KNOWLEDGE ENGINEERING: PRINCIPLES AND TECHNIQUES
Proceedings of the International Conference on Knowledge Engineering,
Principles and Techniques, KEPT2007
Cluj-Napoca (Romania), June 6–8, 2007, pp. 122–129

SIMULATING MICROCAPILLARY NETWORKS USING RANDOM GRAPHS

SERBAN RARES POP [1], CIPRIAN-IONUT DUDUIALA [2], AND CAMELIA CHIRA [3]

Abstract. The blood contains plasma, a Newtonian fluid, and other suspended elements like red blood cells (erythrocytes), white blood cells (leukocytes) and platelets. These components, particularly red blood cells, strongly influence the blood properties and behaviour. A mathematical model is proposed to solve problems like the flow in a microcapillary network that includes the blood rheology and non-linear cell splitting at bifurcations. This model can be used to perform statistical studies on real microcapillary networks. The introduced model facilitates the characterization and prediction of events and behaviours unreachable with standard tools.

1. Introduction

Due to their different structures, microcapillaries and veins have different biological properties. In the case of a vein being destroyed, it can be replaced with an artificial one. However, this procedure can not be applied for a microcapillary. For this reason a representation of microcapillary networks that allows predictions and statistical studies is needed. A method based on random graphs for generating microcapillary networks is proposed. The resulting network is compared with a real microcapillary network. These results can be used to analyze microcapillary networks around the brain or around tumors. Furthermore, prediction of events and behaviours is enabled. For example, the proposed method can predict which network is vulnerable emphasizing that damaging the network or portions of it can be fatal.

2. Mathematical Model of a Microcapillary Network

Let us denote by $\mathcal{V}$, $\mathcal{I}$, $\mathcal{O}$, and $\mathcal{N}$ the sets of microvessels, inlet, outlet and interior nodes. In order to construct a microcapillary network, information about each vessel of uniform radius R_j and length L_j, with $j \in \mathcal{V}$ is needed. Also, the inlet and outlet pressures should be known since the flow is driven by the overall pressure drop (i.e. the difference between the inlet and outlet pressures). Using

Key words and phrases. microcapillary networks, random graph.

network parameters and the Network Solver Algorithm (presented in Section 2.4), the hematocrit in each link, pressures and flow rates distribution in the network can be determined. This actually refers to the way in which the blood moves into the network from the input nodes to the output nodes. An oriented weighted graph can be used to capture structural information about such a network.

2.1. **Hematocrit-dependent viscosity.** The blood viscosity reflects the property of the blood/vessel system for given flow conditions rather than solely the property of the blood itself. This viscosity is called the "apparent" or "effective" viscosity and depends strongly on the hematocrit and diameter of the vessel. This dependence is known as the Fåhræus-Lindqvist effect (see for instance [1]).The proposed model relies on the in vivo viscosity law, provided by Pries [1], which is derived from direct viscosity measurements:

$$\mu_j(H_j(x,t)) = \mu^p \cdot \mu_j^{rel},\ j \in \mathcal{V}, \tag{1}$$

where

$$\mu_j^{rel} = \left[1 + \left(\mu_j^* - 1\right) \frac{\left(1 - H_j(x,t)\right)^{C_j} - 1}{\left(1 - 0.45\right)^{C_j} - 1} \left(\frac{2\bar{R}_j}{2\bar{R}_j - 1.1}\right)^2\right] \left(\frac{2\bar{R}_j}{2\bar{R}_j - 1.1}\right)^2, \tag{2}$$

$$\mu_j^* = 6\exp\left(-0.17\bar{R}_j\right) + 3.2 - 2.44\exp\left(-0.06(2\bar{R}_j)^{0.645}\right), \tag{3}$$

and

$$C_j = \left(0.8 + \exp(-0.15\bar{R}_j)\right)\left(-1 + \frac{1}{1 + 10^{-11}(2\bar{R}_j)^{12}}\right) + \frac{1}{1 + 10^{-11}(2\bar{R}_j)^{12}}. \tag{4}$$

In the above equations, μ_j is the effective blood viscosity, μ^p is the plasma viscosity (4×10^{-3} Pa $\cdot$ s), and $C_j, \mu_j^*, \mu_j^{rel}$ are fitting coefficients which depend on the vessel radius and hematocrit - the proportion of blood occupied by red blood cells is referred to as the hematocrit.

2.2. **Poiseuille law.** The blood flow rate Q_j is given by the Poiseuille law (see [1]), under the assumption that the vessels are long and thin (lubrication theory). Hence, the blood flow rate is related at the pressure drop ΔP_j along the jth vessel by

$$Q_j = \sigma_j \Delta P_j, \tag{5}$$

where

$$\sigma_j = \frac{\pi R_j^4}{8 L_j \mu_j(H_j, R_j)}, \tag{6}$$

and L_j is the length of the jth vessel.

2.3. **Boundary conditions at bifurcations.** An important characteristic of the microcirculation is the non-uniform distribution of the blood components between the outgoing vessels at the splitting nodes. In particular, the higher the fraction of blood an outgoing vessel receives, the higher the hematocrit in that vessel. Hence, it is possible for one branch to receive a higher hematocrit than that of the parent vessel and the other to receive a lower (possibly zero) hematocrit. This phenomenon is called "plasma skimming" or "phase separation" [1].

In what follows, the parametric description of phase separation *in vivo*, proposed by Pries *et al* [1] is used. This model was obtained by fitting *in vivo* experimental data.

For any splitting node $k,\ k \in \mathcal{N}$, connecting the incoming vessel F_k and outgoing vessels α_k and β_k, with $F_k,\ \alpha_k,\ \beta_k \in \mathcal{V}$, mass conservation implies

$$Q_{F_k} = Q_{\alpha_k} + Q_{\beta_k}. \tag{7}$$

The fractional hematocrits of the outgoing vessels are

$$\frac{H_{\alpha_k}}{H_{F_k}} = \frac{1}{Q_k} \begin{cases} F_\alpha(Q_k), & X_{0_k} < Q_k < 1 + X_{0_k} \\ 0, & Q_k \leq X_{0_k} \\ 1, & Q_k \geq 1 + X_{0_k} \end{cases} \tag{8}$$

with $Q_k = Q_{\alpha_k}/Q_{F_k}$. The functions $F_\alpha(Q_k)$ and $F_\beta(Q_k)$ are given by

$$F_\alpha(Q_k) = \frac{1}{1 + \exp\left[-A_{\alpha_k} - B_k \log\left(G(Q_k)\right)\right]}, \tag{9}$$

$$F_\beta(Q_k) = \frac{1}{1 + \exp\left[-A_{\beta_k} - B_k \log\left(G(Q_k)\right)\right]}, \tag{10}$$

where

$$G(Q_k) = \frac{Q_k - X_{0_k}}{1 - Q_k - X_{0_k}}. \tag{11}$$

Likewise H_{β_k}/H_{F_k} satisfies (8) with Q_k replaced by $1 - Q_k$ and F_α by F_β.

The parameters A_{α_k}, A_{β_k}, B_k and X_{0_k} define the main aspects of the phase separation (i.e. asymmetry, sigmoidal shape and threshold):

$$A_{\alpha_k} = -\frac{6.96}{2\bar{R}_{F_k}} \ln\left(\frac{\bar{R}_{\alpha_k}}{\bar{R}_{\beta_k}}\right),\ A_{\beta_k} = -\frac{6.96}{2\bar{R}_{F_k}} \ln\left(\frac{\bar{R}_{\beta_k}}{\bar{R}_{\alpha_k}}\right), \tag{12}$$

$$B_k = 1 + 6.98\left(\frac{1 - H_{F_k}}{2\bar{R}_{F_k}}\right),\ X_{0_k} = \frac{0.2}{\bar{R}_{F_k}}. \tag{13}$$

The Pries-Secomb phase separation rule ensures conservation of hematocrit

$$H_{\alpha_k} Q_{\alpha_k} + H_{\beta_k} Q_{\beta_k} = H_{F_k} Q_{F_k}. \tag{14}$$

At conjunctions (two vessels α_k and β_k meet to form a single output vessel G_k) we need to apply only

$$Q_{\alpha_k} + Q_{\beta_k} = Q_{G_k},\ H_{\alpha_k} Q_{\alpha_k} + H_{\beta_k} Q_{\beta_k} = H_{G_k} Q_{G_k}. \tag{15}$$

2.4. **Network Solver Algorithm.** Proposed model facilitates computation of the hematocrit in each link as well as pressures and flow rates distribution in the network. The computational steps can be described by a procedure called Network Solver Algorithm (NSA). NSA is outlined below.

Network Solver Algorithm

Step 1: For given initial conditions (value of inlet/outlet pressure or flow rate), network geometry (length and radius for each vessel), and assuming an initial uniform hematocrit distribution for each vessel, $H_j,\ j \in \mathcal{V}$, compute viscosity for each link, pressures and flow rates distribution in the entire network.

Step 2: Apply the splitting rule to compute new values for the hematocrit in each link, H_j^{new} and repeat *Step 1* until the absolute error $\epsilon = \max(|H_j - H_j^{new}|),\ j \in \mathcal{V}$ is smaller than a given tolerance.

3. Random Graphs

A random graph [3] is a collection of vertices and edges randomly connecting pairs of nodes. Ussually, it is assumed that the presence or absence of an edge between two vertices is independent of the presence or absence of any other edge, so that each edge may be considered to be present with independent probability p. If the graph has N vertices each of them connected to an average of z edges, then it is trivial to show that $p = z/(N-1)$, which for large N is usually approximated by z/N. The number k of edges connected to any particular vertex is called the degree of that vertex. Ordinary random graphs are characterized by the Poisson distribution of vertex degree:

$$p_k = \binom{N}{k} p^k (1-p)^{N-k} \approx \frac{z^k e^{-z}}{k!}, \tag{16}$$

where p_k is the probability that a randomly chosen vertex on the graph has degree k.

Random graphs serve as models of real-world networks of different types, especially in epidemiology. For example, a disease passing through a community is strongly dependent on the contacts between those infected and those susceptible to disease. The network will have individuals represented by vertices and contacts by edges. Another widely studied network is the Internet. However, random graphs turn out not to be able to simulate with accuracy real-world phenomens.

3.1. **Generating functions.** Generating function [2] is a standard tool that can be used to create microcapillary networks. An example of such a function is $G_0(x)$

for the probability distribution of vertex degrees k. For a unipartite undirected graph of N vertices, with N large, $G_0(x)$ can be expressed as:

$$G_0(x) = \sum_{k=0}^{\infty} p_k x^k, \tag{17}$$

where p_k is the probability distribution of k. Usually the generating functions verify the condition $G_0(1) = 1$, since the distribution p_k is assumed correctly normalized.

Some properties of probability generating functions are listed below:

- probability p_k is given by the kth derivative of G_0: $p_k = \frac{1}{k!}\frac{d^k G_0}{dx^k}(0)$,
- the average degree z of a vertex is $z =< k >= \sum_k k p_k = G_0^{'}(1)$, which means that computing the mean of the probability distribution that the function generates, as well as higher moments of the distribution using the corresponding higher order derivatives of G_0.
- if a generating function is used for the distribution of a property k of an object, then the distribution of the total of k summed over m independent realizations of the object is generated by the m^{th} power of that generating function [2].

Depending on the distribution used, several other function can be defined: for Poisson-distributed graphs the probability $p = z/N$ of the existence of an edge between any two vertices is the same for all vertices and

$$G_0(x) = \sum_{k=0}^{\infty} \binom{N}{k} p^k (1-p)^(N-k) x^k = (1 - p + px)^N, \tag{18}$$

while for exponentially distributed graphs we have $p_k = (1 - e^{-1/K})e^{-k/K}$, with K constant, and

$$G_0(x) = (1 - e^{-1/K}) \sum_{k=0}^{\infty} e^{-k/K} x^k = \frac{1 - e^{-1/K}}{(1 - xe^{-1/K})}. \tag{19}$$

Furthermore, the expression of the function that generates the distribution of outgoing edges can be determined

$$G_1(x) = \frac{G_0^{'}(x)}{G_0^{'}(1)} = \frac{1}{z} G_0^{'}(x), \tag{20}$$

with z the average vertex degree. Using the third property of the generating function described above, the generating function for the probability distribution concerning the number of second neighbors for a vertex is obtained:

$$\sum_{k=0}^{\infty} p_k G_1^k(x) = G_0(G_1(x)). \tag{21}$$

The distribution of third-nearest neighbors is generated by $G_0(G_1(G_1(x)))$, and so on. Since $G_1(1) = 1$, the average number of second neighbors is $z_2 = G_0^{'}(G_1(x))_{x=1} = ... = G_0^{'}(1)G_1^{'}(1) = G_0^{''}(1)$. Taking into acount that we also have $z = G_0^{'}(1)$, one should not think that $z_k = G_0^{(k)}(1)$ because in general this is not true.

All the properties defined above can be used for component sizes, mean component size, phase transition, giant component, numbers of neighbors and average path length analysis [2].

4. Microcapillary Networks Generation Using Random Graphs

An algorithm useful for simulating a microcapillary network using random graphs is presentes. The obtained oriented weighted graph contains all the information needed to analyze the network. It is not possible to use directed random graphs to generate the microcapillary network for two main reasons as follows: (i) the blood direction through a vessel is given by the inlet and outlet pressures, and (ii) the blood direction may change in time (this means that even if at each moment the microcapillary network can be seen as a directed graphs, in reality the graph is in reality the graph is undirected).

When random graphs with various distributions of vertex degree are generated, a set of N random numbers k_i to represent the degrees of the N vertices in the graph is needed. Next, pairs of vertices are randomly chosen and joining edges are placed on the graph . This way a graph is generated (with equal probability) from the set of all possible graphs with the given set of vertex degrees. The only condition that has to be checked is that the sum $\sum_i k_i$ of the degrees is even, since each edge added to the graph has two ends. If the condition is not satisfied, a new set k_i is generated and the procedure is repeated until a suitable set is obtained. Integers representing vertex degrees with any desired probability distribution can be generated using the transformation method or a rejection or hybrid method [2].

Using the above algorithm, the edges of the graph are constructed. In case of microcapillary networks, things are a lot less complicated. As mentioned in Section 2, the network will have a set of in-nodes $\mathcal{I}$, a set of out-nodes $\mathcal{O}$ and a set of interior nodes $\mathcal{N}$. The in-nodes and out-nodes will have degree 1, while the interior nodes, due to the biological properties of microcapillaries, will have degree 3. In other words the set k_i may contain only elements with values 1 or 3, thus the condition that should be satisfied is $|\mathcal{I}| + |\mathcal{O}| \equiv |\mathcal{N}|(mod\ 2)$. This condition means that the number of in-nodes and out-nodes must have the same parity as the number of interior nodes. The generating function for the probability distribution becomes

$$G_0(x) = \frac{|\mathcal{I}| + |\mathcal{O}|}{TOT}x + \frac{|\mathcal{N}|}{TOT}x^3, \tag{22}$$

where $TOT = |\mathcal{I}| + |\mathcal{O}| + |\mathcal{N}|$.

The key of the paper is the fact that the obtained graph contains several connected componets. In reality, we need a graph which is connected, which means that the giant component [2] contains all the nodes of the graph. This is possible only if $|\mathcal{I}| + |\mathcal{O}| \leq |\mathcal{N}|$. The condition is implied by the network structure: an in-node or out-node can be linked only with interior nodes. Hence, it is easier to construct the graph by determining first the edges connecting only interior nodes and only at the end the edges involving in-nodes and out-nodes.

There are two ways of constructing a connected graph, using the random graphs theory: (i) construct the graph as described above and then add as few edges as possible between the different connected components or (ii) construct $\mathcal{N}/2$ edges that form $\mathcal{N}/2$ different connected components (i.e. each interior node is used exactly once), add at each step only edges that connect nodes from different components to reduce their number and when only one component is obtained, pairs of vertices are randomly chosen and joining edges are also placed on the graph, taking into acount the last observation from the previous paragraph.

First method implies modifying the number of interior nodes. Adding an edge between two different connected components acctually means adding two new nodes in the graph and replacing one edges from each compenent with two new edges. Second methods keeps constant the number of nodes and generates - with equal probability - one of the connected graphs that respect the above properties.

On the other hand, a network containing several connected components can also be useful. In such a case, studies about how the flow changes in each component when the components are joined together by adding nodes and edges can be made.

The next step is to assign to each vessel a random radius R_j and a random length L_j, whose values are taken between a minimum and a maximum value obtained from experiments. Observations of microcapillary networks show that some parts of the network may contain larger vessels than others, thus it might be useful to use different minimum and maximum values for different components of the network.

Finally, the flow problem of the network should be solved. Since only one edge leaves the in-nodes only one edge enters the out-nodes enters the out-nodes, the blood direction for these vessels can be exactly determined. For the rest the Network Solver Algorithm (proposed in Section 2) is applied.

5. Conclusions and Future Work

Some properties of the microcapillary networks are presented. A mathematical model for a microcapillary network and an algorithm to solve the flow in such a network are proposed. Random graphs and generating functions represent useful tools for analyzing the networks simulating real-world phenomena. A way in which microcapillary networks can be simulated using undirected graphs (which are then transformed into weighted directed graphs by solving the flow in the network) is indicated.

Future work focuses on analyzing the differences between a simulated microcapillary network and a real one. Simulations can indicate the potential of the proposed model for microcapillary networks analysis. Genetic algorithms can be used to indicate the damaging rate that will not indluence the overall functionality of the network. Furthermore, the vital vessels of the network (i. e. destroying them could lead to a hemorrhage) can be identified using evolutionary techniques.

References

[1] A.R. Pries, T.W. Secomb, P. Gaehtgens - Biophysical aspects of blood flow in the microvasculature. *Cardiovascular Research* **32** (1996), 654-667.

[2] M. E. J. Newman, S. H. Strogatz, and D. J. Watts - Random graphs with arbitrary degree distributions and their applications, Physical Review E, Volume 64, 026118 (2001)

[3] P. Erdos and A. Renyi, On random graphs I, Publ. Math. (Debrecen) 9 (1959), 290-297.

(1) School of Mathematical Sciences, University of Nottingham, UK
E-mail address: popserban@yahoo.com

(2) Centre for Mathematical Medicine and Biology, School of Mathematical Sciences, University of Nottingham, UK
E-mail address: cipriduduiala@yahoo.com

(3) Mathematics and Computer Science Faculty, "Babes-Bolyai" University, Cluj-Napoca, Romania
E-mail address: cchira@cs.ubbcluj.ro