

Modeling Dense Inflorescences

Andrew Owens¹ Mikolaj Cieslak¹ Jeremy Hart¹ Regine Classen-Bockhoff² Przemyslaw Prusinkiewicz¹
¹University of Calgary ²Johannes Gutenberg-Universität Mainz



Figure 1: A photograph and a model of an *Orlaya grandiflora* inflorescence. These images illustrate some of the key elements of our paper: the organization of florets into a planar canopy, hierarchical phyllotaxis, the dependency of floret type and petal size on their position in the inflorescence, and the deformation of some petals due to collisions. Photograph by Holger Casselmann licensed under CC BY-SA 3.0.

Abstract

Showy inflorescences – clusters of flowers – are a common feature of many plants, greatly contributing to their beauty. The large numbers of individual flowers (florets), arranged in space in a systematic manner, make inflorescences a natural target for procedural modeling. We present a suite of biologically motivated algorithms for modeling and animating the development of inflorescences with closely packed florets. These inflorescences share the following characteristics: (i) in their ensemble, the florets form a relatively smooth, often approximately planar surface; (ii) there are numerous collisions between petals of the same or adjacent florets; and (iii) the developmental stage and type of a floret may depend on its position within the inflorescence, with drastic or gradual differences. To model flat-topped branched inflorescences (corymbs and umbels), we propose a florets-first algorithm, in which the branching structure self-organizes to support florets in predetermined positions. This is an alternative to previous branching-first models, in which floret positions were determined by branch arrangement. To obtain realistic visualizations, we complement the algorithms that generate the inflorescence structure with an interactive method for modeling floret corollas (petal sets). The method supports corollas with both separate and fused petals. We illustrate our techniques with models from several plant families.

Keywords: procedural modeling, inflorescence, phyllotaxis, head, umbel, corymb, floret, floral dimorphism, branching architecture, intrinsic interpolation, collision detection and response, position-based dynamics, Ridley’s model, Rodkaew’s algorithm.

Concepts: •Computing methodologies → Procedural animation; Shape modeling; Collision detection; •Applied computing → Life and medical sciences;

1 Introduction

In many plant species, flowers are grouped into multi-flower assemblies called inflorescences. Such floral arrangements have selective value: by being more visible, inflorescences can attract pollinators from a larger distance than individual flowers; furthermore, by supporting walking between adjacent florets, inflorescences may facilitate their pollination by insects. From an aesthetic perspective, showy inflorescences are a visually appealing attribute of many plants occurring in natural and artificial settings. The common occurrence and beauty of inflorescences has made them an attractive modeling subject in computer graphics [Prusinkiewicz et al. 1988; Prusinkiewicz and Lindenmayer 1990; Fowler et al. 1992; Prusinkiewicz et al. 1993; Lintermann and Deussen 1999; Prusinkiewicz et al. 2001; Ijiri et al. 2005; Ijiri et al. 2006]. In this paper we extend the class of inflorescences that can be modeled and animated for image synthesis purposes.

1.1 Biological context

An insight into the form of plants is offered by the first available space theory of plant organ initiation, formulated in the XIX century by Wilhelm Hofmeister (see [Kirchoff 2003] for a recent description). According to this theory, incipient organs, such as leaf or floret primordia, are positioned on the growing surface of the (shoot

Permission to make digital or hard copies of all or part 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 components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org. © 2016 Copyright held by the owner/author(s). Publication rights licensed to ACM.

SIGGRAPH ’16 Technical Paper, July 24 - 28, 2016, Anaheim, CA, ISBN: 978-1-4503-4279-7/16/07

DOI: <http://dx.doi.org/10.1145/2897824.2925982>

¹email: {arowens|mscislak|hartja|pwp}@ucalgary.ca

²email: classenb@uni-mainz.de

or reproductive) meristem when and where there is enough room for them. Depending on the parameters of this process, different regular arrangements (phyllotactic patterns) of primordia emerge [Douady and Couder 1996; Smith et al. 2006]. In some cases, reproductive meristems develop into next-order meristems rather than florets, leading to hierarchically or recursively compounded inflorescences [Classen-Bockhoff and Bull-Hereñu 2013].

The type, form and developmental stage of florets may depend on their position within an inflorescence (floral dimorphism [Weberling 1992]). In general, florets close to the inflorescence margin develop enlarged petals, making the inflorescence more visible to pollinators, while florets in more central positions have comparatively reduced petals allowing for denser packing and thus a larger number of reproductive organs within the inflorescence [Classen-Bockhoff 1992] (English version [Classen-Bockhoff 1994]). Striking examples of floral dimorphism abound in the Aster family. For example, in the sunflower, petals of the showy florets on the margin (ray florets) are orders of magnitude larger than those of florets in the interior of the inflorescence (disk florets). The differences between florets may also be gradual, as illustrated by the photograph and model of *Orlaya grandiflora* in Figure 1, and may occur at different organization levels in compound inflorescences. The developmental mechanisms defining the spatial distribution of florets of different type and size are not yet well understood, but observations of compound heads [Classen-Bockhoff 1992] and wounding experiments on sunflower heads [Hernandez and Palmer 1988] suggest that the inhibition of showy florets by the proximity of other florets is the determining factor at least in some cases. In the context of a regular arrangement of florets into spiral phyllotactic patterns, such inhibition leads to the prevalence of specific numbers of enlarged florets on the inflorescence margin (numerical canalization [Battjes et al. 1993]). For example, in the most common spiral phyllotactic pattern with the golden divergence angle (approx. 137.5°), the prevalent numbers of ray florets belong to the Fibonacci sequence. The availability of space when new primordia are initiated does not guarantee that the florets will not collide during subsequent development. In dense inflorescences such collisions are frequent.

Florets in an inflorescence are supported either by a fleshy voluminous body (the receptacle) or a free-standing branching structure [Weberling 1992]. The former case is exemplified by flower heads (capitula), and is relatively simple from a modeling perspective, as the receptacle can be approximated by a surface of revolution (e.g. [Fowler et al. 1992]). In the latter case, the arrangement of florets in space has traditionally been viewed as a consequence of the underlying branching pattern. This branching-first perspective is useful in modeling practice (e.g. [Frijters 1978; Prusinkiewicz and Lindenmayer 1990; Ijiri et al. 2005]; Figure 8), but does not provide an adequate model for inflorescences in which the ensemble of florets forms a smooth canopy. A quintessential example is that of corymbs, an inflorescence type in which florets are arranged into an almost planar surface (Figure 1). How plants create branching structures satisfying this planarity constraint is an open question, as the problem is nontrivial from a trigonometric perspective. We propose a solution inspired by the current biological understanding of inflorescence development [Reinhardt et al. 2003; O'Connor et al. 2014]. According to it, a growing meristem (Figure 2a) supports emergent primordia (b), which initiate vascular strands (c). These strands gradually extend toward the base of the meristem and merge, forming a branching structure (d). The vascular strand formation is driven by the flow of the plant hormone auxin (auxin canalization¹ [Sachs 1991]) and is analogous to the formation of a branching river network, where tributaries start at different sources

¹Confusingly, the terms *numerical canalization* and *auxin canalization* function concurrently, although they are not closely related.

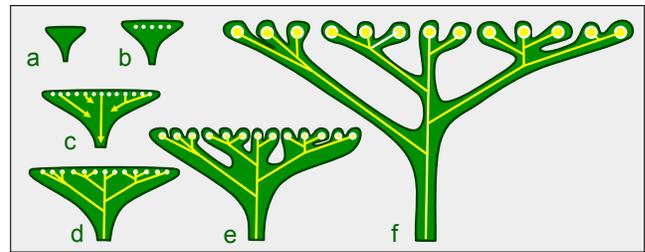


Figure 2: Postulated mechanism of branching inflorescence development. Dots represent florets, yellow lines indicate the vasculature. Vascular strands develop within a growing meristem (a-d) and define the axes of the emerging branching structure (e,f).

and gradually join each other. In heads, the vascular structure remains embedded in the surrounding tissue, which becomes the receptacle as the inflorescence develops to maturity. In branched inflorescences, the vascular strands and the immediately adjacent tissues elongate while tissues further removed from the vasculature do not grow or grow more slowly. A free-standing branching structure thus emerges (e,f). We note that, if the growth of this structure is (approximately) isometric or allometric – in general, if the length of mature segments is proportional to their length in the meristem, and the branching angles do not change or change in concert – the arrangement of florets in the mature inflorescence will reflect their original distribution on the surface of the meristem. For instance, in corymbs, the floret distribution formed on a flat meristem surface will result in a planar canopy. In summary, we assume that the distribution of florets in a predetermined phyllotactic pattern drives the formation of the supporting branching structure, and not vice versa. This florets-first perspective provides a possible explanation for the development of smooth floral canopies. It is also justified from an ecological/evolutionary point of view, as it reflects the importance of flower distribution, rather than the branching structure, to the pollinators. The branching structure is merely a scaffold supporting the flowers in their target positions [Harder and Prusinkiewicz 2013].

Following this description, we propose a method for modeling and animating the development of inflorescences that integrates the following elements: (i) the modeling of individual florets; (ii) the generation of phyllotactic patterns that defines the distribution of florets within floral canopy; (iii) determination of the type, size and developmental stage of each floret according to its position in the inflorescence, (iv) detection and resolution of collisions between petals, and (v) generation of the branching inflorescence structure supporting the florets in space. These elements can be used jointly or selectively, depending on the inflorescence type.

1.2 Overview of the method

The elements of our method can be grouped into three processes: the modeling of individual florets, the generation of their spatial distribution and attributes such as the type and developmental stage, and — when present — the generation of the branching structure that supports the florets. Components of the system and the information flow between them are shown in Figure 3.

Florets are modeled as B-spline surfaces (Section 2). The poses representing key developmental stages for each floret type are specified interactively. This is effected using a specialized graphical editor (Figure 3a), which supports radial and bilateral symmetry as well as partially fused petals found in many florets. The key poses are interpolated to generate sequences representing the de-

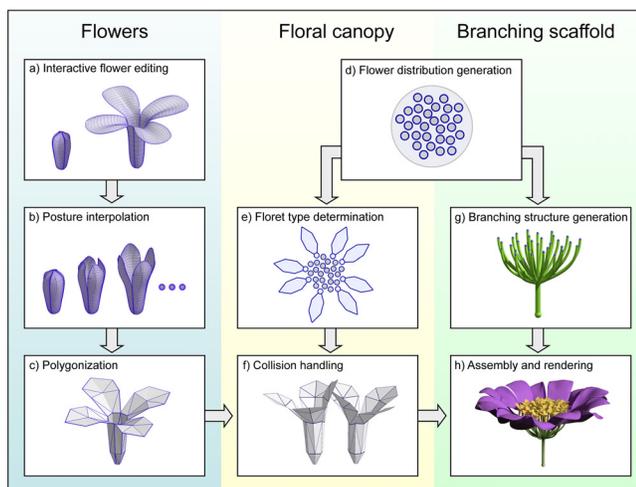


Figure 3: Key components of our modeling method and the associated information flow.

velopment and opening of florets (Figure 3b). The models are then carefully polygonized so that the polygon count in each floret is small and degenerate triangles are avoided (Figure 3c). Low polygon count in individual florets is important to the efficient detection and resolution of collisions in inflorescences with multiple florets. The resulting floret models are instantiated when placed within an inflorescence and may be deformed by collisions between petals.

Collisions may arise in each step of simulated inflorescence development. We resolve them by applying the kinematic description of floret development and opening (the interpolation of key poses) to drive a physically-based model of petal expansion and collision (Figure 3f and Section 3). A related approach, with a kinematic growth model driving a physically-based model, was proposed to animate flower opening by Ijiri et al. [2008]. Given that a floral canopy may include many flowers, we use position-based dynamics [Müller et al. 2007] as a faster alternative to the energy minimization method used by Ijiri et al.

To generate the layout of florets in dense canopies, we extend the algorithm for generating phyllotactic patterns on arbitrary surfaces of revolution proposed by Ridley [1986] and introduced to computer graphics by Prusinkiewicz et al. [2001] (Figure 3d and Section 4). Our extensions capture the dynamics of pattern development and provide information on the developmental stage of each floret within the inflorescence; furthermore, they enable simulation of hierarchically and recursively compound phyllotactic patterns. If the inflorescence is dimorphic, the neighborhood of each floret is inspected and the floret type is determined on this basis (Figure 3e and Section 5).

If florets are supported by a receptacle, distributing florets of the appropriate type according to the phyllotactic pattern and resolving collisions terminates the modeling process. The receptacle, usually obscured by the florets, does not need to be represented explicitly. In contrast, in branched inflorescences, the branching structure supporting the floral canopy must be visualized. We generate it using the spatial layout and age of the florets as input (Figure 3g and Section 6). Our algorithm is inspired by that introduced by Rodkaew et al. [2003] to model leaf vein patterns and trees – both algorithms create branching structures from the outside in – extending it in several directions as needed to model inflorescences. The floral canopy and the supporting branching structure are rendered together to produce the final image (Figure 3h).

2 Interactive modeling of flowers

Previous work. There is a long history of modeling flower petals as interactively defined bicubic surfaces (e.g. [Prusinkiewicz and Lindenmayer 1990; Fowler et al. 1992]) or generalized cylinders [Lintermann and Deussen 1999; Prusinkiewicz et al. 2001], and assembling them into flowers using procedural models of phyllotaxis. Ijiri et al. [2005] advanced the concept of interactive flower modeling by introducing a specialized editor to distribute flower parts in space. They also pioneered sketch-based modeling of plant organs such as leaves and petals [Ijiri et al. 2005; Ijiri et al. 2006]; further work in this direction was pursued by Anastacio et al. [2006].

At any point in time, an inflorescence may incorporate a progression of flowers at different developmental stages, a phenomenon termed the phase effect by d’Arcy Thompson [1942] (see also [Prusinkiewicz et al. 1988; Prusinkiewicz and Lindenmayer 1990]). Simulation of flower growth and opening is thus needed not only to animate flower or inflorescence development, but also to construct static models of inflorescences with the phase effect. To simulate the opening of flowers modeled as bicubic surfaces, Prusinkiewicz et al. [1993] constructed a branching structure that supported the set of control points defining the surface and gradually modified this structure over time. This method amounted to a forward simulation of growth, making the final form of fully open flowers difficult to control. With petals represented as generalized cylinders, Prusinkiewicz et al. [2001] modeled flower growth by interpolating intrinsically-defined carrier curves that represented petal axes (midribs) in closed and open flowers. Our method is related to this concept. A different, physically-based approach was proposed by Ijiri et al. [2008] and improved by Li et al. [2015]. They represented petals as elastic surfaces subject to non-uniform expansion. This expansion affected both the size and shape of the petals. Related methods have also been used in biologically-motivated simulations [Liang and Mahadevan 2011; Green et al. 2010], in the latter case addressing the particularly complex shape of snapdragon flowers. Physically-based techniques are appealing because of their sound biological basis and possible emergence of secondary features, such as wrinkles on the margin of the petals, which are otherwise difficult to model [Li et al. 2015]. On the negative side, both the dynamics of flower opening and the final shape are difficult to specify.

The complexity of inflorescences results from the arrangement of florets, rather than the individual floret forms. Consequently, we devised a specialized interactive editor to quickly model simple flowers in their key poses (stages of development) and a blending technique to interpolate between these poses.

Floret editor. We focus on the modeling of corolla (the set of petals), which is the most visible part of florets. A snapshot of the editor is shown in Figure 4. Petals are represented as clamped B-spline surfaces. To facilitate the modeling process, bilateral symmetry can be imposed on the individual petals, and petals can be multiplied by assuming n -fold radial symmetry. One new element is the modeling of partially fused petals, which are found in many flowers. We implemented it for florets with dihedral (i.e., both rotational and bilateral) symmetry, by constraining a user-defined number of control points defining the petal boundary (starting at the flower base) to the symmetry plane between the adjacent petals.

Simulation of floret development. We assume that the key floret poses are represented by B-spline surfaces defined by control meshes with the same topology (numbers of control points in the u and v directions). Such surfaces can easily be blended by linearly interpolating positions of the corresponding control points, but linear interpolation is not appropriate for simulating the opening of flowers because petals may undergo large rotations. Addressing

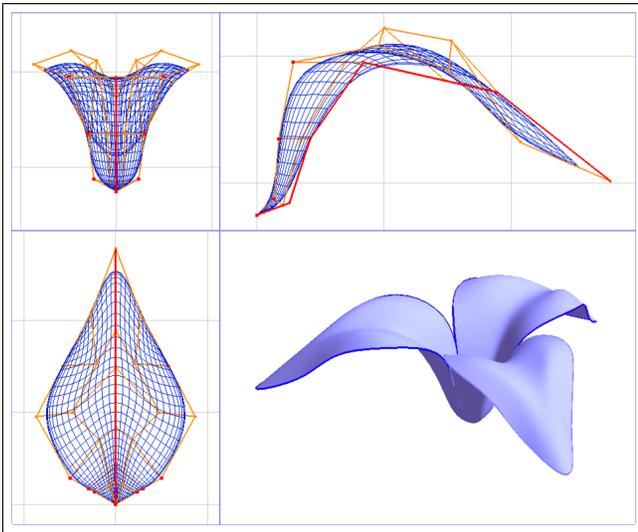


Figure 4: Snapshot of the floret editor. The user defines the shape of the petal by moving control points in the standard front, top and side view. The bottom right view shows the corolla with the petals replicated by rotational symmetry and partially fused.

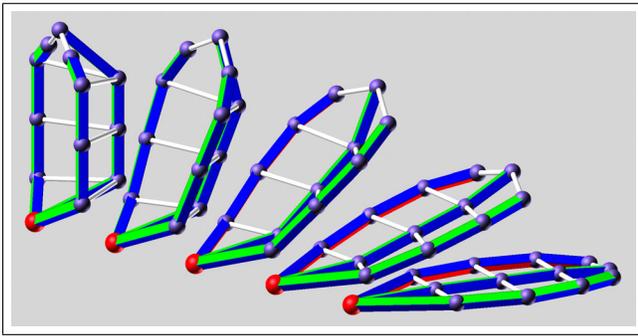


Figure 5: Example of control mesh interpolation. In each pose, control polylines in the v direction are represented by boxes, the colored faces of which are aligned with a rotation-minimizing moving frame. White lines indicate control polylines in the u direction, spanning the vertices of the v polylines.

this problem, we interpolate floret poses by extending the method of blending intrinsically defined polygons [Sederberg et al. 1993] to 3D. To this end, we calculate a rotation-minimizing frame [Bloomenthal 1990] for each (open) control polyline running in the v direction, from the petal base to its tip (Figure 5). At each vertex between consecutive line segments, this frame is rotated around the axis perpendicular to both segments so that the next segment lines up with the previous one (for collinear segments this rotation is 0). We then blend corresponding control polygons in the initial and fi-

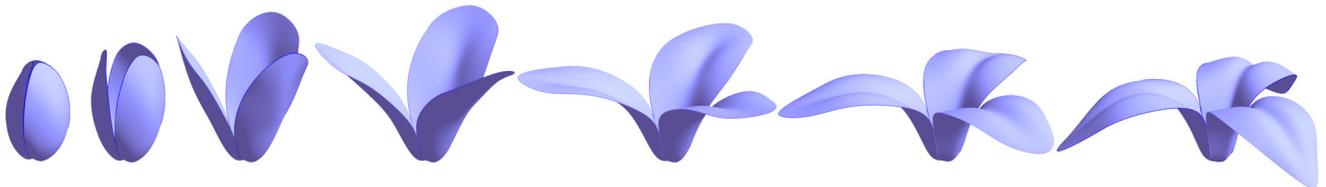


Figure 6: Simulation of floret opening by intrinsic interpolation between the first and the last pose.

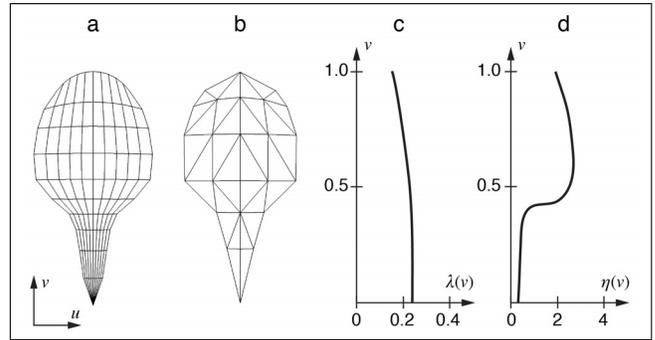


Figure 7: Example of petal polygonization. (a) Default polygonization using a family of isoparametric lines $u = \text{const}$ and $v = \text{const}$, equidistant in the parameter space. (b) A sample result of repolygonization. (c,d) Graphically-defined functions used to create the repolygonization in panel b. Note that the axis representing the independent variable v runs vertically.

nal poses by linearly interpolating the lengths of the corresponding segments, and spherically interpolating the rotations between them. This technique makes it possible to simulate the development and opening of flowers using a minimal number of key poses, usually only two or three (Figure 6).

Polygonization. An inflorescence may comprise many florets. To make the subsequent collision detection and resolution efficient it is thus critical to polygonize them well, into a small number of non-degenerate triangles. The simplest polygonization of tensor product surfaces, producing an array of quadrangles by stepping through the u and v parameters, is inadequate for this purpose, because the shape of the resulting polygons varies greatly between narrow and wide petal regions (Figure 7a). Consequently, we repolygonize the surfaces using a semi-interactive method, in which the modeler can fine-tune the polygonization with two graphically-defined functions (Figure 7c,d). The first function, $\lambda(v)$, defines the spacing between a sequence of vertices v_0, v_1, \dots, v_n along the v axis. The distances between these vertices are calculated by solving the equation

$$\int_{v_i}^{v_{i+1}} \frac{dv}{\lambda(v)} = 1, \quad i = 0, 1, \dots, n-1, \quad (1)$$

which distributes points along the axis according to average values of function $\lambda(v)$ in each interval $[v_i, v_{i+1}]$ and guarantees that this distribution is robust (not sensitive to small perturbations of λ) [Prusinkiewicz et al. 2001]. The second function, $\eta(v)$, defines the number $\lceil \eta(v) \rceil$ of vertices, equidistant in the parameter space, along each isoparametric line $v = v_i$. The final polygon mesh is obtained as the Delaunay triangulation of the resulting set of points (u, v) (Figure 7b). This triangulation is performed in parameter space, so that the mesh topology does not change as the floret develops.

3 Collision detection and resolution

Previous work. In early models of flowers and inflorescences (e.g. [Prusinkiewicz and Lindenmayer 1990; Fowler et al. 1992; Prusinkiewicz et al. 2001]) the modeler minimized the visual impact of intersections between petals by carefully crafting their shapes. More recently, Ijiri et al. [2008] outlined a specialized algorithm for handling collisions between petals or sepals, which exploited their spatial ordering as determined by the flower structure. In inflorescences a similar assumption of spatial ordering cannot be made; moreover, an inflorescence may include hundreds or thousands of florets, making the number of polygons representing the inflorescence as a whole correspondingly higher. Consequently, we have adapted to inflorescences a more general method of collision detection and resolution originally developed for cloth, but also applied to leaves [Müller et al. 2007]. In contrast to cloth, the set of polygons representing an inflorescence can be divided into subsets, which represent individual organs [Zhao and Barbič 2013]. We take advantage of this observation by considering different florets of the same type as instances — modified by collisions — of a common dynamic geometric template (Section 2).

Collision detection. We assume that the initial configuration of florets in a young inflorescence is collision-free, then detect and resolve collisions as they arise during the simulated development. To this end, we follow the continuous collision detection (CCD) paradigm (cf. [Brochu et al. 2012]). With petals represented as triangle meshes growing over time, we need to find intersections between triangular prisms representing the development-driven motion of triangles. We consider vertex-triangle and edge-edge intersections separately, and decompose each case into a broad and narrow phase. The broad phase is accelerated by partitioning space into a regular grid. For vertex-triangle intersections, each voxel stores the IDs of all triangles that may intersect this voxel over a given time step. This is done by constructing axis-aligned bounding boxes (AABB) containing the space-time prisms of the triangles as they move through space, and intersecting them with the voxels. The boxes are enlarged by a small amount to address potential numerical inaccuracies when detecting intersections. The information in each voxel is time-stamped so that out-of-date voxels can be cleared efficiently, only if visited in a subsequent time step [Teschner et al. 2003]. To detect whether a moving vertex may intersect a triangle, an AABB is created around the line segment representing the vertex motion. All triangles in the voxels that intersect with this AABB are considered in the narrow phase. Broad-phase detection of potential edge-edge collisions is carried out in a similar manner, except that, in this case, voxels contain information about the edges which intersect them during their motion through space, rather than the entire triangles. Narrow-phase collision detection is performed using the method of Provot [1997], by detecting whether, during a given time interval, the four vertices representing a moving point and a triangle, or the endpoints of two edges, become co-planar. If this is the case, a test is performed to determine whether the moving point is within the triangle, or whether the intersection point is within both segments, respectively.

Collision resolution. We resolve collisions by adapting position-based dynamics (PBD) [Müller et al. 2007; Bender et al. 2014] to the simulation of growth. Consistent with the PBD method, we map mesh M_0 representing the initial (collision-free) floral canopy into a network of constraints. Distance constraints correspond to the edges of this mesh and represent their length. Angular constraints correspond to the signed dihedral angles between all pairs of adjacent triangles in the mesh. Attachment constraints define positions and orientations of individual florets in the canopy (see Section 4). We denote the resulting set of constraints C_0 . We then simulate the growth of the inflorescence and the opening of flowers.



Figure 8: A model of a lilac inflorescence with collisions resolved.



Figure 9: A zoom into the lilac inflorescence from Figure 8 modeled with collision resolution turned off and on. Two flowers have been false-colored to facilitate comparisons.

To this end, we iteratively progress through the sequence of polygonized meshes M_0, M_1, \dots, M_n representing consecutive stages of the inflorescence canopy development. In each step i , we use the edge lengths, dihedral angles and attachment positions in mesh M_i to define the corresponding constraint set C_i . If one or more collisions occur, we add constraints preventing triangle intersection to the set C_i before resolving it. Once a collision-free steady state is found, we advance developmental time and progress to the next simulation step, $i + 1$. Our simulation is thus similar to the typical application of PBD to simulate cloth, except that not only collision constraints, but also all other constraints, may change from one simulation step to the next as the inflorescence grows.

Figure 8 shows a sample inflorescence model with collisions between petals eliminated. We have used a previously published model of lilac inflorescence architecture [Prusinkiewicz et al. 2001] to support florets in space. A magnified view of a portion of this inflorescence (Figure 9) highlights intersections that would occur if collisions were not addressed, and shows how the algorithm resolves them by displacing individual florets and deforming their petals. Although neglecting collisions does not significantly affect the overall shape of the inflorescence, collision detection and resolution is essential to properly represent details in close-up views.

4 Phyllotactic pattern generation

In the remainder of this paper we focus on plants of the aster and carrot families (Asteraceae and Apiaceae), in which florets are arranged into spiral phyllotactic patterns. We recreate these patterns by extending a previous model [Prusinkiewicz et al. 2001] to simulate the dynamics of pattern formation. Dynamic simulation opens the door to animating the development of inflorescences, and improves the realism of static models by capturing the age and size differences between the florets. In addition, we consider the formation of hierarchical and recursive phyllotactic patterns.

Previous work. Phyllotactic patterns have fascinated researchers for almost two centuries by their conspicuous regularity and intriguing mathematical properties, such as the emergent occurrence of Fibonacci numbers and the golden ratio. The beauty of inflorescences exhibiting diverse phyllotactic patterns has also made them an attractive modeling subject in computer graphics. The first models [Fowler et al. 1989; Prusinkiewicz and Lindenmayer 1990] were based on analytic descriptions of spiral organ packing on the surface of a cylinder [Erickson 1983] or disk [Vogel 1979]. Using ideas similar to Vogel’s, Lintermann and Deussen [1999] developed an analytic description of spiral phyllotactic patterns on the surface of a sphere. Fowler et al. [1992] and Prusinkiewicz et al. [2001] extended the range of modeled patterns to primordia of varying size distributed on an arbitrary surface of revolution. The latter approach, based on an extension of Vogel’s model by Ridley [1986], is well suited for computer graphics applications due to its flexibility and robustness.

Ridley assumed that the supporting surface is defined by rotating a planar generating curve C around the surface axis (Figure 10). The position of primordium i is expressed by two parameters: angle α_i with respect to a reference direction, and position s_i along the generating curve. Ridley proposed to calculate the displacement $s_{i+1} - s_i$ of primordium $i + 1$ with respect to primordium i by equating primordium area A_i with the area of a circular band circumscribing the supporting surface at the elevation of these primordia. This leads to the equation [Prusinkiewicz et al. 2001]:

$$\int_{s_i}^{s_{i+1}} 2\pi r(s) ds = A_i \quad \text{or} \quad \int_{s_i}^{s_{i+1}} \frac{2\pi r(s)}{A_i} ds = 1, \quad (2)$$

where $r(s)$ is the distance of point s on the generating curve C from the axis of the supporting surface. Given the initial position s_0 and a sequence of primordium areas $\{A_i\}$, Equation 2 (left or right) can be solved iteratively for s_1, s_2, \dots , yielding the sequence of primordium positions $\{s_i\}$ along the generative curve C . In modeling practice it is usually more convenient to define the primordium areas as a function $A(s)$ of their positions s rather than index i , which results in

$$\int_{s_i}^{s_{i+1}} \frac{2\pi r(s)}{A(s)} ds = 1. \quad (3)$$

With consecutive primordia positioned along the generative curve according to Equation 3, and the divergence angle $\alpha_{i+1} - \alpha_i$ between consecutive primordia equal to the golden angle $\varphi \approx 137.5^\circ$,

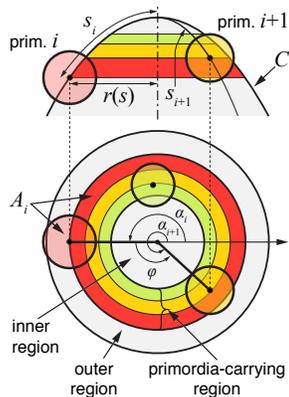


Figure 10: Geometric elements of the extended Ridley phyllotaxis model.

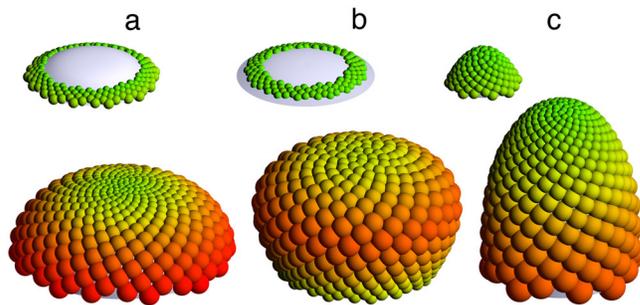


Figure 11: Earlier and later stages of phyllotactic pattern formation. (a) Centripetal fractionation. (b) Divergent fractionation. (c) Segregation. Colors indicate the age of primordia (green: younger; red: older).

the primordia are packed in a dense spiral phyllotactic pattern [Ridley 1986]. We define function $A(s)$ interactively to make the distribution of floret sizes in the generated pattern match that observed in real inflorescences used for reference.

Modeling the dynamics of phyllotaxis. We extend Ridley’s model to capture not only the static distribution of primordia, but also the dynamics of their production and distribution over time. From the dynamic perspective, a plant may produce florets in two distinct modes, termed fractionation and segregation [Classen-Bockhoff and Bull-Hereñu 2013]. In the case of fractionation, the meristem is initially bare or “naked”, and becomes gradually covered by the emerging primordia. In the case of segregation, new primordia originate near the tip of a continuously growing meristem when and where room is created for them.

We simulate two distinct cases of fractionation, as well as segregation, using a unified model. The reproductive meristem is divided into three regions: the naked inner and outer regions, and the primordium-carrying intermediate region (Figure 10). In centripetal fractionation (Figure 11a), commonly observed in developing heads, the outer region is absent, and the inner region initially occupies the entire meristem. Beginning at its rim, consecutive primordia are produced at fixed time intervals at positions determined by Equation 3, building up the primordium-carrying region at the expense of the inner region from the outside in. This process terminates when the area of the inner region falls below that of a single primordium. In divergent fractionation [Harris et al. 1991] (Figure 11b), both the inner and outer regions are initially present. Primordium formation begins at the boundary between these regions and proceeds simultaneously inward and outward. Finally, in the case of segregation (Figure 11c), the initial conditions are similar to those in the centripetal fractionation, but the meristem is small, commensurate with a primordium. A new primordium is produced when the area of the growing inner region exceeds a threshold value. The inner region’s area is then reduced by the area of the primordium, falling below the threshold until further growth increases it again. A sequence of primordia is thus produced, with the period dependent on the growth rate of the inner region.

In general, both the meristem and the primordia may grow over time. The growth rates of the inner and outer regions are specified as functions of the meristem age. Likewise, the growth of each primordium is a function of its age. The modeler defines all functions using a graphical function editor. The same function is used for all primordia of the same type.

The last element of the phyllotaxis model is the growth of the supporting surface – the surface of a meristem or a receptacle – as a whole. To model changes in its size and shape, we interpolate

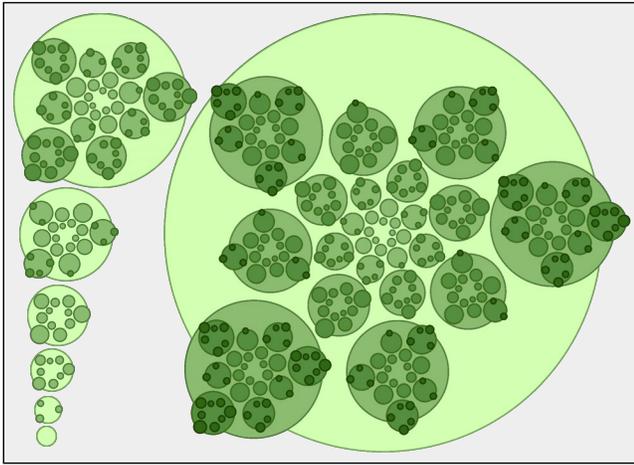


Figure 12: Development of a recursive, self-similar phyllotactic pattern on a uniformly growing flat meristem.

between graphically-defined generating curves \mathcal{C} that describe the profile of the supporting surface at select points in time. Furthermore, to coordinate the size of the supporting surface with the size of its regions, we sum up the current areas of all regions present (the primordium-carrying area is calculated by adding up the areas of all primordia) and scale the supporting surface to match that sum. The growth and changes in the shape of the supporting surface are featured in the simulations in Figure 11.

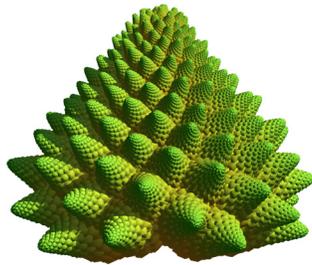


Figure 13: Two-level hierarchical phyllotaxis with conical meristems.

pattern elements at different levels: consistent with the Hofmeister rule the first element at level $n + 1$ within the parent meristem at level n is positioned as far as possible from the centre of the “grand-parent” meristem at level $n - 1$ (cf. [Kirchoff 2003]). Figure 13 shows a two-level pattern, in which meristems have an approximately conical shape.

5 Determining floret type

Previous work. As discussed in Section 1, inflorescences may comprise florets of different types, thus exhibiting floral dimorphism. In computer graphics practice to date, dimorphism has been modeled in the context of heads: a user-specified number of florets at the rim of the head was assigned the fate of petal-like ray florets, and the remaining florets were modeled as disk florets [Prusinkiewicz and Lindenmayer 1990; Fowler et al. 1992]. Inspired by [Hirmer 1931], Battjes and Prusinkiewicz [1998] proposed a mathematical model of ray floret differentiation based on the packing of primordia on the rim of a head. They demonstrated that for the divergence angle $\varphi \approx 137.5^\circ$ the numbers of ray florets are numerically canalized to the Fibonacci series, as often observed

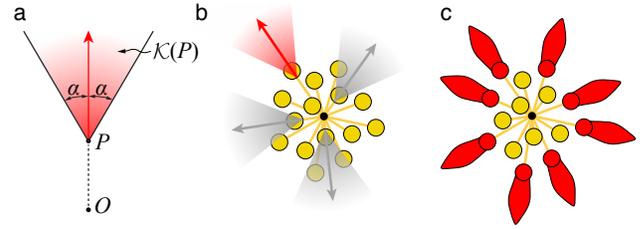


Figure 14: Principle of evaluating floret fate in dimorphic inflorescences. (a) Definition of cone $\mathcal{K}(P)$ associated with floret P in a head or umbellet with centre O . The cone has opening angle α . (b) The fate of a floret depends on other florets that may be within its cone. (c) The simplest case of floret differentiation: a floret becomes a ray floret if no other floret is within its cone.

in real heads. Neither model suffices, however, to capture dimorphism in compound inflorescences, in which the fate of each floret may depend on its position within the inflorescence in a relatively complex manner. Addressing this limitation, we propose an algorithm that applies to both simple and compound inflorescences. The algorithm is based on the biological hypothesis that the fate of florets depends on available space [Classen-Bockhoff 1992].

The model. Our algorithm operates under the assumption that the floral canopy is approximately planar. Each floret P in the set \mathcal{S} of all florets is associated with a cone $\mathcal{K}(P)$ originating at P (Figure 14a). This cone is oriented away from the centre O of the highest-level head or umbellet to which P belongs. In the simplest case, the fate of floret P is determined by the presence or absence of another floret in the cone $\mathcal{K}(P)$ (Figure 14b,c). In general, this fate may also depend on the minimum distance $d_{min}(P)$ between P and a floret $Q \in \mathcal{K}(P)$. In order to find this distance, we first calculate the distances $d(P, Q)$ between P and all florets $Q \in \mathcal{S}$ using the formula:

$$d(P, Q) = \begin{cases} \|Q - P\| & \text{if } \widehat{PQ} \cdot \widehat{OP} < \cos \alpha \\ \infty & \text{otherwise} \end{cases} \quad (4)$$

Here \widehat{AB} denotes the normalized vector from A to B , and α is the opening angle of cone $\mathcal{K}(P)$. The distance $d(P, Q)$ is thus set formally to infinity if floret Q lies outside this cone. The minimum distance $d_{min}(P)$ is then calculated as

$$d_{min}(P) = \min_{Q \in \mathcal{S}} d(P, Q). \quad (5)$$

Application of this formula to simple flower heads is illustrated in Figures 15 and 16. All patterns were generated assuming divergence angle $\varphi = 137.5^\circ$ and primordia of fixed size. In both figures, a floret P becomes a disk floret upon $d_{min}(P) < \infty$. Figure 15 shows the dependence of the number of ray florets N on the total number of florets n , assuming a constant opening angle. We observe that the numbers of ray florets which emerge on heads of different sizes tend to be Fibonacci numbers. A similar bias towards Fibonacci numbers occurs if the number of florets n is fixed and the cone opening angle α is changed instead (Figure 16). Fibonacci numbers occur for large ranges of α values.

Examples of floret fate determination in a hierarchically compound inflorescence are shown in Figure 17. In all cases, the fate of florets is controlled by distance thresholds. In Figure 17a, threshold Th_a is larger than the distance between florets within a second-order head, but smaller than the distance between these heads. A floret P becomes a ray floret if and only if $d_{min}(P) > Th_a$. In this case, ray florets differentiate independently in each second-order head. In Figure 17b, floret P becomes a ray floret if and only if

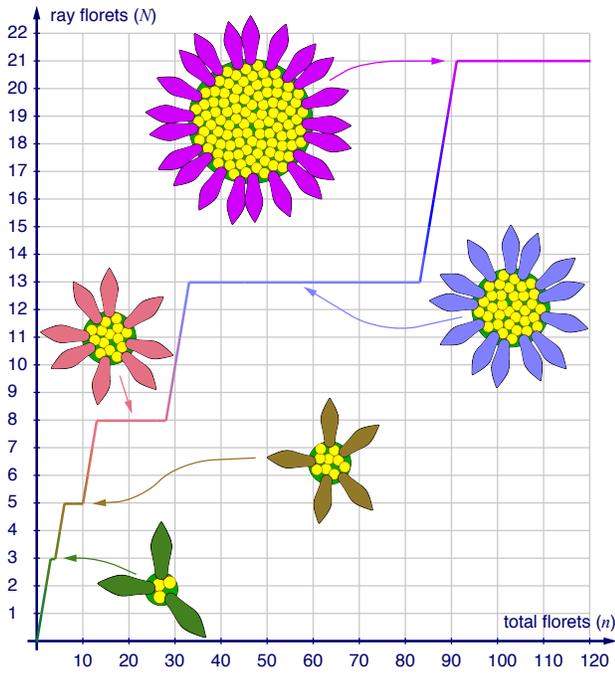


Figure 15: Dependence of the number of ray florets N on the total number of florets n for a constant cone opening angle $\alpha = 75^\circ$.

$d_{min}(P) > Th_b$, where Th_b is larger than the distance between second-order heads. In this case, the differentiation of ray florets is dominated by their position within the entire inflorescence. The model in Figure 17c combines both mechanisms. Large ray florets emerge when $d_{min}(P) > Th_b$, and smaller ones when $Th_a < d_{min}(P) \leq Th_b$. In the latter case, the size of the enlarged petals is proportional to the amount of available space, $d_{min}(P)$, resulting in a continuum of florets sizes.

6 Branching structure generation

Previous work. In inflorescences in which florets are supported by a branching structure (as opposed to a receptacle), this structure is an inherent component of the model. Computational modeling of branching inflorescence structures was an early biological [Frijters 1978] and computer graphics [Prusinkiewicz et al. 1988; Prusinkiewicz and Lindenmayer 1990] application of L-systems. The use of positional information [Prusinkiewicz et al. 2001] facilitated specification of inflorescences by providing the modeler with direct control over the extent and density of branches. Sketch-based methods [Ijiri et al. 2005; Ijiri et al. 2006; Anastacio et al. 2006] facilitated modeling further by introducing a more intuitive user interface. With all these approaches, the distribution of flowers in space was determined by the underlying branching structure. Such

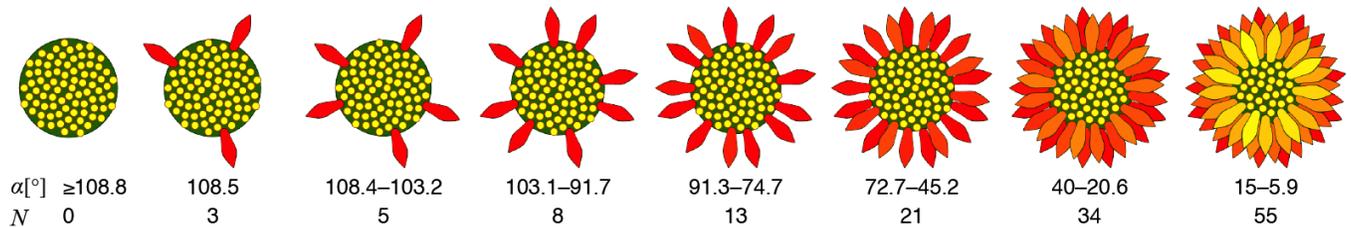


Figure 16: Dependence of the number of ray florets N on the cone opening angle α for a constant total number of florets $n = 82$.

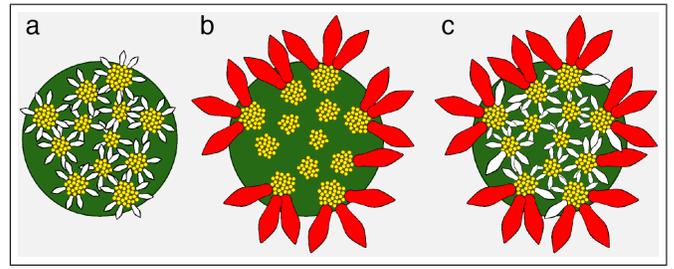


Figure 17: Differentiation of florets in hierarchically organized inflorescences. Ray florets emerge: (a) at the level of second-order heads, (b) at the level of the entire inflorescence, and (c) at both levels of inflorescence organization. In all cases $\alpha = 53^\circ$.

branching-first models work well for many inflorescences (e.g., the lilac in Figure 8), but do not easily capture inflorescences with florets arranged into a smooth, planar canopy. To model these inflorescences we employ the florets-first paradigm (Section 1 and Figure 2), in which a spatio-temporal floret distribution is generated first and provides input for synthesizing the branching structure. Our method is inspired by the algorithm for generating vascular patterns in leaves and modeling trees proposed by Rodkaew et al. [2003] (see also [Neubert et al. 2007]). Rodkaew's algorithm operates centripetally, i.e., from the outside in. For example, in the case of trees, it begins by distributing leaves at the periphery of the tree canopy, initiates branches at these locations, and gradually extends them towards the trunk. Rodkaew et al. described this process in terms of particles that trace branches by moving through space while being attracted to the base of the tree and to their nearest neighbor. Particles that approach each other merge, forming branching points. Our choice of Rodkaew's algorithm as a basis for modeling the architecture of flat-topped branching inflorescences was motivated by two factors. At the microscopic level it can be viewed as a geometric analog of auxin canalization, which defines the vascular system of young inflorescences. It is plausible that this vasculature defines the branching structure of inflorescences in reality (Section 1). At the macroscopic level, Rodkaew's algorithm is a stepping stone for generating three-dimensional branching structures that support florets in predefined positions.

Extended Rodkaew algorithm. To model inflorescences, we extended Rodkaew's algorithm in several directions.

Attraction to branches. According to the original formulation of the algorithm, particles interact with each other: they do not interact with the emerging branching structure. Consequently, a moving particle may run unnaturally close to a branch formed earlier and/or cross it. To avoid this artifact, we consider not only the moving particles, but also the entire structure developed so far (i.e., past positions of all particles) as candidate attracting points (Figure 18a). Furthermore, we consider the previous direction of particle motion \vec{H} as a factor influencing the next direction \vec{H}' . Assuming that

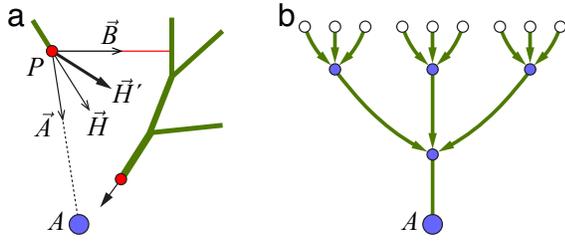


Figure 18: Elements of the extended Rodkaew algorithm. (a) Particle P extends a branch segment in direction \vec{H}' , which is a weighted average of the previous segment direction \vec{H} , vector \vec{A} pointing toward the inflorescence base A , and vector \vec{B} pointing towards the nearest branch or particle. (b) The umbellets of a compound umbel are generated separately from the main umbel. Both the positions of the florets (empty circles) and the umbellet and umbel centers (small blue circles) are determined by a phyllotaxis model.

all vectors defined in Figure 18a have unit length, direction \vec{H}' is calculated in each simulation step as

$$\vec{H}' = \frac{1}{w_a + w_b + w_h} (w_a \vec{A} + w_b \vec{B} + w_h \vec{H}). \quad (6)$$

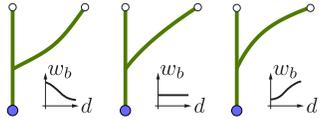


Figure 19: The dependence of branching point configuration on function $w_b(d)$.

Weights w_a , w_b and w_h provide a degree of control over generated forms. With $w_a > 0$ and $w_b = 0$, all particles converge directly on the inflorescence base, creating an umbel (Figure 20a). Increasing w_b results in branching structures (Figure 20c,e). Making w_b a

function of the distance d to the closest neighbor provides a means for controlling the shape of branches near the branching points, producing smaller or larger branching angles (Figure 19). Increasing weight $w_h > 0$ prevents biologically improbable [Kim et al. 2012] highly non-planar branch arrangements at the branching points. It also reduces the curvature of branches, making them more smooth.

Particle initialization. In the original formulation, the initial positions of the particles are random, and all particles begin their motion simultaneously. To model inflorescences, we assume that initial particle positions and the times of their creation are determined by the dynamic phyllotactic model (Section 4). The timing of floret emergence has a significant impact on the resulting branching structure. With primordia created at large time intervals, during which particles can travel relatively long distances compared to the distances between primordia, the emerging branches have a monopodial architecture characterized by clearly delineated

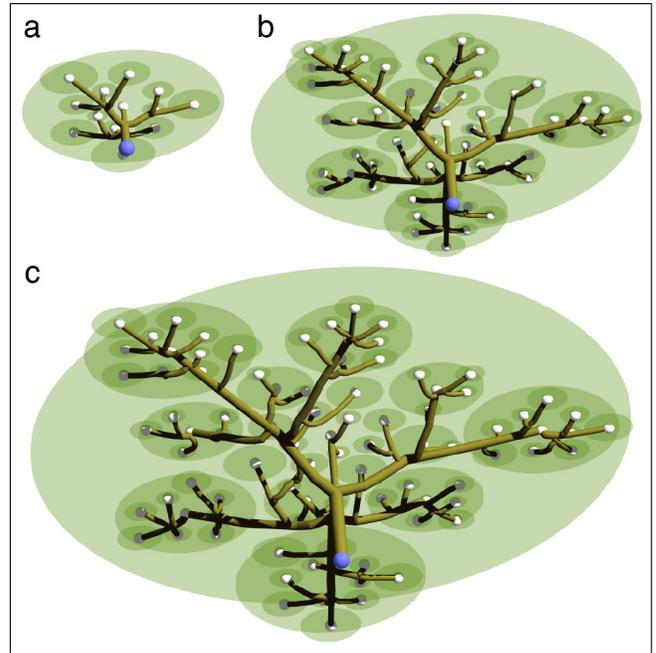


Figure 21: A growing branching structure generated using the extended Rodkaew algorithm. Florets (white spheres) are formed in a recursive phyllotactic pattern and emerge at different points in time according to the model in Figure 12. The emergent branching structure supports all florets in the same plane, as needed for modeling corymbs.

axes (Figure 20b,c). In contrast, if primordia are created quickly, branches have a sympodial architecture characterized by sequences of short segments positioned laterally with respect to their parents (Figure 20d,e).

Growth. We model growth of the branching inflorescence structure as a gradual free-form deformation [Sederberg and Parry 1986] of the space including branches and moving particles. This deformation is coordinated with the dynamic phyllotactic patterning of floral primordia and the canopy they produce. In the models implemented so far, we assumed uniform or allometric expansion of this space. In the allometric case, this space expands at different, size-dependent rates along different axes of the coordinate system, resulting in changes to the branching angles and overall proportions of the inflorescence over time. These deformations do not affect branch width, which is determined independently. An example of the development of a branching structure supporting primordia in a recursive phyllotactic pattern is shown in Figure 21.

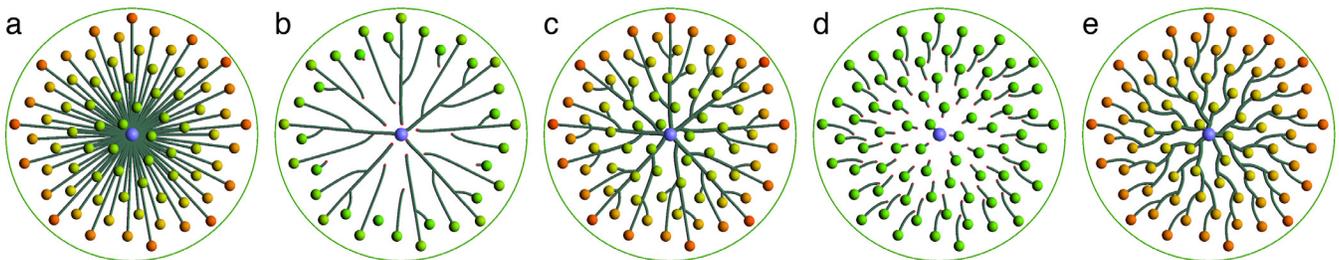


Figure 20: Examples of branching structures generated using the extended Rodkaew algorithm. (a) All particles are attracted to the inflorescence base. (b) Early and (c) final stages of pattern formation with primordia produced slowly. (d) Early and (e) final stages of pattern formation with primordia produced quickly. Colors of primordia indicate their age, as in Figure 11.



Figure 22: Photograph and model of dahlia ‘Jomanda’. Photograph courtesy of the Victoria Dahlia Society, adapted under fair use.

Distinct hierarchy levels. Some inflorescences (e.g. corymbs of heads and compound umbels) are organized into distinct hierarchical levels. We generate branching patterns of these inflorescences by considering each level of the hierarchy separately (Figure 18b).

Branch width. We assume that terminal branch segments have the same diameter and determine the diameter of interior segments using the formula $d^p = d_1^p + \dots + d_m^p$. Here d is the diameter of the branch below the branching point (closer to the inflorescence base), and $d_1 \dots d_m$ are the diameters of the branches supported at this point. Power p controls the rate at which branch widths accumulate towards the inflorescence base. Variants and extensions of this formula have been used in different contexts [Shinozaki et al. 1964; MacDonald 1983; Prusinkiewicz and Lindenmayer 1990]; a special case, with $p = 2$, was proposed by Leonardo da Vinci to capture the relation between branch diameters in trees. In the modeling of inflorescences, smaller changes in diameter obtained for $p > 2$ usually lead to more realistic results.

7 Implementation

We devised the modeling software within the Virtual Laboratory², which provides an environment for exploratory programming and experimentation with simulation models. The presented algorithms are implemented as a suite of programs that communicate via shared files. The files transferred from one program to the next represent the entire developmental sequence, as opposed to a single developmental stage. The flow of information is shown in Figure 3. The input to the models consists of: (i) key poses of the florets, specified interactively using the floret editor (Section 2); (ii) two graphically-defined functions that define the polygonization of petals for each floret type (Section 2); (iii) three graphically-specified functions that define the growth of the floret primordia and the inner and outer regions of the supporting surface (Section 4); (iv) two profile curves that specify the initial and final shape of this supporting surface (Section 4); and (v) a small number (about 5 per program) of numerical parameters controlling each element of the pipeline. The amount of input increases in hierarchical structures,

²<http://algorithmicbotany.org/virtual.laboratory/>

where phyllotactic patterns at different levels of the hierarchy may require separate definition. Models are assembled by a C++ program that instantiates florets of the type defined by the dimorphism model at the locations and orientations defined by the phyllotaxis model, simulates inflorescence development while resolving collisions, and produces a temporal sequence of meshes that represents the developing floral canopy (Figure 3f). This canopy is integrated with the branching structure at the rendering time (Figure 3h). We have rendered final models using Blender³. To improve rendering, we employed Blender to subdivide petals and slightly extrude them to allow for subsurface light scattering. We have also incorporated scanned textures and environment maps from the sIBL Archive⁴.

8 Results

Collision detection is the central feature of the dahlia model in Figure 22. This particular variety has a ball-shaped inflorescence populated almost exclusively by incurved, almost tubular ray florets that touch each other and accommodate their shape to the presence of their neighbors. The distribution of florets in the model was generated using a phyllotaxis model operating on an approximately spherical surface.

Gaillardia (Figure 23) is an example of an inflorescence with strongly dimorphic florets. In the variety shown, ray florets are tubular. The emergent number of ray florets in the model (21) is consistent with their numerical canalization in real *Gaillardia* heads, where Fibonacci numbers commonly occur. The phyllotactic patterning of floret primordia was modeled using the fractionation scheme (Figure 11a). In addition to disk and ray florets, this model incorporates involucral bracts: specialized leaves that surround the head and protect it in its early development. Hairs on all surfaces have been modeled using a built-in Blender function. Figure 24 shows a simulation of head development, with the receptacle changing size and shape. All three organs types were animated using intrinsic interpolation and were subject to collisions.

³<https://www.blender.org/>

⁴<http://www.hdrlabs.com/>



Figure 23: Photograph and model of *Gaillardia x grandiflora* cultivar 'Oranges and Lemons'. Photograph licensed under CC BY-SA 3.0.



Figure 24: Selected frames from an animation of *Gaillardia* growth.



Figure 25: A model of a sunflower.

The sunflower (Figure 25) is another example of an inflorescence with strongly dimorphic florets. As in the case of *gaillardia*, the emergent number of ray florets in the model (34) is a Fibonacci

number. Ray florets are bilaterally symmetric. Three large fused petals point outward, giving the appearance of a single large petal. The remaining two petals are highly reduced. We have only included the conspicuous large petals in the model. Densely packed disk florets have five-fold dihedral symmetry, with petals fused into a tubular corolla almost up to the petal tips. As the flowers open, sexual organs - stamens, then pistils - raise above the level of the petals [Sammataro et al. 1986]. We modeled these organs summarily as cylinders growing along the central axis of the florets. The florets open in a centripetal sequence, so that outer florets are more advanced in their development than those in more central positions. This phase effect has also been captured in the model.

The next three models illustrate different cases of floral dimorphism. The inflorescence of *Dyssodia* (Figure 27) consists of a central head surrounded by 5 other heads. Florets with enlarged petals emerge at the level of the entire inflorescence as in Figure 17b [Classen-Bockhoff 1992]. Incidentally, the modification of disk floret shapes due to collisions is particularly conspicuous in this model (Figure 26).



Figure 26: A magnified view of *Dyssodia*'s florets.



Figure 27: *Photograph and model of Dyssodia decipiens.*



Figure 28: *Photograph and model of a yarrow. Photograph courtesy of Frank L. Hoffman, <http://www.all-creatures.org>.*

Yarrow (Figures 28 and 29, left) is a typical example of a corymb inflorescence. The branching structure supports small heads with disk and ray florets. Their dimorphism has been captured using the model in Figure 17a. The seemingly irregular shape of the yarrow inflorescence was accurately captured by the recursive phyllotaxis algorithm shown in Figure 12. The branching structure (Figure 29, left), supporting the florets in a single plane, was modeled using the extended Rodkaew algorithm: as illustrated in Figure 21, but with parameters promoting a more elongated structure. Leaves were generated using the positional-information method described in [Prusinkiewicz et al. 2001].

The model of orlaya (Figures 1 and 29, right) integrates most elements discussed in this paper. The inflorescence is a compound umbel. Both the umbellets within the main umbel and the individual florets within the umbellets exhibit spiral phyllotaxis. The florets in inward positions within the umbellets are small and have five-fold symmetry. Select outer florets have enlarged petals, with the size depending on the available space. Finally, select florets on the periphery of the whole inflorescence have very large petals. The di-

morphism model in Figure 17c accurately captures this multiplicity of forms. The florets are supported in a single plane by a branching structure organized into two levels of whorls. We reproduced this branching system by modeling each level separately (Figure 18b).

9 Discussion

We have advanced previous methods for modeling inflorescences by introducing methods for: (i) modeling and animating florets with (partially) fused petals; (ii) detecting and resolving collisions that may occur in densely packed inflorescences using position-based dynamics extended to growing surfaces; (iii) extending Ridley's phyllotaxis model to simulate and animate the dynamics of phyllotaxis in simple, hierarchical, and recursive patterns; (iv) algorithmically determining floret type, size, and developmental stage in dimorphic inflorescences; and (v) modeling branching inflorescence structures as a self-organizing process driven by the distribution of florets. This process is particularly useful when modeling inflorescences with a smooth floral canopy, such as corymbs. Vi-



Figure 29: Branching structures of yarrow (corymb of heads) and orlaya (compound umbel) modeled using the extended Rodkaew algorithm.

sual evaluations suggest that our methods can capture the form and animate the development of diverse inflorescences. This makes our methods applicable to computer imagery and supports the biological hypotheses at the foundation of our methods. From a broader perspective, our results demonstrate the usefulness of the geometric approach to morphogenesis, where the emergence of patterns and forms is described in geometric rather than chemical terms.

Our work exposed many open problems deserving further research. They include more detailed modeling of flowers, detection and resolution of collisions between all elements of the plant structure (not only between petals), and view-dependent model generation. The latter would optimize viewing by adjusting the level of detail to the scale at which the models are presented, and suppressing the generation of elements that lie outside the field of view or are obscured. Our method for simulating flower opening by blending key poses works well in practice, but does not conserve petal area, and could potentially be improved by using current methods for shape-preserving mesh transformation, e.g., [Lipman et al. 2007]. Furthermore, our collision-detection method is satisfactory when the initial configuration is collision-free, but in some cases this assumption is difficult to satisfy. Problems occur, for example, when florets are packed in a bud. A possible solution may be to place all elements of the initial inflorescence in an artificial yet easy-to-specify collision-free configuration, then procedurally assemble all elements into the desired structure while detecting and resolving collisions. From a broader perspective, a similar technique could lead to the solution of another challenging problem in computer graphics: modeling bouquets of flowers.

Acknowledgements

We thank Adam Runions, Lawrence Harder and Faramarz Samavati for stimulating discussions, Pascal Ferraro and Brendan Lane for contributions to the Virtual Laboratory software, Lynn Mercer for editorial help, and the referees for their insightful, constructive comments. Support of this research by grants from the Natural Sciences and Engineering Research Council of Canada, Canada First Research Excellence Fund (PP), and Deutsche Forschungsgemeinschaft (RC-B) is gratefully appreciated.

References

ANASTACIO, F., COSTA SOUSA, M., SAMAVATI, F., AND JORGE, J. 2006. Modeling plant structures using concept sketches.

In *Proceedings of the 4th International Symposium on Non-photorealistic Animation and Rendering*, ACM, 105–113.

- BATTJES, J., AND PRUSINKIEWICZ, P. 1998. Modeling meristic characters of Asteracean flowerheads. In *Symmetry in Plants*, D. Barabé and R. Jean, Eds. World Scientific, Singapore, 281–312.
- BATTJES, J., VISCHER, N., AND BACHMANN, K. 1993. Capitulum phyllotaxis and numerical canalization in *Microseris pygmaea* (Asteraceae: Lactuceae). *American Journal of Botany*, 419–428.
- BENDER, J., MÜLLER, M., OTADUY, M. A., TESCHNER, M., AND MACKLIN, M. 2014. A survey on position-based simulation methods in computer graphics. *Computer Graphics Forum* 33, 6, 228–251.
- BLOOMENTHAL, J. 1990. Calculation of reference frames along a space curve. In *Graphics Gems*, A. Glassner, Ed., vol. 1. Academic Press, Boston, 567–571.
- BROCHU, T., EDWARDS, E., AND BRIDSON, R. 2012. Efficient geometrically exact continuous collision detection. *ACM Transactions on Graphics (TOG)* 31, 4, 96.
- CLASSEN-BOCKHOFF, R., AND BULL-HEREÑU, K. 2013. Towards an ontogenetic understanding of inflorescence diversity. *Annals of Botany* 112, 1523–1542.
- CLASSEN-BOCKHOFF, R. 1992. Florale Differenzierung in komplex organisierten Asteraceenköpfen. *Flora* 186, 1-2, 1–22.
- CLASSEN-BOCKHOFF, R. 1994. Functional units beyond the level of the capitulum and cypsela in Compositae. In *Compositae: Biology and Utilization. Proceedings of the International Compositae Conference*, P. Caligari and D. Hind, Eds., vol. 2. Royal Botanic Gardens, Kew, 129–160.
- DOUADY, S., AND COUDER, Y. 1996. Phyllotaxis as a dynamical self organizing process. Parts I–III. *Journal of Theoretical Biology* 178, 255–312.
- ERICKSON, R. O. 1983. The geometry of phyllotaxis. In *The growth and functioning of leaves*, J. E. Dale and F. L. Milthorpe, Eds. University Press, Cambridge, 53–88.
- FOWLER, D., HANAN, J., AND PRUSINKIEWICZ, P. 1989. Modelling spiral phyllotaxis. *Computers & Graphics* 13, 3, 291–296.
- FOWLER, D. R., PRUSINKIEWICZ, P., AND BATTJES, J. 1992. A

- collision-based model of spiral phyllotaxis. *Computer Graphics* 26, 2, 361–368.
- FRIJTERS, D. 1978. Principles of simulation of inflorescence development. *Annals of Botany* 42, 549–560.
- GREEN, A. A., KENNAWAY, J. R., HANNA, A. I., BANGHAM, J. A., AND COEN, E. 2010. Genetic control of organ shape and tissue polarity. *PLoS Biology* 8, e1000537.
- HARDER, D., AND PRUSINKIEWICZ, P. 2013. The interplay between inflorescence development and function as the crucible of architectural diversity. *Annals of Botany* 112, 1477–1493.
- HARRIS, E., TUCKER, S., AND URBATSCH, L. 1991. Floral initiation and early development in *Erigeron philadelphicus* (Asteraceae). *American Journal of Botany*, 108–121.
- HERNANDEZ, L., AND PALMER, J. 1988. Regeneration of the sunflower capitulum after cylindrical wounding of the receptacle. *American Journal of Botany*, 1253–1261.
- HIRMER, M. 1931. Zur Kenntnis der Schraubenstellungen im Pflanzenreich. *Planta* 14, 1, 132–206.
- IJIRI, T., OWADA, S., OKABE, M., AND IGARASHI, T. 2005. Floral diagrams and inflorescences: Interactive flower modeling using botanical structural constraints. *ACM Transactions on Graphics* 24, 3, 720–726.
- IJIRI, T., OWADA, S., AND IGARASHI, T. 2006. Seamless integration of initial sketching and subsequent detail editing in flower modeling. *Computer Graphics Forum* 25, 3, 617–624.
- IJIRI, T., YOKOO, M., KAWABATA, S., AND IGARASHI, T. 2008. Surface-based growth simulation for opening flowers. In *Proceedings of Graphics Interface 2008*, Canadian Information Processing Society, 227–234.
- KIM, Y., SINCLAIR, R., CHINDAPOL, N., KAANDORP, J., AND DE SCHUTTER, E. 2012. Geometric theory predicts bifurcations in minimal wiring cost trees in biology are flat. *PLoS Computational Biology* 8, 4, e1002474.
- KIRCHOFF, B. 2003. Shape matters: Hofmeister’s rule, primordium shape, and flower orientation. *International Journal of Plant Sciences* 164, 4, 505–517.
- LI, J., LIU, M., XU, W., LIANG, H., AND LIU, L. 2015. Boundary-dominant flower blooming simulation. *Computer Animation and Virtual Worlds* 26, 3-4, 433–443.
- LIANG, H., AND MAHADEVAN, L. 2011. Growth, geometry, and mechanics of a blooming lily. *Proceedings of the National Academy of Sciences* 108, 14, 5516–5521.
- LINTERMANN, B., AND DEUSSEN, O. 1999. Interactive modeling of plants. *IEEE Computer Graphics and Applications* 19, 1, 56–65.
- LIPMAN, Y., COHEN-OR, D., GAL, R., AND LEVIN, D. 2007. Volume and shape preservation via moving frame manipulation. *ACM Transactions on Graphics* 26, 1, 5.
- MACDONALD, N. 1983. *Trees and Networks in Biological Models*. J. Wiley & Sons, New York.
- MÜLLER, M., HEIDELBERGER, B., HENNIX, M., AND RATCLIFF, J. 2007. Position based dynamics. *Journal of Visual Communication and Image Representation* 18, 2, 109–118.
- NEUBERT, B., FRANKEN, T., AND DEUSSEN, O. 2007. Approximate image-based tree-modeling using particle flows. *ACM Transactions on Graphics (TOG)* 26, 3, 88.
- O’CONNOR, ET AL. 2014. A division in PIN-mediated auxin patterning during organ initiation in grasses. *PLoS Computational Biology* 10, 1, e1003447.
- PROVOT, X. 1997. Collision and self-collision handling in cloth model dedicated to design garments. In *Proceedings of the Eurographics Workshop on Computer Animation and Simulation*, Springer, 177–189.
- PRUSINKIEWICZ, P., AND LINDENMAYER, A. 1990. *The Algorithmic Beauty of Plants*. Springer, New York. With J. Hanan, F. Fracchia, D. Fowler, M. de Boer, and L. Mercer.
- PRUSINKIEWICZ, P., LINDENMAYER, A., AND HANAN, J. 1988. Developmental models of herbaceous plants for computer imagery purposes. *Computer Graphics* 22, 4, 141–150.
- PRUSINKIEWICZ, P., HAMMEL, M. S., AND MJOLSNESS, E. 1993. Animation of plant development. In *Proceedings of SIGGRAPH 93*, Annual Conference Series, 351–360.
- PRUSINKIEWICZ, P., MÜNDERMANN, L., KARWOWSKI, R., AND LANE, B. 2001. The use of positional information in the modeling of plants. In *Proceedings of SIGGRAPH 2001*, Annual Conference Series, 289–300.
- REINHARDT, D., ET AL. 2003. Regulation of phyllotaxis by polar auxin transport. *Nature* 426, 255–260.
- RIDLEY, J. N. 1986. Ideal phyllotaxis on general surfaces of revolution. *Mathematical Biosciences* 79, 1–24.
- RODKAEW, Y., CHONGSTITVATANA, P., SIRIPANT, S., AND LURSINSAP, C. 2003. Particle systems for plant modeling. In *Plant growth modeling and applications. Proceedings of PMA03*, B.-G. Hu and M. Jaeger, Eds. Tsinghua University Press and Springer, Beijing, 210–217.
- SACHS, T. 1991. *Pattern Formation in Plant Tissues*. Cambridge University Press, Cambridge.
- SAMMATARO, D., GARMENT, M., AND ERICKSON JR, E. 1986. Anatomical features of the sunflower floret. *Reprints — US Department of Agriculture, Agricultural Research Service*.
- SEDERBERG, T., AND PARRY, S. 1986. Free-form deformation of solid geometric models. *Computer Graphics* 20, 4, 151–160.
- SEDERBERG, T., GAO, P., WANG, G., AND MU, H. 1993. 2-D shape blending: an intrinsic solution to the vertex path problem. In *Proceedings of the SIGGRAPH*, ACM, 15–18.
- SHINOZAKI, K., YODA, K., HOZUMI, K., AND KIRA, T. 1964. A quantitative analysis of plant form — the pipe model theory. I. Basic analyses. *Japanese Journal of Ecology* 14, 3, 97–105.
- SMITH, R., KUHLEMEIER, C., AND PRUSINKIEWICZ, P. 2006. Inhibition fields for phyllotactic pattern formation: A simulation study. *Canadian Journal of Botany* 84, 1635–1649.
- TESCHNER, M., HEIDELBERGER, B., MÜLLER, M., POMERANTES, D., AND GROSS, M. 2003. Optimized spatial hashing for collision detection of deformable objects. In *VMV*, vol. 3, 47–54.
- THOMPSON, D. 1942. *On Growth and Form*, 2nd edition. University Press, Cambridge.
- VOGEL, H. 1979. A better way to construct the sunflower head. *Mathematical Biosciences* 44, 179–189.
- WEBERLING, F. 1992. *Morphology of flowers and inflorescences*. Cambridge University Press, Cambridge.
- ZHAO, Y., AND BARBIČ, J. 2013. Interactive authoring of simulation-ready plants. *ACM Transactions on Graphics (TOG)* 32, 4, 84.