mesh. Together, this constitutes the input to our combustion model that describes pyrolysis in trees. This process releases fuel, temperature, and smoke, which are expressed by transport equations in our fire model. Additionally, we also model the water cycle in the atmosphere and couple it with the fire and combustion model. This allows us to capture the feedback between tree combustion, fire spread, and cloud formation.

desired level of detail. A forest $\mathcal{F} := \bigcup_{i=1}^n \mathcal{T}$ containing n trees is defined as the union of these trees.

4.2 Module-level Combustion

Considering a single branch, the mass loss rate dM/dt (or ΔM within a discrete time step Δt) has to be computed according to Eq. (1) with $A = A_{\text{branch}}$ (see Eq. (4)). The branch's radii r and r' have to be adjusted based on ΔM . We enforce a fixed radii ratio $\lambda := r'/r = \text{const}$. The combustion process can then be described as a mapping

$$(r_t, \Delta M) \mapsto r_{t+\Delta t}(r_t, \Delta M),$$

in which $r_{t+\Delta t}$ denotes the updated radius of r_t according to the mass loss rate ΔM . Following Eq. (5–6), we obtain

$$(r_t, \Delta M) \mapsto r_{t+\Delta t} = \sqrt{\frac{3(M + \Delta M)}{\pi l \rho} \frac{1}{1 + \lambda + \lambda^2}}, \quad (7)$$

in which M denotes the (original) mass before the radii update. Please note that $\Delta M \leq 0$. The updated radius $r'_{t+\Delta t}$ of r'_t can be computed by enforcing the radii ratio:

$$r'_{t+\Delta t} := \lambda r_{t+\Delta t}(r_t, \Delta M). \quad (8)$$

As combustion is simulated on a per-module basis, the area A has to be computed as the sum of branch surface areas. The change of mass ΔM due to combustion then has to be distributed among the branches by updating the corresponding radii. Naturally, mass and radii of an individual branch can not be decreased below zero.

4.2.1 Intra-module Radii Update. Let $\mathcal{M} \in \mathcal{F}$ be a module. It can be described by a graph $G_{\mathcal{M}} = (V, E)$ in which V denotes the set of nodes and $E \subseteq V \times V$ the set of edges. As modules are rooted and connected, grow in a specific direction, and do not contain cycles, G is an arborescence, i.e., a directed rooted tree with root node $v^{(0)} \in V$. For all $v^{(i)} \in V$, we define a corresponding radius $r : V \rightarrow \mathbb{R}^{\geq 0}$, $v^{(i)} \mapsto r^{(i)}$, and for all $e^{(i)} = (v, v') \in E$, we define a branch length $l : E \rightarrow \mathbb{R}^{\geq 0}$, $e^{(i)} \mapsto l^{(i)}$ and a radii ratio $\lambda : E \rightarrow \mathbb{R}^{\geq 0}$, $e^{(i)} \mapsto \lambda^{(i)} := r(v')/r(v)$. Please note that usually $\lambda \leq 1$ according to da Vinci's rule of trees [Minamino and Tateno 2014]. This is illustrated in Figure 4.

For each edge $e \in E$, we can compute the mass $M(e)$ of the corresponding branch according to Eq. (5–6). The total mass $M(\mathcal{M})$ of the module \mathcal{M} is then given by $M(\mathcal{M}) = \sum_{e \in E} M(e)$. Using subscript index notation to indicate time, we can write

$$M(\mathcal{M}_{t+\Delta t}) = \sum_{e \in E_{t+\Delta t}} M(e) \stackrel{!}{=} M(\mathcal{M}_t) + \Delta M, \quad (9)$$

which states that the total mass of the module after the radii update $M(\mathcal{M}_{t+\Delta t})$ and before the update $M(\mathcal{M}_t)$ differ by ΔM . Using Eq. (5–6), we can rewrite Eq. (9) as follows:

$$\begin{aligned} M(\mathcal{M}_t) + \Delta M &= \frac{\rho\pi}{3} \sum_{\substack{e \in E_{t+\Delta t} \\ e := (v, v')}} l(e) \left(r^2(v) + r(v)r(v') + r^2(v') \right) \\ &= \frac{\rho\pi}{3} r^2(v^{(0)}) \sum_{\substack{e \in E_{t+\Delta t} \\ e := (v, v')}} l(e) \prod_{\bar{e} \in P(v)} \left(\lambda^2(\bar{e}) \right) \left(1 + \lambda(e) + \lambda^2(e) \right). \end{aligned} \quad (10)$$

The path from the root node v^0 to v is denoted with $P(v)$. As G is an arborescence, the path $P(v)$ is uniquely defined for all $v \in V \setminus \{v^{(0)}\}$ and $P(v^{(0)}) := \emptyset$.

The radii update formula for the root node $v^{(0)}$ within a module \mathcal{M} now follows directly from Eq. (10):

$$(r_t^{(0)}, \Delta M) \mapsto r_{t+\Delta t}^{(0)} := \sqrt{\frac{3}{\pi\rho}} \psi_{\mathcal{M}} \sqrt{M(\mathcal{M}_t) + \Delta M} \quad (11)$$

using a module constant

$$\psi_{\mathcal{M}} := \left(\sum_{e := (v, v') \in E} l(e) \prod_{\bar{e} \in P(v)} \left(\lambda^2(\bar{e}) \right) \left(1 + \lambda(e) + \lambda^2(e) \right) \right)^{-1}.$$

For all $v \in V \setminus \{v^{(0)}\}$, we apply

$$r_{t+\Delta t}^{(i)} := \prod_{\bar{e} \in P(v^{(i)})} \left(\lambda(\bar{e}) \right) r_{t+\Delta t}^{(0)}. \quad (12)$$

If the module \mathcal{M} consists entirely of a single branch, Eq. (11–12) can be simplified to the special case Eq. (7–8).

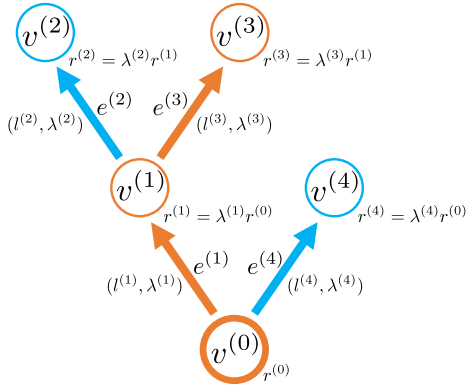


Fig. 4. Illustration of the representation of a module \mathcal{M} by an arborescence $G_{\mathcal{M}} = (V, E)$ with nodes $V = \{v^{(0)}, v^{(1)}, v^{(2)}, v^{(3)}, v^{(4)}\}$, edges $E = \{e^{(1)} := (v^{(0)}, v^{(1)}), e^{(2)} := (v^{(1)}, v^{(2)}), e^{(3)} := (v^{(1)}, v^{(3)}), e^{(4)} := (v^{(0)}, v^{(4)})\}$, and root node $v^{(0)}$. Node features $\{r^{(0)}, r^{(1)}, r^{(2)}, r^{(3)}, r^{(4)}\}$ and edge features $\{(l^{(1)}, \lambda^{(1)}), (l^{(2)}, \lambda^{(2)}), (l^{(3)}, \lambda^{(3)}), (l^{(4)}, \lambda^{(4)})\}$ are shown. The unique path $P(v^{(3)}) := (e^{(1)}, e^{(3)})$ from the root node $v^{(0)}$ to $v^{(3)}$ is highlighted here as an example. The depth-first search order is used for node numbering.

4.2.2 Connection Node Handling. As trees are represented by several modules attached to each other, we distinguish between internal and connection nodes within a module. Connection nodes are those which are shared by different modules (Figure 2). All other nodes are labeled as internal nodes. For connection nodes, we store multiple radii information, i.e., every module sharing a connection node, stores its own radius at the connection node and updates it only with respect to its own mass loss rate ΔM independently from the other modules. However, for rendering a single radius is needed at each node. For this, we simply use the average radii for connection nodes.

4.2.3 Charring Effect. The char insulating parameter c has to be set proportional to the char layer thickness according to Pirk et al. [2017]. The char layer thickness is assumed to be the same for all branches within the module.

4.3 Wind

Wind is described as a time-dependent vector-valued velocity field

$$\mathbf{u} : (\mathbf{x}, t) \mapsto \mathbf{u}(\mathbf{x}, t)$$

which for given time $t \in \mathbb{R}^{\geq 0}$ and position $\mathbf{x} \in \mathbb{R}^3$ returns the corresponding local flow $\mathbf{u}(\mathbf{x}, t) \in \mathbb{R}^3$. The temporal evolution of \mathbf{u} is described by the Navier-Stokes equation [Bridson 2008]

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} + \frac{1}{\rho_{\text{air}}} \nabla p = \nu \nabla \cdot \nabla \mathbf{u} + \mathbf{b} + \mathbf{f}_d + \mathbf{f} \quad (13)$$

covering the chance of momentum, and the continuity equation

$$\nabla \cdot \mathbf{u} = 0 \quad (14)$$

ensuring the conservation of mass. In Eq. (13), the air density is denoted by ρ_{air} , the pressure by p , and the kinematic viscosity by ν . The second term on the left side of Eq. (13) describes phenomena caused by advection followed by the pressure term. The first term on the right side describes viscosity. Buoyancy is taken into account

by the force $\mathbf{b} \in \mathbb{R}^3$. Moreover, trees are influencing the wind which is captured using a drag force

$$\mathbf{f}_d = \frac{\rho_{\text{air}}}{2} c_d A_{cs} \|\mathbf{u}\|^2 \frac{\mathbf{u}}{\|\mathbf{u}\|} \quad (15)$$

with drag coefficient c_d and cross sectional area A_{cs} . All other external forces are combined and described by an additional external net force $\mathbf{f} \in \mathbb{R}^3$. Please note that the wind speed $u := \|\mathbf{u}\|$ influences the reaction rate k of the combustion process directly as stated in Eq. (2-3).

4.4 Smoke

We take smoke into account according to Pirk et al. [2017] who added smoke proportionally to the mass loss and the evaporation of water. Defining a time-dependent scalar field

$$q_s : (\mathbf{x}, t) \mapsto q_s(\mathbf{x}, t)$$

describing the smoke density $q_s \in \mathbb{R}^3$ at given time $t \in \mathbb{R}^{\geq 0}$ and position $\mathbf{x} \in \mathbb{R}^3$, its temporal evolution can be described by

$$\frac{\partial q_s}{\partial t} + \mathbf{u} \cdot \nabla q_s = -s_M \frac{dM}{dt} - s_W \frac{dW}{dt} \quad (16)$$

with smoke parameters s_M and s_W . Pirk et al. [2017] computed the value of the water content W by $dW/dt = -w_T T_M$ applying an evaporation rate w_T . This is problematic because too much water could also get released if the module is not burning at all. Only burning wood significantly releases water vapor into the air when the hydrogen content of the wood binds with atmospheric oxygen. With a hydrogen mass fraction in wood of 6% [Côté 1968], $c_{WM} = 0.5362$ kg of water are released per 1 kg of burned wood when two moles of hydrogen bind with a mole of oxygen. Consequently, we make use of the relation

$$\frac{dW}{dt} = c_{WM} \frac{dM}{dt}. \quad (17)$$

4.5 Clouds and Rain

Next to the smoke density q_s , we define similar time-dependent scalar fields for water vapor $q_v : (\mathbf{x}, t) \mapsto q_v(\mathbf{x}, t)$, condensed water $q_c : (\mathbf{x}, t) \mapsto q_c(\mathbf{x}, t)$, and rain $q_r : (\mathbf{x}, t) \mapsto q_r(\mathbf{x}, t)$, which for given time $t \in \mathbb{R}^{\geq 0}$ and position $\mathbf{x} \in \mathbb{R}^3$ return the corresponding densities. The outputs are dimensionless quantities which we obtain as mass mixing ratios ($[q] = 1 \text{ kg/kg}$) describing the mass of vapor, smoke, or liquid per unit mass of air. Please note that q_c corresponds to what we usually observe as visible clouds. Following Hädrich et al. [2020], we model the relationship between q_v , q_c , and q_r using Kessler's methodology [1969] including a transport equation for a rain phase in addition to vapor and cloud. The transport equations are coupled using the system of differential equations

$$\frac{\partial q_v}{\partial t} + \mathbf{u} \cdot \nabla q_v = -C_c + E_c + E_r + V_r - s_W \frac{dW}{dt}, \quad (18)$$

$$\frac{\partial q_c}{\partial t} + \mathbf{u} \cdot \nabla q_c = C_c - E_c - A_c - K_c, \quad (19)$$

$$\frac{\partial q_r}{\partial t} + \mathbf{u} \cdot \nabla q_r = A_c + K_c - E_r - V_r, \quad (20)$$

and source terms C_c for condensation, E_c for cloud evaporation, E_r for rain evaporation, A_c for autoconversion of raindrops from clouds, and K_c for the accretion of cloud water due to falling drops.

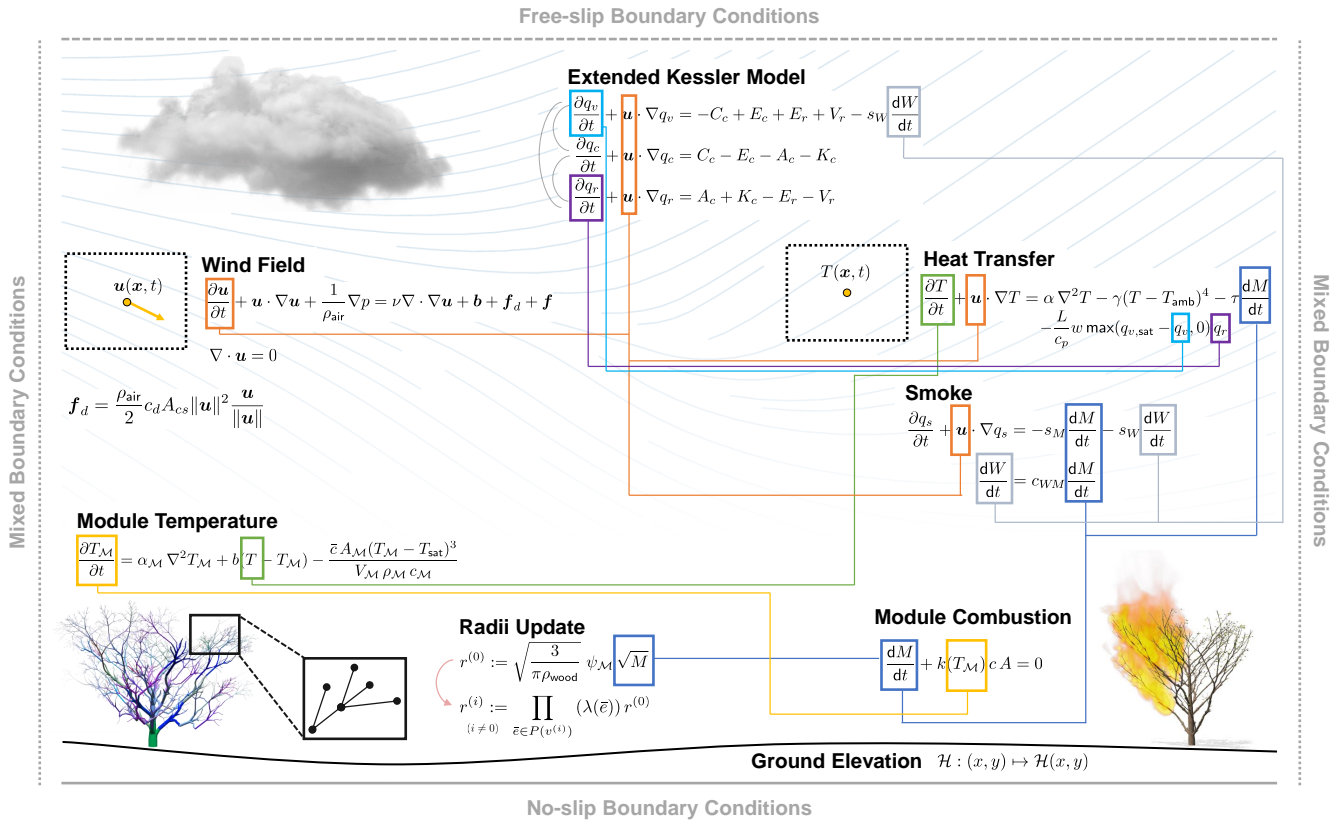


Fig. 5. Overview of our model explicitly showing dependencies of the different quantities. The grid resolution corresponds to the size of the bounding box of an average module. Please note that the temperature of air is used within the computations of the extended Kessler model. The outcome of the Kessler model influences the wind field as well. Wind is furthermore directly influencing the combustion process as its reaction rate depends on the wind speed. The combustion also depends on the updated radii as this influences the tree geometry.

We set E_r according to Eq. (22) and add a vaporization sink term V_r given by Eq. (31) as derived in the upcoming Section 4.6. Moreover, following Harris et al. [2003], we apply $E_c - C_c = \min(q_{v,\text{sat}} - q_v, q_c)$, and according to Hädrich et al. [2020] $A_c = \beta_A \max(q_c - a_T, 0)$ and $K_c = \beta_K q_c q_r$ with constant β_A , β_K , and a_T . Similarly to Eq. (16), we add a term proportional to the change of water content W to Eq. (18) in order to add the moisture released by the combustion process to the water vapor.

4.6 Heat Transfer

The environmental temperature is described as a time-dependent scalar field

$$T : (\mathbf{x}, t) \mapsto T(\mathbf{x}, t),$$

which for given time $t \in \mathbb{R}^{\geq 0}$ and position $\mathbf{x} \in \mathbb{R}^3$ returns the corresponding environmental temperature $T(\mathbf{x}, t) \in \mathbb{R}$. Heat transfer is modeled using the following relationship describing the temporal evolution of the air temperature T :

$$\frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T = \alpha \nabla^2 T - \gamma (T - T_{\text{amb}})^4 - \tau \frac{dM}{dt} - \varphi q_r. \quad (21)$$

The temporal temperature change of a certain fluid parcel, as it flows along the trajectory of the wind, is described by a diffusion component with intensity α , and an ambient cooling component with the radiative cooling term γ [Nguyen et al. 2002] involving the fixed ambient temperature

$$T_{\text{amb}} := T_{\text{amb}}(h) \text{ at altitude } h := h(\mathbf{x})$$

for a given position $\mathbf{x} := (x, y, h)$ evaluated according to atmospheric data [ISO 1975]. Moreover, module to air heat transfer is addressed by adjusting the air temperature proportionally to the mass loss with coefficient τ . The density of rain q_r is taken into account to address the cooling effect of rain. Two physical effects cause this cooling: Heat absorption into cool rain drops and evaporation. As evaporation dominates, we will focus on this effect here. The underlying physical mechanism is that water molecules leaving the rain drops have to overcome the attraction of the other water molecules, and thus cause a decrease in temperature. The driving potential of this evaporation process is the relative humidity $\phi_{\text{rel}} = q_v / q_{v,\text{sat}}$ as the ratio of the local vapor mass ratio q_v and the local saturation vapor mass ratio $q_{v,\text{sat}}$ [Hädrich et al. 2020] of the surrounding air. When the air is saturated, i.e. $q_v = q_{v,\text{sat}}$ or $\phi_{\text{rel}} = 1$, evaporation

stops and no cooling occurs. This results in a rain evaporation rate E_r in the form

$$E_r = q_r w \max(q_{v,\text{sat}} - q_v, 0), \quad (22)$$

where w is an evaporation rate coefficient, and the saturation vapor mixing ratio is given by

$$q_{v,\text{sat}}(T, p) = \frac{380.16}{p} \exp\left(\frac{17.67T}{T + 243.50}\right) \quad (23)$$

for given temperature and pressure in Celsius respectively Pascal [Yau and Rogers 1996]. The evaporation adsorbs the latent heat L which acts to cool the surrounding air of heat capacity c_p , such that we find an approximate expression for φ of Eq. (21):

$$\varphi = \frac{L}{c_p} \frac{E_r}{q_r} = \frac{L}{c_p} w \max(q_{v,\text{sat}} - q_v, 0). \quad (24)$$

Next to the environmental temperature field T , we introduce a module temperature function

$$T_{\mathcal{F}} : (\mathcal{M}, t) \mapsto T_{\mathcal{F}}(\mathcal{M}, t) =: T_{\mathcal{M}}(t),$$

which for given time $t \in \mathbb{R}^{\geq 0}$ and module $\mathcal{M} \in \mathcal{F}$ returns the module's surface temperature $T_{\mathcal{M}}(t) \in \mathbb{R}$. The change of $T_{\mathcal{M}}$ is described by simulating diffusion between adjacent modules, as well as the heating (e.g., caused by a fire) or cooling of its surface due to the temperature T of the surrounding air:

$$\frac{\partial T_{\mathcal{M}}}{\partial t} = \alpha_{\mathcal{M}} \nabla^2 T_{\mathcal{M}} + b(T - T_{\mathcal{M}}). \quad (25)$$

Diffusion and temperature coefficients $\alpha_{\mathcal{M}}$ and b are applied. Moreover, we model cooling of the module's surface due to rain, which depends on the heat transfer from the surface of the wood to the covering water film and on the amount of rain. The heat flux to water per unit area can be approximated by

$$\dot{q}'' = \bar{c}(T_{\mathcal{M}} - T_{\text{sat}})^3 \quad (26)$$

using $\bar{c} = 0.1 \text{ Wm}^{-2} \text{ } ^\circ\text{C}^{-1}$ and the saturation temperature of water corresponding to $T_{\text{sat}} = 100^\circ\text{C}$ [Carey 1992]. This energy is extracted from the module and added to the adjacent rain water phase until all of the rain water q_r is vaporized to q_v . As an effect of this, the wood's temperature is decreased. The energy $U_{\mathcal{M}}$ stored in a module \mathcal{M} of volume $V_{\mathcal{M}}$ and density $\rho_{\mathcal{M}}$ with specific heat capacity $c_{\mathcal{M}}$ is given by

$$U_{\mathcal{M}} = V_{\mathcal{M}} \rho_{\mathcal{M}} c_{\mathcal{M}} T'_{\mathcal{M}}, \quad (27)$$

where $T'_{\mathcal{M}}$ denotes the absolute temperature. For wood, we apply $c_{\mathcal{M}} \approx 2.5 \text{ kJ } ^\circ\text{C}^{-1} \text{ kg}$. The temperature rate of change of the module given a change in energy is then described by

$$\frac{dT_{\mathcal{M}}}{dt} = \frac{dU_{\mathcal{M}}}{dt} \frac{1}{V_{\mathcal{M}} \rho_{\mathcal{M}} c_{\mathcal{M}}}, \quad (28)$$

where the change in energy occurs through heat transfer across the module surface $A_{\mathcal{M}}$, i.e.

$$\frac{dU_{\mathcal{M}}}{dt} = \dot{q}'' A_{\mathcal{M}} = \dot{Q} = \bar{c} A_{\mathcal{M}} (T_{\mathcal{M}} - T_{\text{sat}})^3. \quad (29)$$

Based on this, we then extend Eq. (25) to

$$\frac{\partial T_{\mathcal{M}}}{\partial t} = \alpha_{\mathcal{M}} \nabla^2 T_{\mathcal{M}} + b(T - T_{\mathcal{M}}) - \frac{\bar{c} A_{\mathcal{M}} (T_{\mathcal{M}} - T_{\text{sat}})^3}{V_{\mathcal{M}} \rho_{\mathcal{M}} c_{\mathcal{M}}}. \quad (30)$$

Another effect of this heat conduction is that the rain phase is vaporized with a mass rate of $\dot{m}_r = \dot{Q}/L$ until all rain in contact with the module is consumed and turned into vapor. This results in a rain vaporization sink term

$$V_r = \max\left(\frac{\dot{Q}}{L} \rho_{\text{air}}, 0\right) \quad (31)$$

in the presence of a tree, which has to be added to the vapor ratio's transport Eq. (18) and subtracted from the rain ratio's transport Eq. (20).

In summary, our heat model transports a temperature field, thus accounting for convective heat transfer including heat transfer between modules and the atmosphere. Radiative cooling is taken into account [Nguyen et al. 2002] while radiative heating is not included as recent research [Finney et al. 2015] finds that it can be neglected compared to convective heat transfer.

5 ALGORITHMICS

The model described in the previous section provides the basis of our simulator. The whole procedure is summarized in Algorithm 1. Moreover, Figure 5 presents an overview pointing out the dependencies of the different quantities.

ALGORITHM 1: Overview of our simulator's numerical procedure.

*Please note that $\psi_{\mathcal{M}}$ can be precomputed for all $\mathcal{M} \in \mathcal{F}$.

Input: Current system state.

Output: Updated system state.

```

1 for each module  $\mathcal{M} \in \mathcal{F}$  do
2   | Update mass  $M := M(\mathcal{M})$  according to Eq. (1).
3   | Perform radii update according to Eq. (11–12)*.
4   | Update temperature  $T_{\mathcal{M}}$  according to Eq. (30).
5   | Update released water content  $W := W(\mathcal{M})$  according to Eq. (17).
6 end
7 for each grid point  $\mathbf{x} \in \mathcal{D}(\Omega)$  do
8   | Update  $M := M(\mathbf{x})$  and  $W := W(\mathbf{x})$  as described in Section 5.1.
9 end
10 Update temperature  $T$  according to Eq. (21).
11 Update drag forces  $\mathbf{f}_d$  according to Eq. (15).
12 Update  $q_v$ ,  $q_c$ ,  $q_r$ ,  $q_s$ , and  $\mathbf{u}$  according to Hädrić et al. [2020]
    including vorticity confinement with intensity  $\epsilon$ .
13 for each module  $\mathcal{M} \in \mathcal{F}$  do
14   | if  $M := M(\mathcal{M}) = 0$  then  $\mathcal{F} \leftarrow \mathcal{F} \setminus (\{\mathcal{M}\} \cup \text{children}(\mathcal{M}))$ 
15 end

```

5.1 Numerical Procedure

We have to distinguish between those computations acting within the module space \mathcal{F} , and those carried out on a grid discretizing our spatial domain $\Omega \subset \mathbb{R}^3$. As before, we denote modules as $\mathcal{M} \in \mathcal{F}$, and set up a set of grid points denoted as $\mathcal{D}(\Omega)$ discretizing Ω . We apply a single uniform grid scale of Δx . We allow for non-flat ground terrains for which reason a height map $\mathcal{H} : (x, y) \mapsto \mathcal{H}(x, y)$ is introduced defining the lower boundary of Ω :

$$\partial\Omega_{\text{bottom}} := \{(x, y, \mathcal{H}(x, y))^T \in \Omega\}.$$

Table 1. Overview of the different scenes presented in this paper. For each scene, the spatial resolution measured in meters, the number n of trees and the number $N := |\mathcal{F}|$ of modules, the computation time (CT) per forward integration in milliseconds, and the relevant parameters are listed. Each forward integration is carried out using a constant time step size of $\Delta t = 0.016$ s. The parameters are listed in $[\Delta x] = 1$ m, $[\gamma] = 1$ s $^{-1}$ °C $^{-3}$, $[\tau] = 1$ °C kg $^{-1}$, and $[b] = 1$ s $^{-1}$. Identical parameters $u_{\text{ref}} = 15$ m/s, $\eta_{\text{max}} = 2$ kg s $^{-1}$ m $^{-2}$, $\alpha = 0.02$ m 2 s $^{-1}$, $\alpha_{\mathcal{M}} = 0.75$ m 2 s $^{-1}$, $c_d = 1.2$, $s_{\mathcal{M}} = s_W = 0.05$ kg $^{-1}$, and $\epsilon = 0.2$ are used throughout our simulations. The density ρ_{wood} of wood (averaged for different moisture intensities) depends on the specific tree type: $\rho_{\text{deciduous}} = 660$ kg m $^{-3}$ (birches and oak trees), $\rho_{\text{conifer}} = 420$ kg m $^{-3}$ (pine trees and spruces), and $\rho_{\text{shrub}} = 300$ kg m $^{-3}$ have been applied.

Figure	Scene	Resolution	Δx	n	N	CT	γ	τ	b
1	Burning Yosemite	160 3	25.0	68K	75K	82	0.003	200	1.83
6	Vertical Spread	64 2 × 48	0.25	2	0.8K	2	0.003	163	0.75
8	Varying Slope	112 × 64 × 128	2.00	3.4K	12.2K	16	0.003	163	0.75
10	Fire Resistance	160 3	2.00	19.6K	25.7K	75	0.004	200	1.83
13	Wind Effect	160 3	1.75	19.6K	25.7K	75	0.004	200	1.83
14	Forest Cover	160 3	1.75	≤19.6K	≤25.7K	≤75	0.004	200	1.83
15	Fire Extinguishing	64 2 × 48	1.00	78	5K	4	0.003	163	0.75
15	Fire Barrier Zones	112 × 64 × 128	2.00	≤3.4K	≤12.2K	≤13	0.003	163	0.75
18	Flammagenitus	128 × 24 × 92	10	1K	2.5K	6	0.003	163	0.75
19	Flammagenitus	160 3	25.0	120K	120K	95	0.003	200	1.83

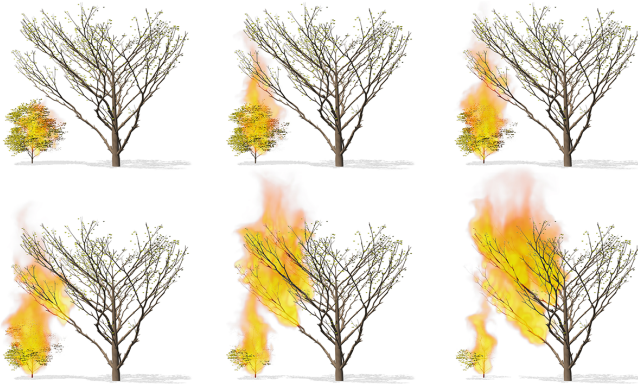


Fig. 6. Vertical fire spread onto a big tree from a small tree underneath.

The procedure summarized in Algorithm 1 starts with an iteration over all modules $\mathcal{M} \in \mathcal{F}$ updating their masses M according to Eq. (1). For this, reaction rates k have to be computed according to Eq. (2) for a given module temperature using a threshold velocity of $u_{\text{ref}} = 15$ m/s corresponding to strong wind boosting the reaction rate by a factor of η_{max} . The local velocity of $u = \|u\|$ is computed as the average absolute wind speed within the cells which contain (a part of) the corresponding module. The mass updates are followed by radii updates according to Eq. (11–12). Then, module temperatures $T_{\mathcal{M}}$ are computed according to Eq. (30), and released water contents W are estimated according to Eq. (17).

Please note that mass $M = M(\mathcal{M})$ and water content $W = W(\mathcal{M})$ are computed per module and have to be transferred to the grid in order to update temperature (Eq. (21)) and water vapor density (Eq. (18)), and to simulate smoke (Eq. (16)). Consequently, we introduce time-dependent scalar fields

$$M : (\mathbf{x}, t) \mapsto M(\mathbf{x}, t) \quad \text{and} \quad W : (\mathbf{x}, t) \mapsto W(\mathbf{x}, t),$$

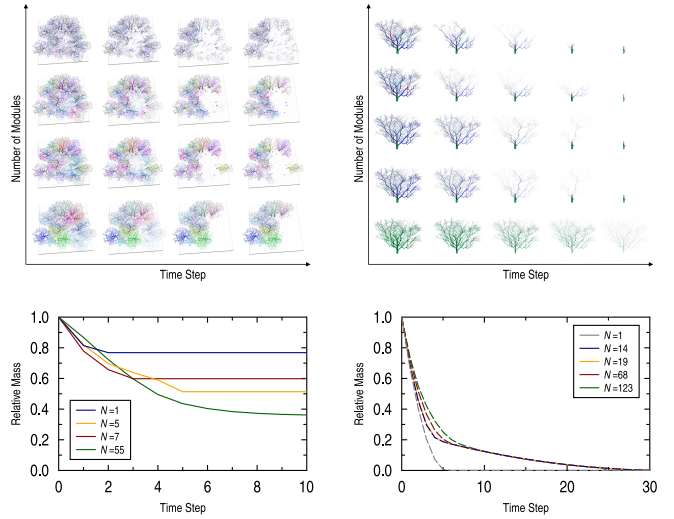


Fig. 7. Analysis of the effect of different numbers of modules by studying a small ecosystem (left) and an individual tree (right). The average number of modules per tree within the ecosystem (left) is given by 55, 7, 5, and 1 (from top to bottom). The single tree (right) is decomposed into 123, 68, 19, 14 modules, and into a single module (from top to bottom). The corresponding mass loss is shown in the diagrams below.

which return mass $M(\mathbf{x}) := M(\mathbf{x}, t)$ and water content $W(\mathbf{x}) := W(\mathbf{x}, t)$ relative to a grid point (i.e. cell center point) $\mathbf{x} \in \mathcal{D}(\Omega)$ for a fixed point in time $t \in \mathbb{R}^{\geq 0}$. These quantities are computed by taking into account all modules \mathcal{M} which (partially) overlap with the cell around \mathbf{x} , summing up their masses M or water content W weighted by a factor of $(1 - \|\mathbf{x} - \text{center}(\mathcal{M})\|/\Delta x)$ in which center(\mathcal{M}) denotes the center of mass of \mathcal{M} . Based on this, temperatures T can be updated according to Eq. (21) as well as q_v , q_c , q_r , q_s , and \mathbf{u} according to the Eulerian fluid solver of Hädrich et al. [2020]. Here, we include drag forces \mathbf{f}_d according to Eq. (15) into the external forces \mathbf{f} in Eq. (13). We apply no-slip boundary conditions at the ground, free-slip conditions at the top, and mixed boundary conditions at the sides [Hädrich et al. 2020]. Moreover, vorticity confinement is included as nonphysical damping caused by numerical dissipation removes interesting turbulent flow features. To avoid this, a vorticity confinement force as introduced by Steinhoff and Underhill [1994] is applied, which injects the dissipated energy back into the system. The strength of the vorticity confinement can be adjusted using a parameter ϵ according to Fedkiw et al. [2001].

Finally, we check which modules \mathcal{M} completely burned down (i.e. $M(\mathcal{M}) = 0$) and remove them from the module space \mathcal{F} as well as their children within the arborescence.

5.2 Implementation

We implemented Algorithm 1 within a C++/CUDA framework. When updating mass, water content, and temperatures, we employ regular forward finite differences. Moreover, Hädrich et al. [2020] kindly provided the source code of their cloud simulator, which we extended by including an additional term proportional to the change of water content W to Eq. (18) as well as our novel formulations for E_r given by Eq. (22) and V_r given by Eq. (31). Please note that their implementation utilises the concept of potential temperature, hence

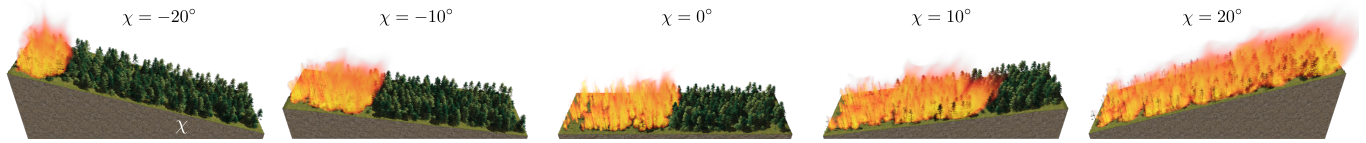


Fig. 8. Snapshots showing simulations of fire spread on inclined planes with different slope angles $\chi \in \{-20^\circ, -10^\circ, 0^\circ, 10^\circ, 20^\circ\}$ (from left to right).

we have to convert our absolute temperature values to potential temperature and vice versa when using their solver.

For rendering purposes, volume ray casting [Pharr et al. 2016] is employed using OpenGL/GLSL evaluating rays of light as they pass through the volume. Tree geometries and leaves are handled dynamically in the geometry shader. For each pixel intersecting the volume, opacity and color are returned and visualized on the screen in real-time, which allows for the interactive exploration of our simulations.

6 RESULTS

We present an overview of our results simulated according to Algorithm 1 implemented as described in the previous section. Initial tree geometries were generated using the ecosystem simulator developed by Makowski et al. [2019], who kindly provided their software. An overview of the different scenes presented throughout this section is provided in Table 1 including relevant parameters. The scenes can be simulated interactively or even in real-time. The computation times listed in Table 1 are measured on an up-to-date desktop computer running our simulation framework on an NVIDIA® GeForce® GTX 1080.

6.1 Fire Spread

The use of detailed branch geometry allows us to realistically capture the three-dimensional fire spread, which cannot be easily covered with other representations or statistical models. For example, when a tree underneath another tree is burning, fire can spread vertically as illustrated in Figure 6. This would not occur in simplified spatial representations such as statistical models. We further demonstrate the effect of fire spread in a forest using a constant horizontal wind field. We vary the slope of the ground terrain. As expected, for a positive slope the fire spreads faster compared to a flat terrain or a negative slope. This is qualitatively illustrated in Figure 8 and quantitatively evaluated in Figure 11 (left), indicating an exponential decrease of the propagation time with increasing slope.

6.2 Mesoscale Representation

The mesoscale representation for trees enables us to model collections of trees with detailed branch geometry for individual modules. Specifically, modeling trees with modules enables controlling the level of detail. Neither coarse-scale nor fine-scale representations offer these two benefits at once. Furthermore, trees dynamically adapt to their individual environment, which cannot simply be represented by fixed proxy shapes such as cones. Modules capture both the self-organization as well as the recursive attribute of tree architecture, thus leading to a more realistic geometric representation.

We evaluate our module-based representation by comparing simulation results using different numbers of modules. As shown in Figure 7 (top), we provide a qualitative comparison by simulating an individual tree as well as an ecosystem. A quantitative comparison is given in Figure 7 (bottom) monitoring the mass loss over time for different numbers of modules.

6.3 Coarse-scale Comparison

To showcase the impact of taking detailed branch geometry into account, we simulate fire spread within an ecosystem composed of several conifers arranged on a flat terrain. As shown in Figure 9, the fire spread is simulated with our mesoscale approach as well as by using a simplified coarse-scale representation which mimics certain properties of the ecosystem on a higher level of abstraction. Each tree is represented as a single cylindrically shaped trunk whose mass corresponds to the mass of the whole tree in the mesoscale representation. In particular, the wood density is adopted, the trunk’s height corresponds to the height of the tree in the mesoscale representation, and the trunk’s diameter is adjusted in a way that the trunk’s mass equals the sum of the tree’s module masses. While using the mesoscale approach, some time is required until a single tree potentially burns completely, the coarse-scale representation results in a direct inflammation of the whole trunk which immediate starts to burn brightly as detailed branch geometry is not considered. As we mimic the original tree widths, the fire propagates remarkably fast through the ecosystem resulting in an unrealistic wildfire scenario. This is quantitatively evaluated in Figure 11 (right) which shows the relative change of mass within the whole ecosystem. It can be observed that the coarse-scale representation results in a fast combustion of a major part of the ecosystem while the mesoscale approach leads to a distinctively different result.

6.4 Ecosystem Properties

Our framework allows for studying the influence of different ecosystem properties. First, we investigate the impact of different tree types as illustrated in Figure 10. In particular, we simulate the temporal evolution of a wildfire within an ecosystem which contains conifers and deciduous trees. We observe that the deciduous trees are more fire resistant compared to the conifers, which is partially caused by their higher density. This results in a characteristic change of the forest pattern caused by the wildfire, changing the ratio of conifers and deciduous trees within the ecosystem.

Moreover, we study the impact of the forest cover (i.e. the relative amount of land covered by trees) on the wildfire. As reported in the literature [Abades et al. 2014], we observe that below a specific forest coverage, fire spread is inhibited as there is not sufficient biomass to

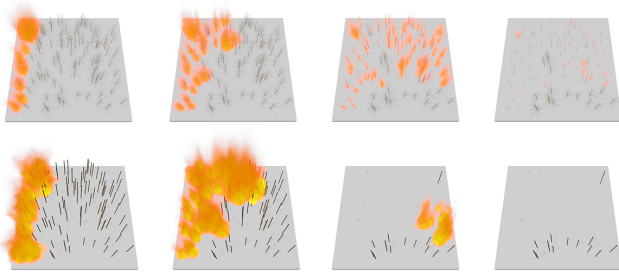


Fig. 9. Temporal evolution (from left to right) of a wildfire burning down conifers simulated with our detailed mesoscale approach (top) and by using a simplified coarse-scale model (bottom). The mesoscale representation contains on average 85 modules per tree. As shown in the left images, the complete left tree front is initially set on fire.

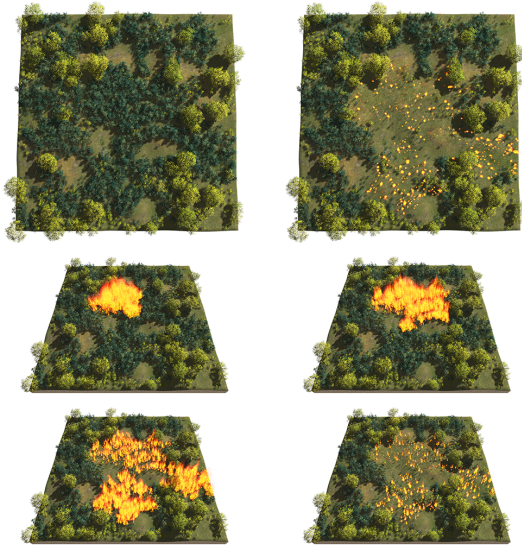


Fig. 10. Temporal evolution (middle and bottom row) of a wildfire burning down conifers and deciduous trees. The fact that the deciduous trees are more fire resistant compared to the conifers results in a different forest pattern (top right) compared to the state before the wildfire (top left).

propagate the fire. In contrast, once a specific threshold is reached, the fire spreads intensively. The graph in Figure 12 (left) shows the relative number of trees with burned modules for different forest cover. The corresponding ecosystems are shown in Figure 14.

6.5 Wildfire Management

Modern wildfire management comprises fire prevention and response as well as recovery work. To evaluate the human impact on wildfires, our model allows us to interactively extinguish fire by manually distributing fire retardant as well as by setting up fire barriers. This is qualitatively illustrated in Figure 15. Moreover, we quantitatively evaluate the impact of fire barriers using different barrier widths as shown in Figure 12 (right). Once a specific threshold width of about 7.9 m in our example is reached, the fire is not able to jump over the barrier. Please note that the discrete jump

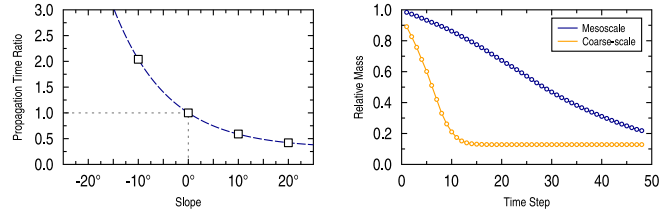


Fig. 11. Left: Dependence of the speed of the fire spread on different slope values indicating an exponential trend. For an angle of -20° the fire extinguished itself, not being able to propagate over the whole negatively sloped plane. Right: Temporal evolution of the mass (relative to the initial mass) measured from the simulation using our detailed mesoscale approach (blue) and by using a simplified coarse-scale model (orange).

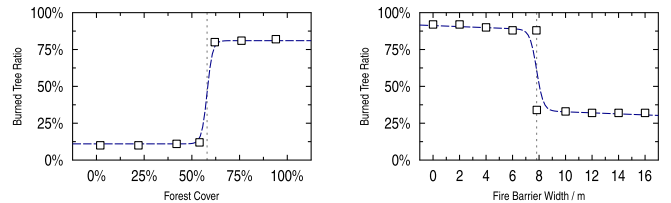


Fig. 12. Left: Dependence of the relative number of trees with burned modules on the forest coverage. Once a threshold of about 58% of forest cover is reached, the fire spreads over the whole ecosystem. Right: Cohort of trees are cut resulting in empty fire barrier zones of different widths. The number of trees with burned modules depending on the width is shown here.

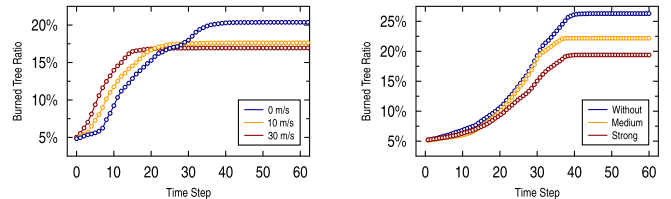


Fig. 13. Left: Temporal evolution of the relative number of burned trees for different scenarios without wind (blue), with moderate wind (orange), and with strong wind (red). Right: Equivalent results predicted by a percolation model expressing the effects of wind on fire spread for the same vegetation scene. Both models predict a decrease in burned tree ratios with increasing wind speed. The handling of the percolation model is described by Abades et al. [2014] who kindly provided their source code.

in Figure 12 (right) is expected as flying sparks are ignored in our model.

6.6 Atmospheric Conditions

We study the impact of different atmospheric conditions by varying the speed of a constant horizontal wind field as shown in Figure 17. This is quantitatively evaluated in Figure 13 showing the relative number of burning trees (including already burned trees) over time. As expected, the fire spreads faster with increasing wind speed, which is caused by the propagation of flames due to wind as well as the fact that fire increases its temperature and accelerates the combustion process if wind is blowing oxygen into the fire (see Eq. (3)). Consequently, the relative number of burned trees increases faster in the beginning of the wildfire when wind is stronger. However, as

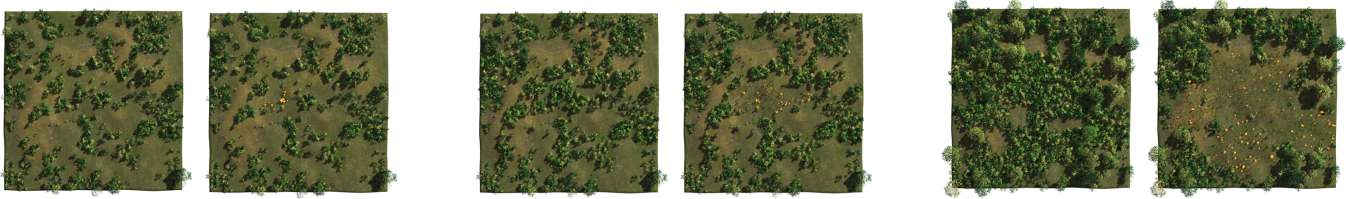


Fig. 14. Illustration of the fire spread in different ecosystems varying with respect to their initial forest cover of 25% (left), 45% (middle), and 90% (right).



Fig. 15. Top: Illustration of the user experience of our interactive wildfire simulator. Simulation and rendering are running in real-time so that the user can conveniently explore the scene. After inspecting the wildfire, fire retardant is interactively distributed partially extinguishing the fire. The application of fire retardant is implemented by setting the module temperature to the environmental temperature at selected trees. Bottom: The temporal evolution of a scenario is shown in which the user sets up an empty fire barrier zone by cutting trees. As the width of the barrier zone is not sufficient to protect the forest, the fire spreads over the zone further increasing forest damage.

it can be observed from Figure 13 as well, wind also blows the fire in a dominant direction, potentially hindering it to spread isotropically, resulting in less forest damage in the long-term.

6.7 National Park Wildfire

We simulate a complex wildfire originating from a randomly positioned fire source which could be caused by a lightning strike. The fire spreads within the valley around Half Dome in California’s Yosemite National Park as shown in Figure 1. This scene contains about 68K trees composed of about 75K individual modules. The simulation runs at interactive rates.

6.8 Flammagenitus Clouds



Fig. 16. Photography of a flammagenitus.

Flammagenitus clouds are dense grayish to brown cumuliform clouds which potentially emerge from wildfires or volcanic eruptions. While it may seem counterintuitive that water vapor condenses to form clouds in the vicinity of a hot flame, Eq. (17) reveals that burning wood releases large amounts of water that supersaturate the air above a fire. The effect of flammagenitus clouds on wildfires is two-fold and dependent on the size and intensity of the fire. Flammagenitus clouds can condense, resulting in rainfall contributing to potentially extinguishing the fire. On the other hand, huge wildfires can produce growing flammagenitus clouds (cumulonimbus flammagenitus) which can trigger

thunderstorms from which lightning can become an additional fire source [Dowdy et al. 2017]. Figure 18 shows the simulation of a wildfire scenario from which a flammagenitus cloud emerges whose rainfall finally extinguishes the fire. The scene is rendered from a cross-sectional perspective highlighting water vapor and condensed cloud water showcasing the formation of the flammagenitus cloud. A more complex scene comprising a huge flammagenitus cloud over a wildfire is shown in Figure 19. This scene comprises 120K individual trees grown on a mountainous terrain. It runs at interactive rates. As a comparison, a photograph of a real flammagenitus cloud is shown in Figure 16. While dark smoke is present at lower altitudes, the condensation of water at higher altitudes results in a white cumuliform cloud. This can be observed in our simulation as well as in the photography.

7 DISCUSSION

An important result of analytical studies on forest fires (e.g. using percolation models) is that the geometric distribution of vegetation in space is a major determinant of fire spread. Consequently, the predictive power of wildfire simulators depends on detailed tree form representations. Existing theoretical methods study wildfires by employing coarse tree representations, such as 2D grids, voxels or cones [Mendoza et al. 2019]. In contrast, our method proposes a significantly more detailed representation of tree form based on branch modules. Our comparison to a coarse-scale model (Section 6.3) showcases the importance of detailed tree geometry.

Our simulation results indicate that wildfires are characterized by a tipping-point phenomenon. Specifically, we report that with increasing forest cover the burned tree ratio increases following a hill-shaped relation (Figure 12, left). Such a non-linear relation is

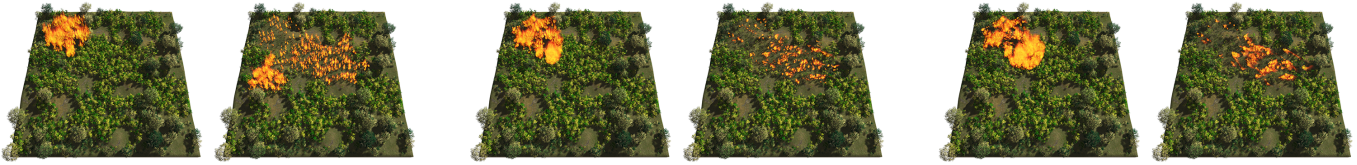


Fig. 17. Temporal evolution of wildfires without wind (left), with moderate horizontal wind (middle, 10 m/s), and with strong horizontal wind (right, 30 m/s).

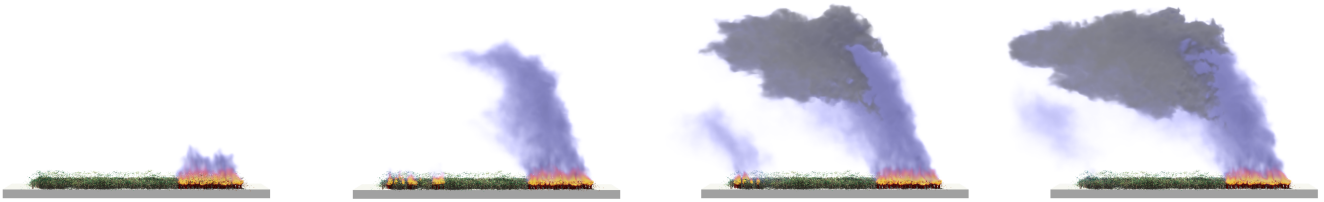


Fig. 18. Simulation of a wildfire (left) forming a flammagenitus cloud (second from the left) which produces rainfall (second from the right) partially extinguishing the fire (right). The condensed (cloud) water is visualized in gray while water vapor is shown in blue.

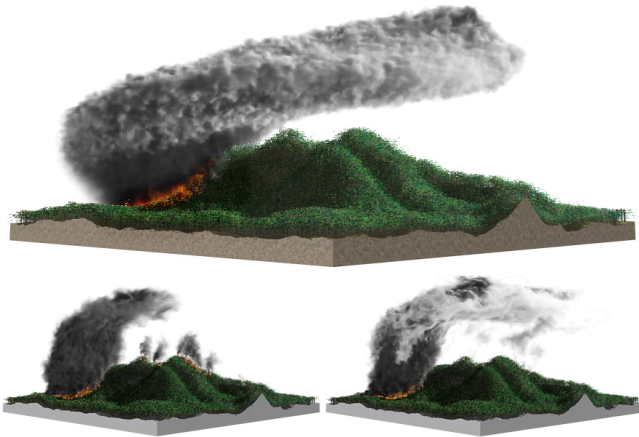


Fig. 19. Simulation of a complex wildfire in a mountainous region from which a typical flammagenitus cloud emerges (top). Dark smoke is present at lower altitudes while the condensation of water at higher altitudes results in a white cumulus cloud. A flammagenitus cloud produces rainfall extinguishing a separately emerging fire source (bottom, left to right).

well documented by simulations with percolation models [Abades et al. 2014]. Furthermore, let us compare our simulation results of evaluating the impact of wind speed on forest fires (Figure 13, left) with equivalent results of such a percolation model that expresses the effects of uniform wind (Figure 13, right). For this, we calculated the vegetation input (a 2D grid) for the percolation model based on the forest scene used in our simulation results. Qualitatively, both models exhibit a similar relation of wind speed on burned tree ratio. Specifically, burned tree ratios decrease with increasing wind for our method as well as for the percolation model.

Our model describes terrain-wind and tree-wind interaction by employing a state-of-the-art fluid solver. Due to boundary effects of the fluid domain this can readily result in turbulent air flows. However, capturing turbulence caused by tree-wind interaction can be practically impeded as this is not captured adequately if the

modules are smaller than the grid resolution. We leave validating turbulent air flow for future work.

8 LIMITATIONS AND FUTURE WORK

We close with suggestions for multiple avenues of future work. A common phenomenon associated with tree combustion is fire spread on the ground facilitated by grass, branch litter, and undergrowth vegetation as modeled by Kapp et al. [2020]. An individual-based representation for these elements would significantly increase the simulation time of our method, which is why we ignored it in the present work. Another major mediator of fire spread of wildfires are sparks flying through the air. The processes underlying the formation of sparks happen at spatial scales much smaller than the processes considered in our model. However, an accurate averaged representation of flying sparks would improve the realism of fire spread in our approach. Another path would be to include the effect of tree resin as this highly flammable substance is an ideal natural fuel for starting fires and even forms explosive mixtures with air. On another trajectory, the integration of lightning effects into the model would be attractive to capture their impact on wildfires as, e.g., flammagenitus clouds can trigger a thunderstorm potentially initiating another fire. This forms a positive feedback loop which would benefit from computational analysis.

Another important aspect for future work is including realistic tree dynamics. To simulate this plausibly, we would have to take the influence of wind on the trees into account. This would allow to capture branch modules swaying in the wind. One option could be to utilize the concept of Cosserat rods [Deul et al. 2018; Michels et al. 2015] which has already been applied to simulate tree dynamics [Pirk et al. 2017] on a branch level and modify it in order to directly act on the level of modules.

9 CONCLUSION

We have introduced a novel method for modeling and simulating wildfires. Our method employs branch modules as an efficient geometric representation for tree models that provides us with complex

and realistic branching structures for individual trees and consequently entire forests. We use this representation to define a novel combustion model for plants that allows us to capture various effects to realistically simulate the pyrolysis of wood, including char insulation, mass loss, and heat transfer. We have shown that our method can be used to realistically capture the fire spread on terrain with varying vegetation occupancy. Furthermore, by using a physically plausible model for fluid dynamics of the atmosphere we can capture the emergent fire spread over terrain topography. By coupling combustion, fire and atmosphere models, we have been able to capture the formation of flammagenitus clouds.

In summary, our method enables the exploration of complex wildfire scenarios. We have been able to simulate scenes with more than 100K individual trees represented by complex and detailed geometry at interactive rates. Our approach is scalable to even larger forests when compromising interactivity. This would be justifiable as we see the main contribution of our work in improving physical and biological realism of wildfires. To perform high-fidelity simulations of recent real wildfires, our model has to be extended by including fire spread on the ground facilitated by grass, branch litter, and undergrowth vegetation, taking the role of leaves into account, and including sparks flying through the air. Continuing research in this direction is an exciting avenue for applying our methodology.

ACKNOWLEDGMENTS

This work was supported and funded by KAUST through the baseline funding of the Computational Sciences Group within the Visual Computing Center. The authors are thankful to Miłosz Makowski and Weronika Skowrońska who generated the initial ecosystems. The insightful discussions with Jorge Alejandro Amador Herrera and Franziska Lissel as well as the reviewers' valuable comments that improved the manuscript are gratefully acknowledged.

REFERENCES

- S. R. Abades, A. Gaxiola, and P. A. Marquet. 2014. Fire, percolation thresholds and the savanna forest transition: a neutral model approach. *Journal of Ecology* 102, 6 (2014), 1386–1393.
- C. Anand, B. Shotorban, S. Mahalingam, S. McAllister, and D. Weise. 2017. Physics-Based Modeling of Live Wildland Fuel Ignition Experiments in the FIST Apparatus. *Combustion Science and Technology* 189 (2017).
- M. Aono and T.L. Kunii. 1984. Botanical Tree Image Generation. *IEEE Comput. Graph. Appl.* 4(5) (1984), 10–34.
- O. Argudo, C. Andújar, A. Chica, E. Guérin, J. Digne, A. Peytavie, and E. Galin. 2017. Coherent multi-layer landscape synthesis. *The Visual Computer* 33, 6 (2017), 1005–1015.
- O. Argudo, A. Chica, and C. Andujar. 2016. Single-picture Reconstruction and Rendering of Trees for Plausible Vegetation Synthesis. *Comput. Graph.* 57, C (2016), 55–67.
- O. Argudo, E. Galin, A. Peytavie, A. Paris, and E. Guérin. 2020. Simulation, Modeling and Authoring of Glaciers. *ACM Transactions on Graphics (SIGGRAPH Asia 2020)* 39, 6 (2020).
- S. Behrendt, C. Colditz, O. Franzke, J. Kopf, and O. Deussen. 2005. Realistic real-time rendering of landscapes using billboard clouds. *Computer Graphics Forum* 24, 3 (2005), 507–516.
- B. Beneš, N. Andryscio, and O. Stava. 2009. Interactive Modeling of Virtual Ecosystems. In *Proceedings of the Fifth Eurographics Conference on Natural Phenomena (NPH'09)*. Eurographics Association, Goslar, DEU, 9–16.
- J. Bloomenthal. 1985. Modeling the Mighty Maple. *SIGGRAPH Comput. Graph.* 19, 3 (July 1985), 305–311.
- C. F. Bohren and D. B. Thorud. 1973. Two theoretical models of radiation heat transfer between forest trees and snowpacks. *Agric. For. Meteorol.* 11 (1973), 3–16.
- R. Bridson. 2008. *Fluid Simulation for Computer Graphics*. A K Peters, CRC Press.
- E. Bruneton and F. Neyret. 2012. Real-time Realistic Rendering and Lighting of Forests. *Comput. Graph. Forum* 31, 2pt1 (2012), 373–382.
- V. P. Carey. 1992. *Liquid–Vapor Phase-Change Phenomena*. Taylor & Francis.
- X. Chen, B. Neubert, Y.-Q. Xu, O. Deussen, and S. B. Kang. 2008. Sketch-Based Tree Modeling Using Markov Random Field. *ACM Trans. Graph.* 27, 5, Article 109 (Dec. 2008), 9 pages.
- N. P. Cheney, J. S. Gould, and W. R. Catchpole. 1993. The Influence of Fuel, Weather and Fire Shape Variables on Fire-Spread in Grasslands. *International Journal of Wildland Fire* 3, 1 (1993), 31–44.
- N. Chiba, K. Muraoka, H. Takahashi, and M. Miura. 1994. Two-dimensional visual simulation of flames, smoke and the spread of fire. *JVCA* 5, 1 (1994), 37–53.
- B. V. Chileen, K. K. McLauchlan, P. E. Higuera, M. Parish, and B. N. Shuman. 2020. Vegetation response to wildfire and climate forcing in a Rocky Mountain lodgepole pine forest over the past 2500 years. *The Holocene* 30, 11 (2020), 1493–1503.
- J. Coen. 2005. Simulation of the Big Elk Fire using coupled atmosphere–fire modeling. *International Journal of Wildland Fire* 14 (2005), 49–59.
- R. L. Cook, J. Halstead, M. Planck, and D. Ryu. 2007. Stochastic Simplification of Aggregate Detail. *ACM Trans. Graph.* 26, 3 (July 2007), 79.
- G. Cordonnier, P. Ecomier, E. Galin, J. Gain, B. Benes, and M.-P. Cani. 2018. Interactive Generation of Time-evolving, Snow-Covered Landscapes with Avalanches. *CGF* 37, 2 (2018), 497–509.
- G. Cordonnier, E. Galin, J. Gain, B. Benes, E. Guérin, A. Peytavie, and M.-P. Cani. 2017. Authoring Landscapes by Combining Ecosystem and Terrain Erosion Simulation. *ACM Trans. Graph.* 36, 4, Article 134 (2017), 12 pages.
- W. A. Côté. 1968. *Chemical Composition of Wood*. Springer Berlin Heidelberg, Berlin, Heidelberg, 55–78.
- P. de Reffye, C. Edelin, J. Françon, M. Jaeger, and C. Puech. 1988. Plant Models Faithful to Botanical Structure and Development. *SIGGRAPH Comput. Graph.* 22, 4 (June 1988), 151–158.
- C. Deul, T. Kugelstadt, M. Weiler, and J. Bender. 2018. Direct Position-Based Solver for Stiff Rods. *Computer Graphics Forum* 37, 6 (2018), 313–324.
- O. Deussen, C. Colditz, M. Stamminger, and G. Drettakis. 2002. Interactive Visualization of Complex Plant Ecosystems. *VIS '02* (2002), 219–226.
- O. Deussen, P. Hanrahan, B. Lintermann, R. Mëch, M. Pharr, and Przemyslaw Prusinkiewicz. 1998. Realistic Modeling and Rendering of Plant Ecosystems. *ACM Trans. Graph.* (1998), 275–286.
- A. J. Dowdy, M. D. Fromm, and N. McCarthy. 2017. Pyrocumulonimbus lightning and fire ignition on Black Saturday in southeast Australia. *Journal of Geophysical Research (Atmospheres)* 122, 14 (July 2017), 7342–7354.
- J.-L. Dupuy and M. Larini. 2000. Fire spread through a porous forest fuel bed: a radiative and convective model including fire-induced flow effects. *International Journal of Wildland Fire* 9, 3 (2000), 155–172.
- D. Ebert, F. Musgrave, D. Peachey, K. Perlin, and S. Worley. 2002. *Texturing and Modeling: A Procedural Approach* (3rd ed.). Morgan Kaufmann Publishers Inc.
- L. Hernández Encinas, S. Hoya White, A. Martín del Rey, and G. Rodríguez Sánchez. 2007. Modelling forest fire spread using hexagonal cellular automata. *Appl. Math. Model.* 31, 6 (2007), 1213–1227.
- R. Fedkiw, J. Stam, and H. W. Jensen. 2001. Visual Simulation of Smoke. *Proc. of ACM SIGGRAPH* (2001), 15–22.
- M. Finney, J. Cohen, J. Forthofer, S. McAllister, M. Gollner, D. Gorham, K. Saito, N. Akafuah, B. Adam, and J. English. 2015. Role of buoyant flame dynamics in wildfire spread. *Proceedings of the National Academy of Sciences* 112, 32 (2015), 9833–9838.
- A. Fournier, D. Fussell, and L. Carpenter. 1982. Computer Rendering of Stochastic Models. *Commun. ACM* 25, 6 (1982), 371–384.
- E. Galin, E. Guérin, A. Peytavie, G. Cordonnier, M.-P. Cani, B. Benes, and J. Gain. 2019. A Review of Digital Terrain Modeling. *Computer Graphics Forum* 38, 2 (2019), 553–577.
- É. Guérin, J. Digne, É. Galin, A. Peytavie, C. Wolf, B. Benes, and B. Martinez. 2017. Interactive Example-Based Terrain Authoring with Conditional Generative Adversarial Networks. *ACM Trans. Graph.* 36, 6, Article 228 (Nov. 2017), 13 pages.
- N. Gustenyov, N. K Akafuah, A. Salaimeh, M. Finney, S. McAllister, and K. Saito. 2018. Scaling nonreactive cross flow over a heated plate to simulate forest fires. *Combustion and Flame* 197 (2018), 340–354.
- R. Habel, A. Kusternig, and M. Wimmer. 2009. Physically Guided Animation of Trees. *Comp. Graph. Forum* 28, 2 (2009), 523–532.
- T. Hädrich, B. Benes, O. Deussen, and S. Pirk. 2017. Interactive Modeling and Authoring of Climbing Plants. *CGF* 36, 2 (2017), 49–61.
- T. Hädrich, M. Makowski, W. Pałubicki, D. T. Banuti, S. Pirk, and D. L. Michels. 2020. Stormscapes: Simulating Cloud Dynamics in the Now. *ACM Transaction on Graphics* 39, 6, Article 175 (12 2020).
- M. J. Harris, W. V. Baxter, T. Scheuermann, and A. Lastra. 2003. Simulation of Cloud Dynamics on Graphics Hardware. In *ACM SIGGRAPH/EUROGRAPHICS Conference on Graphics Hardware (HWWS '03)*. Eurographics Association, 92–101.
- Y. Hong, D. Zhu, X. Qiu, and Z. Wang. 2010. Geometry-based Control of Fire Simulation. *Vis. Comput.* 26, 9 (2010), 1217–1228.
- C. Horvath and W. Geiger. 2009. Directable, High-Resolution Simulation of Fire on the GPU. *ACM Trans. Graph.* 28, 3, Article 41 (July 2009), 8 pages.
- ISO. 1975. *Standard Atmosphere*. Technical Report ISO 2533:1975. International Organization for Standardization.

- M. Jaeger and J. Teng. 2003. Tree and plant volume imaging - An introductory study towards voxelized functional landscapes. *PMA* (2003).
- K. Kapp, J. Gain, E. Guérin, E. Galin, and A. Peytavie. 2020. Data-driven Authoring of Large-scale Ecosystems. *ACM Trans. Graph.* (2020).
- Y. Kawaguchi. 1982. A Morphological Study of the Form of Nature. *SIGGRAPH Comput. Graph.* 16, 3 (July 1982), 223–232.
- A. D. Kelley, M. C. Malin, and G. M. Nielson. 1988. Terrain Simulation Using a Model of Stream Erosion. *SIGGRAPH Comput. Graph.* 22, 4 (1988), 263–268.
- E. Kessler. 1969. *On the Distribution and Continuity of Water Substance in Atmospheric Circulations*. American Meteorological Society, Boston, MA, 1–84.
- A. Lamorlette and N. Foster. 2002. Structural Modeling of Flames for a Production Environment. In *Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '02)*. Association for Computing Machinery, New York, NY, USA, 729–735.
- B. Lane and P. Prusinkiewicz. 2002. Generating Spatial Distributions for Multilevel Models of Plant Communities. *Graphics Interface* (2002), 69–80.
- M. J. Lawes, A. Richards, J. Dathe, and J. J. Midgley. 2011. Bark thickness determines fire resistance of selected tree species from fire-prone tropical savanna in north Australia. *Plant Ecol.* 212, 12 (2011), 2057–2069.
- B. Lintermann and O. Deussen. 1999. Interactive Modeling of Plants. *IEEE Comput. Graph. Appl.* 19, 1 (Jan. 1999), 56–65.
- S. Liu, T. An, Z. Gong, and I. Hagiwara. 2012. *Physically Based Simulation of Solid Objects Burning*. Springer Berlin Heidelberg, Berlin, Heidelberg, 110–120.
- Y. Livny, S. Pirk, Z. Cheng, F. Yan, O. Deussen, D. Cohen-Or, and B. Chen. 2011. Texture-lobes for Tree Modelling. *ACM Trans. Graph.* 30, 4, Article 53 (2011), 10 pages.
- Y. Lizhong, C. Xiaojun, Z. Xiaodong, and F. Weicheng. 2002. A modified model of pyrolysis for charring materials in fire. *Int. J. Eng. Sci.* 40, 9 (2002), 1011–1021.
- S. Longay, A. Runions, F. Boudon, and P. Prusinkiewicz. 2012. TreeSketch: interactive procedural modeling of trees on a tablet. In *Proc. of the Intl. Symp. on SBIM*. 107–120.
- M. Makowski, T. Hädrich, J. Scheffczyk, D. L. Michels, S. Pirk, and W. Palubicki. 2019. Synthetic Silviculture: Multi-Scale Modeling of Plant Ecosystems. *ACM Trans. Graph.* 38, 4, Article 131 (2019), 14 pages.
- M. M. Masinda, L. Sun, G. Wang, and T. Hu. 2020. Moisture content thresholds for ignition and rate of fire spread for various dead fuels in northeast forest ecosystems of China. *Journal of Forestry Research* (05 Jun 2020).
- Z. Melek and J. Keyser. 2002. Interactive simulation of fire. *Pacific Graphics* (2002), 431–432.
- H. Mendoza, A. Brown, and A. Ricks. 2019. Modeling High Heat Flux Combustion of Coniferous Trees Using Chemically Reacting Lagrangian Particles (WSSCI Fall Technical Meeting of the Western States Section of the Combustion Institute).
- D. L. Michels, J. P. T. Mueller, and G. A. Sobottka. 2015. A physically based approach to the accurate simulation of stiff fibers and stiff fiber meshes. *Comput. Graph.* 53 (2015), 136–146.
- R. Minamino and M. Tateno. 2014. Tree branching: Leonardo da Vinci's rule versus biomechanical models. *PLoS one* 9, 4 (2014), e93535.
- S. Monedero, J. Ramirez, D. Molina-Terrén, and A. Cardil. 2017. Simulating wildfires backwards in time from the final fire perimeter in point-functional fire models. *Environmental Modelling & Software* 92 (2017), 163–168.
- R. Mèch and P. Prusinkiewicz. 1996. Visual models of plants interacting with their environment. In *Proc. of SIGGRAPH*. ACM, 397–410.
- B. Neubert, T. Franken, and O. Deussen. 2007. Approximate Image-based Tree-modeling Using Particle Flows. *ACM Trans. Graph.* 26, 3, Article 88 (2007).
- B. Neubert, S. Pirk, O. Deussen, and C. Dachsbacher. 2011. Improved Model- and View-Dependent Pruning of Large Botanical Scenes. *Computer Graphics Forum* 30, 6 (2011), 1708–1718.
- D. Q. Nguyen, R. Fedkiw, and H. W. Jensen. 2002. Physically Based Modeling and Animation of Fire. *ACM Trans. Graph.* 21, 3 (2002), 721–728.
- D. Q. Nguyen, R. P. Fedkiw, and M. Kang. 2001. A Boundary Condition Capturing Method for Incompressible Flame Discontinuities. *J. Comput. Phys.* 172, 1 (2001), 71–98.
- M. Okabe, S. Owada, and T. Igarashi. 2007. Interactive Design of Botanical Trees Using Freehand Sketches and Example-based Editing. In *ACM SIGGRAPH Courses*. ACM, Article 26.
- P. E. Oppenheimer. 1986. Real time design and animation of fractal plants and trees. *Proc. of SIGGRAPH* 20, 4 (1986), 55–64.
- W. Palubicki, K. Horel, S. Longay, A. Runions, B. Lane, R. Mèch, and P. Prusinkiewicz. 2009. Self-organizing Tree Models for Image Synthesis. *ACM Trans. Graph.* 28, 3, Article 58 (2009), 10 pages.
- Z. Pan and D. Manocha. 2017. Efficient Solver for Spacetime Control of Smoke. *ACM Trans. Graph.* 36, 5, Article 162 (July 2017), 13 pages.
- E. Pastor, L. Zárate, E. Planas, and J. Arnaldos. 2003. Mathematical models and calculation systems for the study of wildland fire behaviour. *Progress in Energy and Combustion Science* 29, 2 (2003), 139–153.
- V. Pegoraro and S. G. Parker. 2006. Physically-Based Realistic Fire Rendering. In *Eurographics Workshop on Natural Phenomena*, N. Chiba and E. Galin (Eds.). The Eurographics Association.
- K. Perlin. 1985. An Image Synthesizer. In *Proceedings of the 12th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '85)*. Association for Computing Machinery, 287–296.
- M. Pharr, W. Jakob, and G. Humphreys. 2016. *Physically Based Rendering: From Theory to Implementation* (3rd ed.). Morgan Kaufmann Publishers Inc.
- S. Pirk, M. Jarzabek, T. Hädrich, D. L. Michels, and W. Palubicki. 2017. Interactive Wood Combustion for Botanical Tree Models. *ACM Trans. Graph.* 36, 6, Article 197 (Nov. 2017), 12 pages.
- S. Pirk, T. Niese, O. Deussen, and B. Neubert. 2012a. Capturing and animating the morphogenesis of polygonal tree models. *ACM Trans. Graph.* 31, 6, Article 169 (2012), 10 pages.
- S. Pirk, T. Niese, T. Hädrich, B. Benes, and O. Deussen. 2014. Windy Trees: Computing Stress Response for Developmental Tree Models. *ACM Trans. Graph.* 33, 6, Article 204 (2014), 11 pages.
- S. Pirk, O. Stava, J. Kratt, M. A. M. Said, B. Neubert, R. Mèch, B. Benes, and O. Deussen. 2012b. Plastic trees: interactive self-adapting botanical tree models. *ACM Trans. Graph.* 31, 4, Article 50 (2012), 10 pages.
- P. Prusinkiewicz. 1986. Graphical applications of L-systems. In *Proc. on Graph. Interf.* 247–253.
- L. Quan, P. Tan, G. Zeng, L. Yuan, J. Wang, and S. B. Kang. 2006. Image-based Plant Modeling. *ACM Trans. Graph.* 25, 3 (July 2006), 599–604.
- N. Rasmussen, D. Q. Nguyen, W. Geiger, and R. Fedkiw. 2003. Smoke Simulation for Large Scale Phenomena. *ACM Trans. Graph.* 22, 3 (July 2003), 703–707.
- A. Reche-Martinez, I. Martin, and G. Drettakis. 2004. Volumetric reconstruction and interactive rendering of trees from photographs. 23, 3 (2004), 720–727.
- G. D. Richards. 1990. An elliptical growth model of forest fire fronts and its numerical solution. *Internat. J. Numer. Methods Engrg.* 30, 6 (1990), 1163–1179.
- D. W. Schwilk. 2003. Flammability Is a Niche Construction Trait: Canopy Architecture Affects Fire Intensity. *The American Naturalist* 162, 6 (2003), 725–733.
- R. Seidl, W. Rammer, R. M. Scheller, and T. A. Spies. 2012. An individual-based process model to simulate landscape-scale forest ecosystem dynamics. *Ecological Modelling* 231 (2012), 87–100.
- H. Shao, T. Kugelstadt, T. Hädrich, W. Palubicki, J. Bender, S. Pirk, and D. L. Michels. 2021. Accurately Solving Physical Systems with Graph Learning. arXiv:physics.comp-ph/2006.03897
- A. R. Smith. 1984. Plants, Fractals, and Formal Languages. In *Proceedings of the 11th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '84)*. Association for Computing Machinery, New York, NY, USA, 1–10.
- J. Stam. 1999. Stable Fluids. *Proc. of ACM SIGGRAPH* (1999), 121–128.
- M. Stamminger and G. Drettakis. 2001. Interactive Sampling and Rendering for Complex and Procedural Geometry. In *Proceedings of the 12th Eurographics Conference on Rendering (EGWR'01)*. Eurographics Association, Goslar, DEU, 151–162.
- O. Stava, S. Pirk, J. Kratt, B. Chen, R. Mèch, O. Deussen, and B. Benes. 2014. Inverse Procedural Modelling of Trees. *CGF* 33, 6 (2014), 118–131.
- J. Steinhoff and D. Underhill. 1994. Modification of the Euler equations for “vorticity confinement”: Application to the computation of interacting vortex rings. *Phys. Fluids* 6, 8 (1994), 2738–2744.
- A. Stomakhin, C. Schroeder, C. Jiang, L. Chai, J. Teran, and A. Selle. 2014. Augmented MPM for Phase-change and Varied Materials. *ACM Trans. Graph.* 33, 4, Article 138 (2014), 11 pages.
- R. Sun, S. K. Krueger, M. A. Jenkins, M. A. Zulauf, and J. J. Charney. 2009. The importance of fireatmosphere coupling and boundary-layer turbulence to wildfire spread. *International Journal of Wildland Fire* 18, 1 (2009), 50–60.
- P. Tan, T. Fang, J. Xiao, P. Zhao, and L. Quan. 2008. Single Image Tree Modeling. *ACM Trans. Graph.* 27, 5, Article 108 (2008), 7 pages.
- V. D. Thi, M. Khelifa, M. El Ganaoui, and Y. Rogaume. 2016. Finite element modelling of the pyrolysis of wet wood subjected to fire. *Fire Safety Journal* 81 (2016), 85–96.
- J. van Lawick van Pabst and H. Jense. 1996. Dynamic Terrain Generation Based on Multifractal Techniques. In *High Performance Computing for Computer Graphics and Visualisation*, M. Chen, P. Townsend, and J. A. Vince (Eds.). London, 186–203.
- H. Y. Wang, M. Z. Kang, J. Hua, and X. J. Wang. 2013. Modeling Plant Plasticity from a Biophysical Model: Biomechanics. In *Proceedings of the 12th ACM SIGGRAPH Intl. Conf. on VRCAI*. ACM, 115–122.
- J. Weber and J. Penn. 1995. Creation and Rendering of Realistic Trees. In *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '95)*. Association for Computing Machinery, New York, NY, USA, 119–128.
- J. Wither, F. Boudon, M.-P. Cani, and C. Godin. 2009. Structure from silhouettes: a new paradigm for fast sketch-based design of trees. *CGF* 28, 2 (2009), 541–550.
- H. Xu, N. Gossett, and B. Chen. 2007. Knowledge and heuristic-based modeling of laser-scanned trees. 26, 4 (2007), Article 19, 13 pages.
- M. K. Yau and R. R. Rogers. 1996. *A Short Course in Cloud Physics*. Elsevier Science.
- Y. Zhao and J. Barbić. 2013. Interactive Authoring of Simulation-ready Plants. *ACM Trans. Graph.* 32, 4, Article 84 (2013), 12 pages.
- Y. Zhao, X. Wei, Z. Fan, A. Kaufman, and H. Qin. 2003. Voxels on fire [computer animation]. In *IEEE Visualization, 2003. VIS 2003*. 271–278.