Interactive Synthesis of 3D Geometries of Blood Vessels

N. Rauch and M. Harders

Interactive Graphics and Simulation Group, University of Innsbruck, Austria

Abstract

In surgical training simulators, where various organ surfaces make up the majority of the scene, the visual appearance is highly dependent on the quality of the surface textures. Blood vessels are an important detail in this; they need to be incorporated into an organ's texture. Moreover, the actual blood vessel geometries also have to be part of the simulated surgical procedure itself, e.g. during cutting. Since the manual creation of vessel geometry or branching details on textures is highly tedious, an automatic synthesis technique capable of generating a wide range of blood vessel patterns is needed. We propose a new synthesis approach based on the space colonization algorithm. As extension, physiological constraints on the proliferation of branches are enforced to create realistic vascular structures. Our framework is capable of generating three-dimensional blood vessel networks in a matter of milliseconds, thus allowing a 3D modeller to tweak parameters in real-time to obtain a desired appearance.

CCS Concepts

• ***Computing methodologies*** → *Computer graphics;* • ***Applied computing*** → ***Life and medical sciences;***

1. Introduction

Blood vessel models are essential for surgical training simulations. First of all, a surgeon needs to know the location and diameter of blood vessels when manipulating tissue. In order to simulate bleeding caused by cutting procedures, vessels (and their network geometry) have to be incorporated into the tissue deformation model. Next, blood vessels are an essential detail in the visual appearance of an organ's surface and are therefore an integral component in the visualization of a surgical scene. While texture creation could be carried out manually by an artist, it is a highly tedious task, which is further complicated due to the diversity of vascular patterns in real life. Therefore, the modelling of an organ's appearance by applying surface textures to a geometric model, through automatically generated blood vessel networks, would be a tremendous benefit. Finally, it is also vital that training scenes vary from one to another to improve the learning. Otherwise, a trainee could recall specifics of a certain setting, e.g. the vessel locations. Because vessel networks are complex by nature, creating many of such tailored scenes also requires an automatic synthesizing framework. Motivated by these considerations we introduce a novel method for synthesizing blood vessels in a matter of milliseconds, based on a modified space colonization algorithm [RFL*05].

One of the most prominent biological growth mechanisms of blood vessels is the process of *angiogenesis*. It is initiated when cells undergo a level of hypoxia and as a response secrete growth factors into the tissue. Existing blood vessels in the proximity sprout and proliferate into the gradient direction of the growth factor concentration to reach said tissue regions and support them with oxygen – at which point the process arrests. This growth mechanism resembles a competition for space between blood vessels, which is a behaviour often observed in nature as the underlying cause for the formation of branching patterns. In order to model this behaviour we extend the space colonization algorithm, originally introduced for leaf venation [RFL*05] [RLP07]. We modify the growth process to follow the physiological vessel bifurcation principles by Murray (see Section 2), and also incorporate the interdependent development of the venous system.

Our system permits the creation of realistic blood vessels by enforcing local geometry constraints during development. Due to this, we can maintain the benefit of the fast computation times of the algorithm, while avoiding the need for any global optimization procedures [HJ10] or data-driven models [BMB*16]. Thus, we trade physiological realism for interactive control of the visual output. From the resulting topology, vessel geometries and then organ textures can be automatically synthesized. Various parameters provide a modeller with control over the visual appearance. In addition, desired growth paths as well as areas of oxygen deficiency can be manually included. Two exemplary results are depicted in Figure 1; the textures were created with our proposed method, applied in a rendering of a colon and the conjunctiva of an eye.

For further material: https://igs.uibk.ac.at/EG/Suppl.pdf.

2. Physiological Background

Blood vessel modelling algorithms often enforce morphological constraints on the geometry of the vessel trees, which stem from physical and observed principles. Most notably, the shrinkage of vessel radii and the branching angles in bifurcations are governed by physiological laws; the most prominent being *Murray's Law*. Originally it was derived from the optimality principles of mini-

DOI: 10.2312/egs.20211012

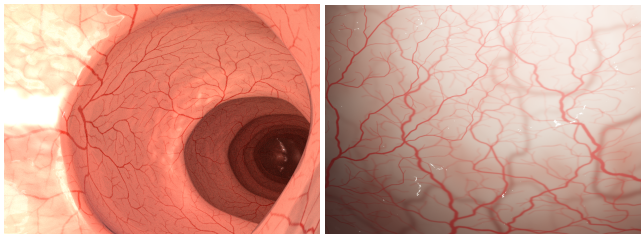


Figure 1: *Rendered images of a colon (left) and the conjunctiva in an eye (right), with vessel trees synthesized in under one second.*

mum work and minimum energy dissipation [Mur26]. In a bifurcation it relates the radius of the parent branch r_p to the radii of its daughter branches r_ℓ and r_r; its generalized version is given by $r_p^\kappa = r_\ell^\kappa + r_r^\kappa$, where κ denotes the bifurcation exponent. Murray, also derived the following relationships for optimal bifurcation angles α, β, by optimizing for the minimum volume in a junction:

$$\cos(\alpha) = \frac{r_p^4 + r_\ell^4 - r_r^4}{2r_p^2 r_\ell^2}, \quad \cos(\beta) = \frac{r_p^4 - r_\ell^4 + r_r^4}{2r_p^2 r_r^2}. \tag{1}$$

3. Vessel Tree Synthesis

Following the space colonization algorithm, free space, i.e. tissue devoid of any vessels, is represented by a set of discrete points, referred to as *attraction points*. Blood vessels develop on a per node basis, i.e. each vessel node decides, based on attraction points in their proximity, if and how it should grow to reach them. The fundamental concept of the method is a feedback loop, in which attraction points guide the development of the branching structure. This in turn controls the regions in which attraction points newly emerge [RFL*05]. Both an arterial and a vessel tree are grown in parallel. The algorithm iterates through five steps. We first provide an overview, while details are addressed thereafter:

1. New attraction points are sampled within the tissue (i.e. oxygen drains representing small hypoxic tissue regions).
2. In response to these drains, the arterial blood vessel system develops to satisfy the oxygen demand of the tissue. The growth is geometrically constrained to satisfy the functional properties of blood vessel networks; i.e., growth directions follow the optimal bifurcation principles derived by Murray, while also avoiding unnatural bending. Further, vessel radii are also adjusted according to Murray's Law – similar to Runions et al. [RFL*05].
3. Next, oxygen drains deemed as sufficiently supplied by close arterioles are added as carbon-dioxide sources, i.e. new points in the venous tree, affecting growth of the latter. This captures the interdependence of the arterial and venous systems. In response to the venous attraction points the venous blood vessel network develops in the same way as the arterial one.
4. The overall tissue domain undergoes growth, as specified by a growth function. This models the organ development, and ensures fractal-like similarities.

The vascular system is represented by a directed graph $G = (N, E)$, consisting of nodes $n \in N$ and edges $e \in E$. An edge or link, denoted by $\langle i, j \rangle$, describes the connection between two nodes i and j, and is oriented from the root to the leaves of the tree. A node n is associated with a position in three-dimensional space $\mathbf{p}_n$. The length of an edge $\langle i, j \rangle$ is the Euclidean distance $l_{ij} = \|\mathbf{p}_j - \mathbf{p}_i\|$. The edge $\langle i, j \rangle$ is referred to as the parent edge of node j, and its direction is given by $\mathbf{e}_{ij} = nrm(\mathbf{p}_j - \mathbf{p}_i)$ (with $nrm(.)$ being a normalization operator). Furthermore, an edge is given a radius r_{ij}.

Oxygen Drain Placement: The first simulation step consists of repeatedly generating new oxygen drains at locations within the tissue. These act as initiators for the growth of the arterial blood vessels, i.e. the latter will proliferate to reach them. Regions that are in proximity to some arterial node are considered as satisfied in terms of oxygen support. Thus, positions sampled closer than a defined threshold distance to any existing arterial node are discarded. Furthermore, a minimal distance between oxygen drains is enforced. In the same manner as in the original algorithm, the generation of attraction points permits to restrict vessel growth to a certain region, e.g. inside a volume or along the surface of an organ.

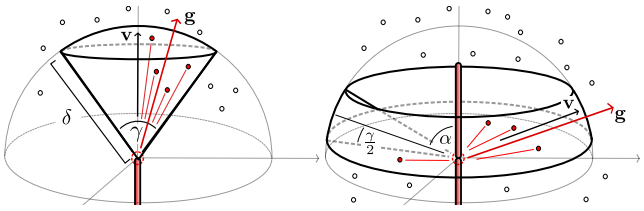
Vessel Proliferation: Every *leaf node* j has a conical *perception volume* (see Figure 2a), defined by the angle γ and distance δ. Only attraction points inside of this volume are considered to influence a node's growth. This is the case if an attraction point at position $\mathbf{p}_o$ satisfies the following two conditions for its distance r and angle θ:

$$r = \|\mathbf{p}_o - \mathbf{p}_j\| \leq \delta, \quad \theta = \cos^{-1}\left(\mathbf{e}_{ij} \cdot nrm(\mathbf{p}_o - \mathbf{p}_j)\right) \leq \gamma/2. \tag{2}$$

In contrast, in case node j is an *inter-node* (i.e. has only one child), the perception volume is aligned with the theoretically optimal branching according to Murray's derivation, with user-defined radius r and angle α. In the third dimension this branch can be thought of as a vector, freely rotating around the parent edge $\mathbf{e}_{ij}$. This causes the perception volume to resemble a spherical segment, as depicted in Figure 2b. An attraction point is inside such a spherical zone if the angle θ and distance r satisfy the conditions:

$$\alpha - \gamma/2 \leq \theta \leq \alpha + \gamma/2, \qquad r \leq \delta. \tag{3}$$

Instead of enforcing a strict theoretical optimum, this approach gives a node an approximate direction into which to look. Note that in reality blood vessels also deviate from the optimal direction. Thus, a user can control the visual appearance by changing the above parameters and thresholds.



(a) leaf node **(b)** inter-node

Figure 2: *Perception volumes of vessel nodes; attraction points inside these influence the growth process.*

Any oxygen drain can only influence the growth of its closest vessel node. If the set $\mathcal{S}(j)$ of drains closest to node j is not empty, a new node k connected to j with an edge $\langle j, k \rangle$ is created. Node k is placed at a user-defined distance d away from $\mathbf{p}_j$, into the average direction $\mathbf{g}$ of all attraction points in $\mathcal{S}(j)$ i.e. $\mathbf{p}_k = \mathbf{p}_j + d \cdot nrm(\mathbf{g})$. If the perception volume is large, unrealistic bending of vessels could occur. We counteract such artifacts by setting a weighting parameter $\omega \geq 0$ which further biases the growth direction $\mathbf{g}$ towards

the perception volume direction $\mathbf{v}$:

$$\mathbf{g} = \omega\mathbf{v} + (1-\omega) \sum_{\mathbf{p}_s \in S(j)} nrm(\mathbf{p}_s - \mathbf{p}_j). \quad (4)$$

For a *leaf node* or *inter-node*, $\mathbf{v}$ is the direction of the parent edge $\mathbf{e}_{ij}$ or the hypothetical optimal branch closest to the average attraction vector, respectively (see Figures 2a and 2b).
Further, in order to develop the characteristic symmetric bifurcations often found in real blood vessel trees a second growth process for leaf nodes is introduced. It is applied when the standard deviation of the angles formed by the attraction vectors is larger than a predefined threshold ϕ. A symmetric bifurcation is then created by computing the two optimal angles according to Murray, with the predefined terminal radius r. The two branches are placed within the plane given by the leaf node's position $\mathbf{p}_n$ and the line $\mathbf{l}$, which minimizes the orthogonal distances to every attraction point $s \in \mathcal{S}(n)$. Thus, these new segments are aligned with the spread of the attraction points. The line $\mathbf{l}$ goes through the mean of all attraction points $\mathbf{c} = \frac{1}{n}\sum_{i=0}^{n} \mathbf{s}_i$. Computing the direction of the line equates to finding the most significant principal component of the attraction point coordinates. Let $\mathbf{X}$ be the $3 \times n$ matrix with each column x_i containing the centralized attraction points $(\mathbf{s}_i - \mathbf{c})$. The sought vector $\mathbf{l}$ is then the eigenvector with the largest corresponding eigenvalue of the co-variance matrix $\frac{1}{n}XX^T$.

Vessel Tree Updates: Every time a lateral branch or a bifurcation has been created the radii of all parent edges up to the root have to be adjusted. To this end Murray's law is applied with a user-defined bifurcation exponent $2.0 < \alpha < 4.0$. After the arterial development, all oxygen drains in close proximity, i.e. user-defined distance, of an arterial node are changed to carbon-dioxide sources, which in turn guide the growth of the venous vessels. Assuming in our model that only tissue receiving oxygenated blood will produce waste products, the growth of the venous system is directed to reach regions of said tissue, while ignoring areas that are not yet provided with nutrients. The final step in the approach is the tissue growth, i.e. simulating the expansion of space in the tissue due to cell proliferation. Our framework is capable to include any complex growth model; however, for simplicity we currently only include a spatially uniform scaling. This permits all calculations to be performed with relative coordinates – positions of points and nodes relative to the tissue domain. However, we need to scale the unit size of the parameters, i.e. threshold distances, as well as the terminal vessel radius r and the growth length d. With the change in size per time step $\Delta\sigma$, we compute per time instant t the scaling factor $\sigma_t = \sigma_{t-1} + \Delta\sigma$, and the corresponding unit size with $\lambda_t = \lambda_0 \frac{\sigma_0}{\sigma_t}$. Multiplying the relative coordinates with σ_t thus yields the positions at time t. As a result of the parameter scaling, the growth process is increasingly localized with respect to the entire simulation domain. This establishes the fractal properties of blood vessel trees, i.e. repetition of similar patterns on smaller scales.

4. Results and Visualizations

The restriction of the development of vessels to a certain domain via placement of attraction points is highlighted in Figure 3; two arterial blood vessel trees are grown – only on the surface or inside the volume (the combination of both is also indicated). It should be noted that due to the piece-wise linear nature of triangle mesh

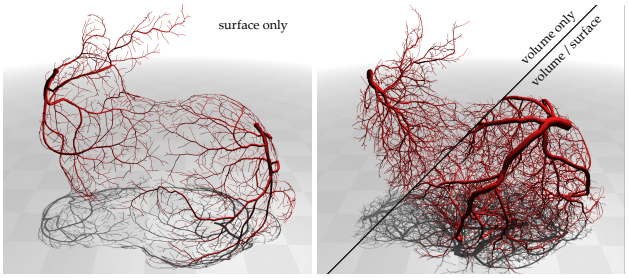


Figure 3: *Placement of oxygen drains controls vessel growth domain – on the surface (left) or inside the volume/combined (right).*

geometries, surface vessels could potentially be located outside the domain. This could be alleviated by projecting the vessel geometry onto the nearest surface. For proliferation on an organ surface, attraction points are sampled on the mesh primitives. For growth inside of a volume, we voxelize the surface mesh and place samples into the resulting voxels. These ideas can be combined by first sampling on the surface, and then growing inwards (see in right part of Figure 3). Following this notion, a user can guide the growth along manually predefined paths, before the automatic sampling of the tissue region occurs. This facilitates the generation of characteristic large scale vessels on organs (Figure 4).
Runions et al. [RFL*05] generated their varying patterns by modifying threshold distances and the leaf growth model. Our model allows for additional control, through influencing the local geometry constrains via three parameters. To illustrate this further, Figure 5 depicts growth examples of three blood vessel trees within the same circular domain (diameter of one), using different parameters. The results after 100 iterations, with $\Delta\sigma = 0.02$ and 1000 arterial attraction points per time step, are shown. Subfigures 5a and 5b illustrate the arterial and venous blood vessels, developed inter-dependently with default parameters ($\approx$ 6500 vessel nodes in total). The most influential parameter in terms of altering the visual appearance is the bifurcation exponent κ. Due to its direct relation to the shrinkage of vessel radii it has a significant effect on the computed bifurcation angles. Higher values increase the contribution of child segments to its parent radius and thus increase the difference between their radii. Consequently, the bifurcation angles throughout a vessel tree get larger (Subfigure 5c). The influence of the growth bias direction is controlled by weighting parameter ω. Larger values encourage the development of straighter instead of tortuous vessel trees (Subfigure 5d). Finally, the threshold ϕ controls the development of symmetric bifurcations. Higher values lead to more prominent lateral branches, while lower values increase the amount of sym-

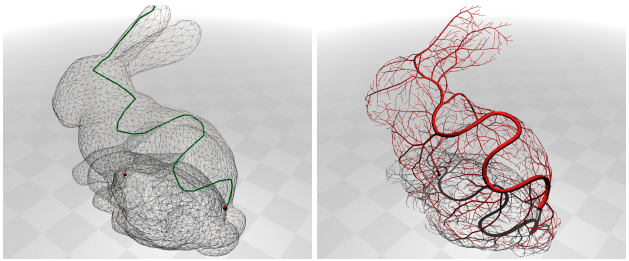


Figure 4: *Guiding initial growth along user defined paths (green) permits modelling of characteristic large-scale vessels.*

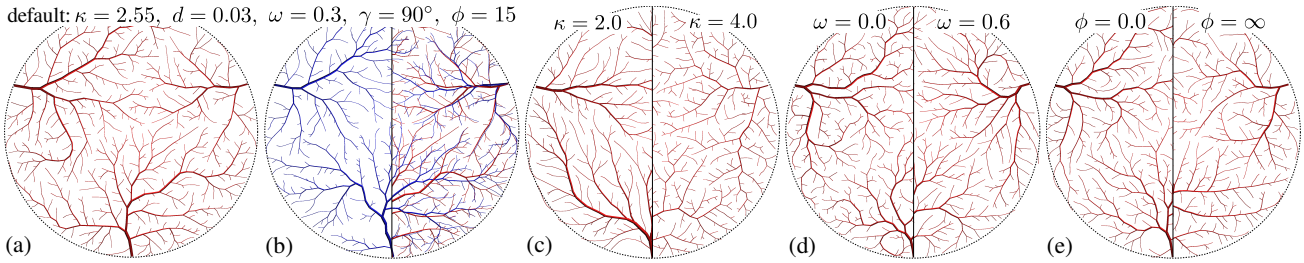


Figure 5: *Three arterial vessel trees in a circular tissue domain with varying parameters controlling vessel proliferation.*

metric bifurcations (Subfigure 5e).

These examples were generated with a standard Desktop PC (Core i7), with the tree synthesis taking on average 75.84 ± 6.27 ms. Note that no optimizations were performed, except for the use of an Octree implementation for the nearest neighbour search. The latter represents the main bottleneck of the algorithm. Thus, any parameter settings that increase the number of attraction points will increase the overall computation time.

The algorithm parameters exhibit some inter-dependencies and require expertise to control. Similar to the original space colonization algorithm, distance parameters and thresholds must be chosen with each other in mind. For instance, the perception distance should be larger than the threshold for satisfying oxygen drains. A noticeable visual artifact can appear with a large growth distance d: vessels overshoot their target and collide with others vessels (Figure 6 left). Other unrealistic patterns can emerge if growth constraints are too low or too high, e.g. for a too large perception volume angle γ or a too strong direction bias ω, as shown in Figure 6, center and right.

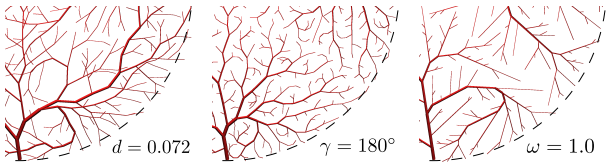


Figure 6: *Artifacts due to inappropriate parameter selections.*

The vessel networks resulting from the growth algorithm are in the end converted into triangle meshes. For a fast preview of the vascular structures we employ an abstract geometry representation, with cubes aligned and scaled according to each vessel edge. Even though this does not properly represent bifurcations, it provides a good overall impression of the tree, while being computationally efficient (e.g. via instanced rendering). For the creation of the final, closed, high-resolution mesh, we employ a more sophisticated geometry construction method [HBC*10]. A comparison of the meshes is shown in Figure 7. Furthermore, the vessel trees can be used to generate two-dimensional textures, encapsulating vessels in close proximity to the organ's surface. To this end, we iteratively project the vessel geometries onto each mesh primitive and map its rasterization into a texture.

We highlight the potential of the growth framework and the generated textures by re-creating real world examples, as illustrated in Figure 1. For the inside of the colon the vascular network follows typical theoretical characteristics of blood vessels in terms of branching angles and bending. The vessel growth of the conjunctiva of the eye was more difficult and requires a specialized approach; easily realizable with our system. First, the development of lateral branches was prohibited; thus, bifurcations could only develop in leaf nodes. Second, tissue growth was disabled and the proliferation of individual vessel trees constrained into a vertical region. Trees were then grown independent from each other, at different sizes. The trees were baked into textures, blurred, and combined. Larger vessel structures were considered to lie deeper in the tissue and thus experienced stronger blurring.

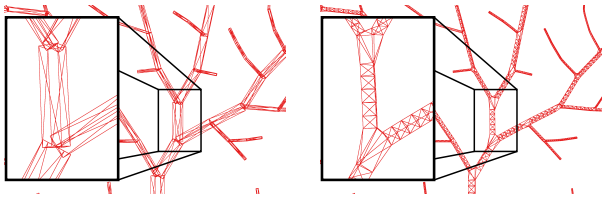


Figure 7: *Abstract preview (left) and high-res mesh (right).*

5. Conclusion

Our proposed synthesis framework is capable of generating vessel trees in milliseconds, which can be turned into triangle meshes as well as surface textures for various geometries. In combination with the control of large-scale vessel growth and real-time feedback of developing networks, vascular trees can be generated tailored to an artist's needs. The latter are, for instance, of tremendous value for VR-based surgical training systems. In order to obtain convincing results, a small set of parameters needs to be adjusted by a user, requiring some experience with the framework.

References

[BMB*16] BONALDI L., MENTI E., BALLERINI L., RUGGERI A., TRUCCO E.: Automatic Generation of Synthetic Retinal Fundus Images: Vascular Network. *Procedia Comp. Science 90* (2016), 54–60. 1

[HBC*10] HIJAZI Y., BECHMANN D., CAZIER D., KERN C., THERY S.: Fully-automatic Branching Reconstruction Algorithm: Application to Vascular Trees. In *Shape Modeling Int. Conf.* (2010), pp. 221–225. 4

[HJ10] HAMARNEH G., JASSI P.: VascuSynth: Simulating vascular trees for generating volumetric image data with ground-truth segmentation and tree analysis. *Comp. Med. Imag. & Graph. 34*, 8 (2010), 605–616. 1

[Mur26] MURRAY C. D.: The Physiological Principle of Minimum Work. I. The Vascular System and the Cost of Blood Volume. *Proc. of the Natl. Academy of Sciences of the USA 12*, 3 (1926), 207–214. 2

[RFL*05] RUNIONS A., FUHRER M., LANE B., FEDERL P., ROLLAND-LAGAN A., PRUSINKIEWICZ P.: Modeling & Visualization of Leaf Venation Patterns. *ACM ToG 24*, 3 (2005), 702–711. 1, 2, 3

[RLP07] RUNIONS A., LANE B., PRUSINKIEWICZ P.: Modeling Trees with a Space Colonization Algorithm. In *Proc. of the 3rd Eurographics Conf. on Natural Phenomena* (2007), Eurographics Assoc., pp. 63–70. 1