Manuscript submitted to **Biophysical***Journal*

Article

Timelapsing perfusion: Proof of concept of a novel method to study drug delivery in whole organs.

A. K. Diem, K. Valen-Sendstad

ABSTRACT Perfusion is one of the most important processes maintaining organ health. From a computational perspective, however, perfusion is amongst the least studied physiological processes of the heart. The recent development of novel nanoparticle-based targeted cardiac therapy calls for novel simulation methods that can provide insights into the distribution patterns of therapeutic agents within the heart tissue. Additionally, resolving the distribution patterns of perfusion is crucial for gaining a full understanding of the long-term impacts of cardiovascular diseases that can lead to adverse remodelling, such as myocardial ischemia and heart failure. In this study we have developed and used a novel particle tracking-based method to simulate the perfusion-mediated distribution of nanoparticles or other solutes. To model blood flow through perfused tissue we follow the approach of others and treat the tissue as a porous medium in a continuum model. Classically, solutes are modelled using reaction-advection-diffusion kinetics. However, due to the discrepancy of scales between advection and diffusion in blood vessels, this method becomes practically numerically unstable. Instead, we track a bolus of solutes or nanoparticles using particle tracking based purely on advection in arteries. In capillaries we employ diffusion kinetics, using an effective diffusion coefficient to mimic capillary blood flow. We first demonstrate the numerical validity and computational efficiency of this method on a 2D benchmark problem. Finally, we demonstrate how the method is used to visualise perfusion patterns of a healthy and ischemic human left ventricle geometry. The efficiency of the method allows for nanoparticle tracking over multiple cardiac cycles using a conventional laptop, providing a framework for the simulation of experimentally relevant time frames to advance pre-clinical research.

STATEMENT OF SIGNIFICANCE Computational cardiac modelling has gained more and more relevance over the past twenty years. Despite being the most fundamental process in maintaining organ health and the major pathway for drug delivery, perfusion has, from a computational point of view, received very little attention. Here we address this issue by presenting a novel and computationally efficient method to model perfusion patterns and distributions of tracers or drugs. As opposed to advection-diffusion kinetics, our method is numerically stable for advection-dominated problems like blood flow. Its efficiency allows for the simulation of multiple cardiac cycles of a left ventricle using only a conventional laptop, enabling simulations over experimentally relevant time frames.

INTRODUCTION

The heart has famously been described as the "most highly integrated example of a virtual organ" (1), and the field of whole-heart computational modelling has seen growing demand over the past twenty years. A substantial part of this research has been driven by research into normal cardiac function and arrhythmias, and thus the focus has been on the development of electromechanical simulations of the left ventricle (1). Perfusion, that is the blood supply to the tissue via the circulatory system, on the other hand, has been studied much less extensively from a computational point of view. However, cardiac perfusion is one of the most fundamental processes maintaining organ health and the major route for drug delivery. The recent development of novel, targeted nanoparticle-based cardiac therapy (2–4), drives the need for computationally efficient and accurate models to accelerate the development of this new technology. In addition, resolving the full impact of cardiac diseases is highly dependent on precise knowledge of the mechanism of perfusion and the effects of pathophysiological conditions such as heart failure or myocardial ischemia.

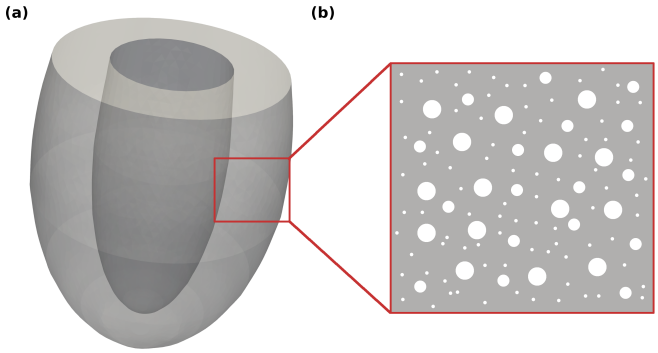

Figure 1: Schematic illustration of the porous media modelling approach for left ventricle tissue. Here, myocardial tissue is treated as a sponge-like material, such that continuum models over the domain can be applied. Pores in the material (white) represent the perfusing vasculature that are embedded in the solid tissue (grey). To represent the varying diameters of blood vessels embedded in the myocardial tissue, we use a multi-compartment Darcy model as introduced by Michler et al. (5). In this approach, three porous materials (visualised by white circles of three different radii) with separate parameters for porosity and permeability are superimposed onto the myocardium to represent large arterioles, small arterioles, and capillaries.

Heart tissue is perfused by thousands of very small arteries and capillaries branching off the coronary arteries. State-of-the-art imaging techniques generally have a resolution limit of 0.5 mm such that arteries with diameters less than 2 mm cannot be resolved (6). Obtaining truthful representations of the exact geometries of the perfusion circulation using state-of-the-art imaging techniques is therefore virtually impossible. Instead, we regard heart tissue as a porous material and use continuum models to obtain approximate solutions of perfusion pressure and flow velocities, following the work of Michler et. al. (5, 7, 8). Here, perfused heart tissue is modelled as a superposition of N porous materials, enabling the representation of materials with inhomogeneous porosities. In this particular context, we choose $N = 3$ to represent perfusion at the artery, arteriole, and capillary level (Figure 1). However, given that blood vessel diameters gradually reduce with the number of bifurcations, any other number greater than three could be chosen to build a perfusion model. Applying advection-diffusion kinetics to the velocity solutions allows for the study of the distribution and effects of tracers or therapeutic compounds. In our application, the numerical solution of the nonlinear advection-diffusion equation requires the use of fine meshes and small time steps. In mechanics modelling this problem is overcome by simulating one heart beat and extrapolating these results to multiple heart beats. However, when one is interested in modelling the distribution of nanoparticle-based therapeutic compounds and their effects on the mechancis of the heart, such an extrapolation from one heartbeat to a series of multiple heartbeats is no longer appropriate. For example, a therapeutic compound could improve flow towards an ischemic borderzone of the heart, altering perfusion and/or wall motion in this particular area. Experimental effects are expected to take place over the course of approximately 30 min (2). Assuming an average heart beat duration of 0.85 s results in approximately 2100 heart beats. Taking eight CPU hours as an average compute time for one heart beat then results in nearly 17,000 CPU hours compute time, or around two years. Thus the successful prediction of efficacy and efficiency of therapeutic compounds demands the development of efficient numerical methods that exploit the underlying physical properties of the specific problem.

We address this challenge by presenting a novel numerical method of tracking nanoparticle delivery via perfusion to the myocardium of the left ventricle. The major advantage of this method is the ability to account for the discrepancy of scales between the size of a therapeutic nanoparticle, and blood flow in capillaries, arterioles and arteries, as well as the size of a human left ventricle. For simplicity we will for the remainder of the paper only refer to nanoparticles as the distributed substance,

however, the method is equally applicable to any other kind of tracer substance. The method is based on the observation that the diffusion coefficient of a nanoparticle is many orders of magnitude smaller than blood flow velocities We assume thus that nanoparticle distribution within the arteries and arterioles is solely goverened by advection, which follows the directionality of muscle fibers represented by the permeabilility tensor. Furthermore, owing to the length and distributions of capillaries, we follow the observations by (7, 9) and assume isotropy of the capillary permeability tensor.

Our method is based on Lagrangian particle tracking that provides a significant improvement in both computational cost and accuracy of the results, compared to conventionally used advection-diffusion kinetics. It exploits the assumption that blood flow velocities in the large and small arterioles result in large Peclet numbers, which renders the use of advection-diffusion kinetics inadequate. Instead, we assume that the location of nanoparticles is solely influenced by advection, taking into account the directionality of cardiac muscle fibers. Perfusion in capillaries on the other hand is assumed to be isotropic, and thus can be represented by a large diffusion coefficient based on capillary flow velocities. To obtain perfusion velocities in the large arterioles, small arterioles, and capillaries, perfusion is represented by a three-compartment porous media continuum model based on Darcy's law as previously done by other groups (5, 10), while nanoparticle delivery is approximated by splitting advection and diffusion kinetics across compartments. We demonstrate the efficiency of our method by simulating perfusion to the left ventricle, driven by inflation and contraction, over multiple cardiac cycles using a conventional laptop, instead of having to rely on the use of a high-performance computing cluster. Additionally, we show how the method can be used to study effects of cardiac conditions by demonstrating the application to ischemia.

METHODS

We model the myocardium of the left ventricle (domain Ω) as a three-compartment reduced Darcy continuum model (5), where a compartment represents large arteriole, small arteriole and capillary beds, respectively. Denoting different compartments using index i,

$$-\nabla \cdot (\mathbf{K}_i \cdot \nabla p_i) + \sum_{k=1}^{3} \beta_{i,k}(p_i - p_k) = s_i \qquad \text{in } \Omega \tag{1}$$

with $\mathbf{K_i}$ permeability of the porous medium, p_i pressure, and $\beta_{i,k}$ exchange coefficient between compartments i and k, and s source/sink terms. We use Neumann boundary conditions on boundaries Γ_N to model the inflow of blood from the coronary arteries via volumetric inflow Ψ_i

$$(-\mathbf{K}_i \cdot \nabla p_i) \cdot \mathbf{n} = \Psi_i \qquad \text{on } \Gamma_N. \tag{2}$$

Nanoparticle distribution follows the same advection-diffusion kinetics as the distribution of, for example, an MRI contrast agent (11). The concentration c_i in each compartment i is thus

$$\frac{\partial c_i}{\partial t} = \nabla \cdot (D_i \nabla c_i) - \mathbf{w}_i \cdot \nabla c_i + \sum_{k=1}^{3} \gamma_{i,k} c_i + s_i \qquad \text{in } \Omega. \tag{3}$$

with D_i diffusion coefficient, $\mathbf{w_i}$ advection velocity, $\gamma_{i,k}$ exchange coefficients between compartments i and k, and s_i source/sink terms.

Separating method for nanoparticle deposition

Deposition of nanoparticles into myocardial tissue is assumed to strongly rely on margination (12), which allows for endocytosis on nanoparticles into myocardial cells. As margination occurs at the blood vessel wall, this process relies on small fluid velocities such that nanoparticles are able to move across fluid streamlines. The reliance of endocytosis after margination on low shear rates that allow for sufficient adhesion time of nanoparticles at the blood vessel wall has been shown in a number of studies (12–14). Thus, it is most likely that nanoparticle deposition occurs primarily within the capillary compartment. The nonlinear advection-diffusion equation is time consuming to solve and is well known to be associated with numerical issues at high Peclet numbers, where the effects of advection dominate over the effects of diffusion. These numerical issues refer to stability issues leading to a loss of conservation of mass. The numerical Peclet number is

$$\mathrm{Pe} = \frac{||\mathbf{w}||h}{2D} \tag{4}$$

for velocity norm $||\mathbf{w}||$ and mesh element size h. In our case, advection velocities in the large arteriole compartments are expected to reach an order of magnitude of $0.1\,\mathrm{mm\,s^{-1}}$, while the diffusion coefficient of a therapeutic nanoparticle can be estimated from the Stokes-Einstein equation $D = 6kT/\pi\mu r$, using Boltzmann constant k, body temperature T in units of K,

blood viscosity μ, and nanoparticle radius r, resulting in an order of magnitude of $1 \times 10^{-13}\,\mathrm{m^2\,s^{-1}}$. Thus the Peclet number for nanoparticle transport in blood vessels approaches infinity for fine meshes.

To overcome this problem we introduce the following method: At large Peclet numbers the effects of diffusion can be neglected such that we can approximate the position $\mathbf{x}$ of any particle advected by the fluid evolves according to a simple first order transient model

$$\frac{\partial \mathbf{x}}{\partial t} = \mathbf{w}. \tag{5}$$

To efficiently track nanoparticles through the myocardium we model a *bolus* as a collection of nanoparticles using Lagrangian particle tracking in the artery and arteriole bed (compartments 1 and 2). Furthermore, while blood flow in arterioles is generally considered to follow muscle fiber directions, we assume that capillary flow within the myocardium is isotropically distributed. This assumption is based on the imaging data by (7, 9) who found that capillaries were below the resolution of the data. Thus, a single capillary branch is sufficiently small that its spatial orientation cannot have any significant influence on the results. Therefore, we can approximate the distribution of nanoparticles in the capillary compartment by a sufficiently large diffusion coefficient that represents the velocity of capillary flow. The combined nanoparticle tracking model equation thus reads

$$\frac{\partial \mathbf{x}_i}{\partial t} = \mathbf{w}_i \quad i = 1, 2 \qquad \text{in } \Omega \tag{6}$$

$$\frac{\partial c_3}{\partial t} = \nabla \cdot (D_3 \nabla c_3) + \gamma_{2,3} c_2 + s_3 \qquad \text{in } \Omega. \tag{7}$$

We use the finite element method to solve the model equations. The variational form of the multi-compartment Darcy model reads: Find $p_i \in H^1(\Omega)$ such that $\forall q_i \in H^1(\Omega)$

$$(\mathbf{K}_i \cdot \nabla p_i, \nabla q_i)_\Omega + (\Psi_i, q_i)_{\Gamma_N} + \left(\sum_{k=1}^{3} \beta_{i,k}(p_i - p_k), q_i \right)_\Omega - (s_i, q_i)_\Omega = 0 \tag{8}$$

Because blood flows from large arterioles to small arterioles to capillaries, we set the source terms $\Psi_2 = \Psi_3 = 0$, while Ψ_1 governs inflow into the myocardium. Analogously we set sink terms $s_1 = s_2 = 0$ and $s_3 = -0.1\,\mathrm{Pa^{-1}\,s^{-1}}(p_3 - 3\,\mathrm{kPa})$ following (5). Additionally, exchange coefficients β are restricted to only allow exchange from compartment 1 to compartment 2, and from compartment 2 to compartment 3, such that $\beta_{1,3} = \beta_{2,1} = \beta_{3,1} = 0$. Velocities $\mathbf{w}_i$ are then easily calculated from p_i using Darcy's law with porosities Φ_i.

$$\mathbf{w}_i = -\Phi_i \mathbf{K}_i \nabla p_i. \tag{9}$$

and serve as the input for the particle tracking method. In the large and small arteriole compartments, each bolus of nanoparticles is advected according to the first order advection model (5). In the capillary compartment, concentrations follow diffusion kinetics, whose variational form reads: Find $c \in H^1(\Omega)$ such that $\forall v \in H^1(\Omega)$

$$\left(\frac{\partial c}{\partial t}, v \right)_\Omega + (D \nabla c, \nabla v)_\Omega = r, \tag{10}$$

where r represents the rate at which nanoparticles reach the capillary compartment from the small arteriole compartment. The equations are implemented in Python 3.5.2 using the finite element framework FEniCS (15).

RESULTS

The accuracy and efficiency of the splitting method was first tested using a benchmark problem on a unit square geometry. Following the successful benchmark, we present the application of the splitting method on the simulation of nanoparticle distributions on a human left ventricle geometry.

Benchmark problem

To demonstrate the accuracy and efficiency of the splitting method numerical tests were performed on a unit square geometry using a sinusoidal velocity profile on two compartments (Figure 2). Using a small and simple geometry for benchmark testing is better suited for a numerical comparison for three main reasons: (a) It allows easy and fast visual validation of the effectiveness of the particle tracking method, (b) the problem clearly demonstrates the inaccuracy of advection-diffusion kinetics compared to our particle tracking method, and (c) the problem solves fast enough also for the advection-diffusion model, such that it allows for easy evaluation of the efficiency of our particle tracking model compared to the advection-diffusion model. The velocity

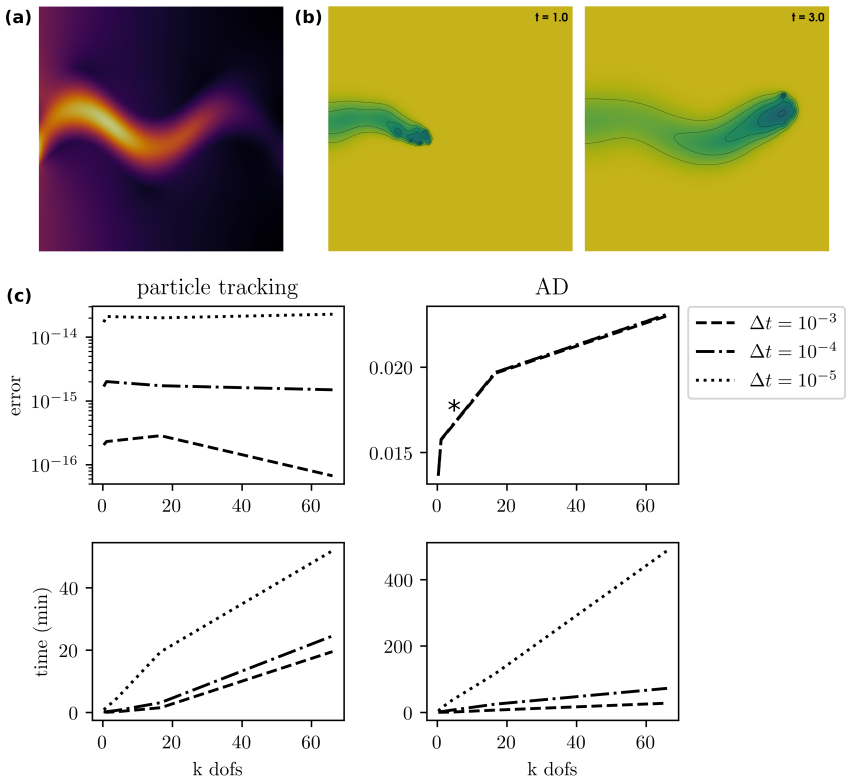

Figure 2: Numerical tests of the particle tracking method on a 2D unit square mesh using $N = 2$ compartments. (a) Magnitude of the velocity profile $\mathbf{w_i}$ used in the simulations. (b) Simulation results using the particle tracking method on the unit square mesh at $t = 1.0$ s and $t = 3.0$ s. (c) Relative error and simulation times for numerical tests (test simulation time 0.2 s) comparing the particle tracking method to the classical advection-diffusion approach over an increasing number of dof at $\Delta t = 1 \times 10^{-3}$, 1×10^{-4}, 1×10^{-5} s. Depending on the mesh resolution the Peclet number varies between 2.3 to 34.5. Simulation time increases linearly with the number of dof for both the particle tracking and advection-diffusion method. The longest simulation time for the particle tracking method was ≈ 28 min, while for the advection-diffusion method the longest simulation time was ≈ 8.1 h. Errors are large for the advection-diffusion method and do not vary with the time step size. One simulation (4k dofs, $*$) diverged and thus data from this simulation has not been included in the figure. Errors for the particle tracking method range between 1×10^{-14} to 1×10^{-16}, but increase for smaller Δt.

Table 1: Simulation parameters for the 2D unit square test simulations.

Parameter	Unit	Value
$\mathbf{K}_1, \mathbf{K}_2$	$\text{mm}^2\,\text{Pa}^{-1}\,\text{s}^{-1}$	$\exp(-(y - 0.5 - 0.1 * \sin(10x))/0.1)^2)$
Φ_1	1	0.21
Φ_2	1	0.23
$\beta_{1,2}$	$\text{Pa}^{-1}\,\text{s}^{-1}$	2×10^{-3}
$\gamma_{1,2}$	1	0.1
D_2	mm^2	1×10^{-3}
Δt	s	1×10^{-3}
Δt_{AD}	s	1×10^{-4}

profile was obtained by solving Darcy's law for a sinusoidal permeability tensor $\mathbf{K}_i$ (Figure 2a, Table 1). Figure 2b shows the results of a simulation over 3.0 s using the particle tracking method. Using the parameters listed in Table 1, a numerical analysis of the two methods was perfomed on regular meshes ($16 \times 16, 32 \times 32, 64 \times 64, 128 \times 128, 256 \times 256$) over 0.2 s, using $\Delta t = 1 \times 10^{-3}$, 1×10^{-4}, 1×10^{-5} s (Figure 2c). The test simulations do not contain any source or sink terms and are initialised by setting an initial concentration (advection-diffusion method) or initial number of particles (particle tracking method). To

compare the two methods we thus measured total mass in the domain

$$m = \int c \cdot dx + M * c_0, \tag{11}$$

where the second term corresponds to the total mass released by M particles containing an initial concentration c_0 each. The relative error can then be calculated as the change in mass

$$\epsilon = |m_i - m_{i+1}| \tag{12}$$

per time step i.

$$m = \int c dx \tag{13}$$

Errors and simulation times over dof are shown in Figure 2c. In the case of the advection-diffusion equations the errors are high and do not improve with Δt due to the increasing Peclet number. It is well known that the advection-diffusion equations show numerical instabilities for Pe $>$ 1. One simulation (4k dof, $*$) diverged, that is, total mass diverged to infinity and subsequently NaN (not a number), and thus the data has been omitted from the figure. Errors for the particle tracking method are close to machine precision in all cases, ranging between 1×10^{-14} to 1×10^{-16}, but increase for smaller Δt. Because the errors are close to machine precision and from a numerical point of view negligible, they are influenced by rounding errors. As smaller Δt mean that the diffusion problem is solved for more time steps, rounding errors increase with decreasing Δt. Simulation time increases linearly with the number of dof for both the particle tracking and advection-diffusion method. The longest simulation time for the particle tracking method was $\approx$ 28 min, while for the advection-diffusion method the longest simulation time was $\approx$ 8.1 h. The results demonstrate that the particle tracking method provides significant improvement both in terms of simulation time and errors over the advection-diffusion equations.

Nanoparticle distributions on a human left ventricle

Table 2: Simulation parameters for the left ventricle simulations.

Parameter	Unit	Value
K_1	$\mathrm{mm^2\,kPa^{-1}\,s^{-1}}$	1
K_2	$\mathrm{mm^2\,kPa^{-1}\,s^{-1}}$	10
K_3	$\mathrm{mm^2\,kPa^{-1}\,s^{-1}}$	20
Φ_1	1	0.021
Φ_2	1	0.029
Φ_3	1	0.061
$\beta_{1,2}$	$\mathrm{Pa^{-1}\,s^{-1}}$	0.02
$\beta_{2,3}$	$\mathrm{Pa^{-1}\,s^{-1}}$	0.05
$\gamma_{1,2}, \gamma_{2,3}$	1	0.1
D_3	$\mathrm{mm^2}$	1.0
Δt	s	1×10^{-4}

Following the successful tests of the particle tracking method on a 2D unit square mesh, we applied the same method to modelling perfusion of an idealised left ventricle geometry, with and without ischemia. The simulation parameters are listed in Table 2, where parameters relating to the pressure equations are taken from (5). These parameters have been estimated in a series of imaging experiments on porcine hearts (7, 9). In the literature, the average capillary flow velocity has been reported as $\approx$ 1 mm s^{-1} (16) and thus D_3 has been chosen as 1 mm^2 accordingly, while γ has been chosen to provide nanoparticle distribution within a reasonable time frame ($\sim$ one cardiac cycle).

Figure 3 shows pressure in units of kPa in the three compartments (a, left: large arterioles, middle: small arterioles, right: capillaries) and Darcy flow magnitude (velocity scaled by tissue porosity) in units of mm s^{-1} ((b, left: large arterioles, middle: small arterioles, right: capillaries)). Our pressure values ranges (22 mmHg to 45 mmHg) are similar to the simulation results presented in (5), whilst our simulation reaches a faster peak velocity magnitude of 0.5 mm s^{-1} compared to the results by Michler et al. (5). The large arterioles compartment (left) shows the largest gradients in both pressure and velocity. The capillary compartment shows very small pressure gradients of 7 mmHg, with a corresponding peak velocity of 0.5 mm s^{-1}. These results indicate that it is appropriate to disregard velocities within the capillary compartment and instead use a diffusion coefficient that

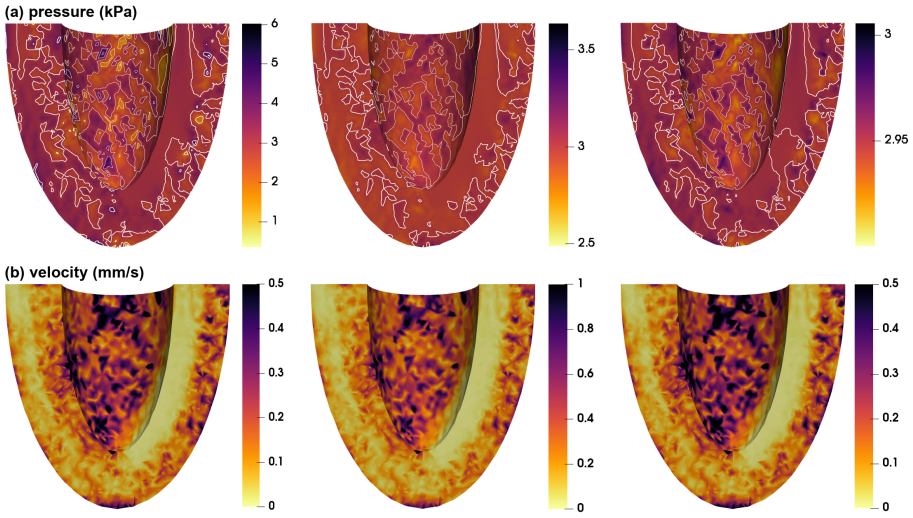

Figure 3: Perfusion pressure with isocontours in units of kPa (a) and velocity magnitude in units of mm s^{-1} (b) in the three compartments (from left to right: large arterioles, small arterioles, capillaries). Pressure values range between 22 mmHg to 45 mmHg, which agree with the simulation results presented in (5), whilst our simulation reaches a faster peak velocity magnitude of 0.5 mm s^{-1} compared to the results by (5). The large arterioles compartment (left) shows the largest gradients in both pressure and velocity. The capillary compartment shows very small pressure gradients of 7 mmHg, with a corresponding peak velocity of 0.5 mm s^{-1}.

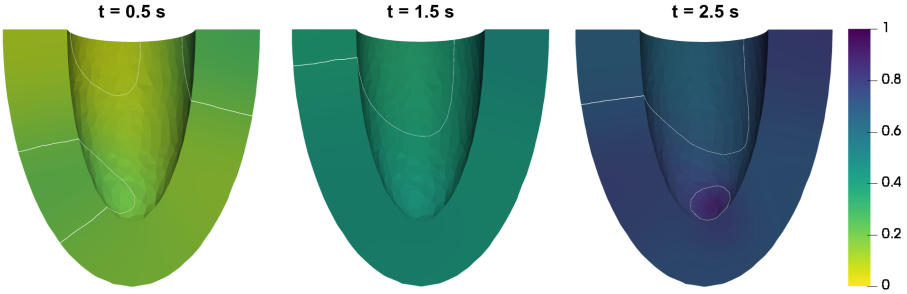

Figure 4: Nanoparticle distribution with isocontours in the left ventricle geometry over three cardiac cycles at $t = 0.5, 1.5, 2.5$ s (left to right). Values are shown as a fraction of the maximum concentration.

matches capillary flow conditions. Figure 4 shows the distribution of nanoparticles in the left ventricle over three cardiac cycles at $t = 0.5, 1.5, 2.5$ s (left to right). Complete perfusion is reached shortly after the second cardiac cycle.

Following from the results of left ventricle perfusion, we next tested our method for its ability to account for myocardial ischemia. Ischemia was introduced via the permeability tensor $\boldsymbol{K}$, which was set to zero in the ischemic zone (see black contour line in Figure 5). The diffusion coefficient that represents capillary perfusion was also set to zero in the ischemic zone, but otherwise was not altered. Perfusion velocity subject to ischemia is smaller by one order of magnitude in both the large and small arteriole compartment, and the region with close to zero perfusion velocity is much larger than the ischemic zone. Nanoparticle distribution at $t = 0.5, 1.5, 2.5$ s in the ischemic case is schown in Figure 6. The ischemic zone contains no nanoparticles, while the rest of the left ventricle receives a similar distribution of nanoparticles to the non-ischemic case.

DISCUSSION

Computational methods are more and more prevalent in medical research and provide a crucial tool to aid the advancement of clinical experiments and personalised medicine. However, large computational costs requiring the use of long hours in high performance computing centres remain a major limitation for the direct application of computational methods in biomedical development. In this study we have addressed this issue by presenting an efficient method for simulating the distribution of

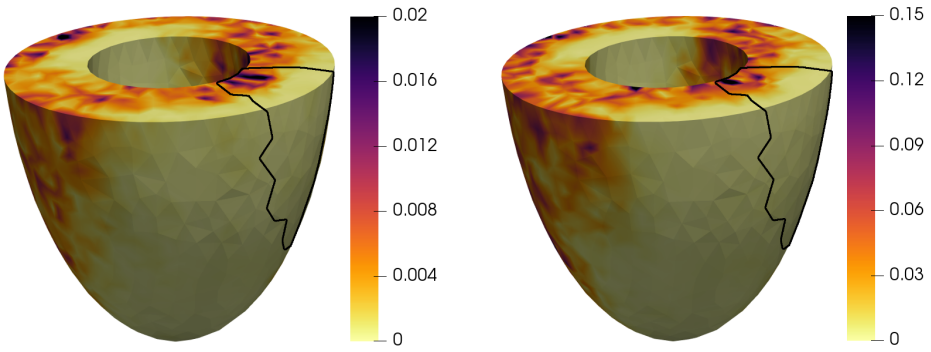

Figure 5: Perfusion velocity in a left ventricle subject (large and small arterioles) to ischemia. The ischemic zone is marked by a black contour. Perfusion velocity reaches a peak of 0.15 mm s^{-1} in the small arteriole compartment, and is close to zero in a large area around the ischemic zone.

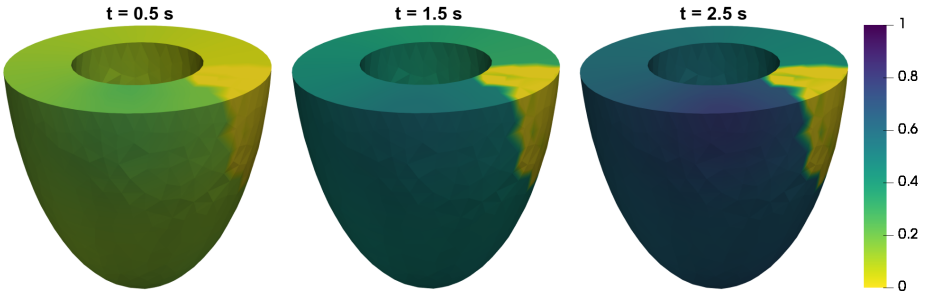

Figure 6: Nanoparticle distribution subject to ischemia over three cardiac cycles at $t = 0.5, 1.5, 2.5$ s.

nanoparticles or any other perfusion tracer. Using benchmark simulations on a unit square geometry we demonstrate that this method is also valid at large Peclet numbers, where conventional advection-diffusion kinetics are known to be subject to numerical errors. Our method exploits the discrepancy of scales between blood vessels within the myocardium, and allows for the simulation of a full left ventricle geometry over multiple cardiac cycles using only a conventional laptop within minutes.

To test our method we have compared simulations on a unit square geometry between the particle tracking method and the conventional advection-diffusion equations. One of the major limitations of the advection-diffusion equations is their reliance on a small Peclet number (< 1) in order to be numerically stable, which was confirmed by the divergence of the advection-diffusion simulation at 4k dofs on the unit square. For larger Peclet numbers close to the limit it is generally possible to reduce numerical effects using various upwind schemes. However, in this application the Peclet numbers far exceed the limit of such upwind schemes. Due to these effects, mass is not conserved, rendering the advection-diffusion equations effectively inappropriate for tracking tracers through the blood stream at large Peclet numbers. The particle tracking method on the other does not suffer from such numerical instabilities. To demonstrate this, we use mass conservation as our error criterion, after visual and manual validation of the particle tracking method. Here, we have used a first order transient model to approximate the position of a particle. We have demonstrated that the errors in total mass using the particle tracking method are in the order of 1×10^{-14} to 1×10^{-16} (close to machine precision). Additionally, the particle tracking method offers a significant simulation time speed up compared to solving the advection-diffusion equations. In the largest test case (256×256 mesh, $\Delta t = 1 \times 10^{-5}$ s) the particle tracking method solves in less than half an hour, while the advection-diffusion equations require over eight hours computing time. Simulation time for the particle tracking method mostly depends on the number of particles introduced into the geometry.

We demonstrate the application of our method to cardiovascular research by simulation nanoparticle distribution in an idealised human left ventricle geometry. To achieve this, we have implemented a Darcy's law type model of myocardial perfusion based on the work by Michler et al. (5, 7–9). It should be noted that the parameters for this model obtained by Hyde et al. (7, 9) are based on imaging experiments using porcine hearts. Therefore, a detailed parameter sensitivity analysis should be performed in order to understand the significance of each parameter and its implication for human heart models. It is, however, appropriate to perform a qualitative analysis using these parameters. The results between our implementation of the perfusion model and that by Michler et al. are very similar, but some differences in peak pressure and velocity magnitude are observed. One major

difference in our implementation is that we do not allow connections between non-neighbouring compartments in the perfusion model, as blood flow is strictly hierarchical, that is, blood does not flow back from capillaries to larger arterioles. Michler et al. justified their choice by observing that the results better matched their expectations. Due to a lack of perfusion validation data, we have decided to instead observe physiological principles and implement a strictly hierarchical perfusion model.

To demonstrate the use of our method to studying cardiac conditions, we additionally applied our particle tracking model to a left ventricle subject to ischemia. Ischemia was modelled by modifying both the permeability tensor and diffusion coefficient in the ischemic region. In the perfusion model, this change leads to an overall reduction in blood flow velocity by an order of magnitude. Additionally, velocity is close to zero in a much larger region surrounding the ischemic zone. The ischemic region does not receive any nanoparticles, however, nanoparticle distribution outside of the ischemic region appears mostly unaffected by the presence of the ischemic zone and the reduction in perfusion velocity. This is most likely explained by the unaltered diffusion coefficient that drives perfusion in the capillary compartment. Following the velocity results in the large and small arteriole compartments, it should be expected that the capillary compartment experiences a similar reduction in perfusion velocity, which should have been reflected by the diffusion coefficient. A test simulation (result not shown) indicated that this is indeed the case. We included the results of the simulation with the unaltered diffusion coefficient instead as it provides a much better visualisation of the effect of the ischemic zone on the simulation .

The rapid advancement of pre-clinical research from small to large animals crucially depends on improving the interaction of in vivo and in silico experiments. Building on the work by Michler et al. (5, 8), with a particular motivation by the experimental work on non-invasive drug delivery by Miragoli et al. (2), we have hereby provided a next step in this direction. Our particle-based method of modelling tracer distributions through the left ventricle eliminates the requirement to compute timesteps in the order of 1×10^{-5} s, as well as our model's dependence on the Peclet number. With these improvements we have been able to demonstrate that it is possible to simulate a number of cardiac cycles on a conventional laptop, instead of having to rely on high-performance computing clusters. These improvements open the doors to simulating experimentally relevant time frames of 30 min (2) or more, which contain 30–40 heart beats in a human, but $\sim 1 \times 10^5$ heart beats in a mouse.

AUTHOR CONTRIBUTIONS

A.K.D and K.V.S. designed the study. A.K.D carried out all simulations, A.K.D and K.V.S. analyzed the data. A.K.D. and K.V.S wrote the article.

ACKNOWLEDGMENTS

The authors acknowledge the European Community for its financial support in the framework of the Horizon 2020 project CUPIDO (www.cupidoproject.eu) H2020-NMBP2016 720834. This work was additionally supported by the Centre for Cardiological Innovation, and SIMMIS (262827), funded by the Research Council of Norway. We further thank Henrik Finsberg (Simula Research Laboratory) for providing the left ventricle mesh.

REFERENCES

1. Trayanova, N. A., 2011. Whole-heart modeling : Applications to cardiac electrophysiology and electromechanics. *Circulation Research* 108:113–128.

2. Miragoli, M., P. Ceriotti, M. Iafisco, M. Vacchiano, N. Salvarani, A. Alogna, P. Carullo, G. B. Ramirez-Rodríguez, T. Patrício, L. Degli Esposti, F. Rossi, F. Ravanetti, S. Pinelli, R. Alinovi, M. Erreni, S. Rossi, G. Condorelli, H. Post, A. Tampieri, and D. Catalucci, 2018. Inhalation of peptide-loaded nanoparticles improves heart failure. *Science Translational Medicine* 10.

3. Degli Esposti, L., F. Carella, A. Adamiano, A. Tampieri, and M. Iafisco, 2018. Calcium phosphate-based nanosystems for advanced targeted nanomedicine. *Drug Development and Industrial Pharmacy* 44:1223–1238.

4. Adamiano, A., M. Iafisco, M. Sandri, M. Basini, P. Arosio, T. Canu, G. Sitia, A. Esposito, V. Iannotti, G. Ausanio, E. Fragogeorgi, M. Rouchota, G. Loudos, A. Lascialfari, and A. Tampieri, 2018. On the use of superparamagnetic hydroxyapatite nanoparticles as an agent for magnetic and nuclear in vivo imaging. *Acta Biomaterialia* 73:458–469.

5. Michler, C., A. N. Cookson, R. Chabiniok, E. Hyde, J. Lee, M. Sinclair, T. Sochi, A. Goyal, G. Vigueras, D. A. Nordsletten, and N. P. Smith, 2013. A computationally efficient framework for the simulation of cardiac perfusion using a multi-compartment Darcy porous-media flow model. *International Journal for Numerical Methods in Biomedical Engineering* 29:217–232.

6. Ghekiere, O., R. Salgado, N. Buls, T. Leiner, I. Mancini, P. Vanhoenacker, P. Dendale, and A. Nchimi, 2017. Image quality in coronary CT angiography: Challenges and technical solutions. *British Journal of Radiology* 90.

7. Hyde, E. R., C. Michler, J. Lee, A. N. Cookson, R. Chabiniok, D. A. Nordsletten, and N. P. Smith, 2013. Parameterisation of multi-scale continuum perfusion models from discrete vascular networks. *Medical and Biological Engineering and Computing* 51:557–570.

8. Lee, J., A. Cookson, R. Chabiniok, S. Rivolo, E. Hyde, M. Sinclair, C. Michler, T. Sochi, and N. Smith, 2015. Multiscale Modelling of Cardiac Perfusion. *In* A. Quarteroni, editor, Modelling the Heart and the Circulatory System, Springer, Cham.

9. Hyde, E. R., A. N. Cookson, J. Lee, C. Michler, A. Goyal, T. Sochi, R. Chabiniok, M. Sinclair, D. A. Nordsletten, J. Spaan, J. P. H. M. Van Den Wijngaard, M. Siebes, and N. P. Smith, 2016. Multi-scale parameterisation of a myocardial perfusion model using whole-organ arterial networks. *Annals of Biomedical Engineering* 42:797–811.

10. Chapelle, D., J.-F. Gerbeau, J. Sainte-Marie, and I. E. Vignon-Clementel, 2010. A poroelastic model valid in large strains with applications to perfusion in cardiac modeling. *Computational Mechanics* 46:91–101.

11. Cookson, A. N., J. Lee, C. Michler, R. Chabiniok, E. Hyde, D. Nordsletten, and N. P. Smith, 2014. A spatially-distributed computational model to quantify behaviour of contrast agents in MR perfusion imaging. *Medical Image Analysis* 18:1200–1216.

12. Carboni, E., K. Tschudi, J. Nam, X. Lu, and A. W. K. Ma, 2014. Particle Margination and Its Implications on Intravenous Anticancer Drug Delivery. *AAPS PharmSciTech* 15:762–771.

13. Toy, R., E. Hayden, C. Shoup, H. Baskaran, and E. Karathanasis, 2011. The effects of particle size , density and shape on margination of nanoparticles in microcirculation. *Nanotechnology* 22:115101.

14. Namdee, K., A. J. Thompson, P. Charoenphol, and O. Eniola-adefeso, 2013. Margination Propensity of Vascular-Targeted Spheres from Blood Flow in a Micro fl uidic Model of Human Microvessels. *Langmuir* 29:2530–2535.

15. Alnaes, M. S., J. Blechta, J. Hake, A. Johansson, B. Kehlet, A. Logg, C. Richardson, J. Ring, M. E. Rognes, and G. N. Wells, 2015. The FEniCS Project Version 1.5. *Archive of Numerical Software* 3.

16. Ivanov, K. P., M. K. Kalinina, and Y. I. Levkovich, 1981. Flow Velocity in Capillaries of Brain and Muscles Physiological Significance. *Microvascular Research* 22:143–155.

17. Balaban, G., H. Finsberg, H. H. Odland, M. E. Rognes, S. Ross, J. Sundnes, and S. Wall, 2017. High-resolution Data Assimilation of Cardiac Mechanics. *International Journal for Numerical Methods in Biomedical Engineering* 33.

SUPPLEMENTARY MATERIAL

An online supplement to this article can be found by visiting BJ Online at http://www.biophysj.org.

SUPPLEMENTARY MATERIAL

Implementation of the particle tracking method

We use the finite element method to solve the model equations. The variational form of the multi-compartment Darcy model reads: Find $p_i \in H^1(\Omega)$ such that $\forall q_i \in H^1(\Omega)$

$$(\mathbf{K}_i \cdot \nabla p_i, \nabla q_i)_\Omega + (\Psi_i, q_i)_{\Gamma_N} + \left(\sum_{k=1}^{3} \beta_{i,k}(p_i - p_k), q_i \right)_\Omega - (s_i, q_i)_\Omega = 0 \tag{14}$$

Because blood flows from large arterioles to small arterioles to capillaries, we set the source terms $\Psi_2 = \Psi_3 = 0$, while Ψ_1 governs inflow into the myocardium. Analogously we set sink terms $s_1 = s_2 = 0$ and $s_3 = -0.1\,\mathrm{Pa}^{-1}\,\mathrm{s}^{-1}(p_3 - 3\,\mathrm{kPa})$ following (5). Additionally, exchange coefficients β are restricted to only allow exchange from compartment 1 to compartment 2, and from compartment 2 to compartment 3, such that $\beta_{1,3} = \beta_{2,1} = \beta_{3,1} = 0$. Here, we differ from the approach by Michler et. al. (5), who allowed symmetrical β, that is, they additionally allowed flow from the capillary compartment to the arteriole compartment, and from the arteriole compartment to the artery compartment. Velocities $\mathbf{w}_i$ are then easily calculated from p_i using Darcy's law with porosities Φ_i.

$$\mathbf{w}_i = -\Phi_i \mathbf{K}_i \nabla p_i. \tag{15}$$

and serve as the input for the particle tracking method. In the large and small arteriole compartments, each bolus of nanoparticles is advected according to the first order advection model (5). In the capillary compartment, concentrations follow diffusion kinetics, whose variational form reads: Find $c \in H^1(\Omega)$ such that $\forall v \in H^1(\Omega)$

$$\left(\frac{\partial c}{\partial t}, v \right)_\Omega + (D\nabla c, \nabla v)_\Omega = 0. \tag{16}$$

The model is implemented in Python 3.5.2 using the finite element framework FEniCS (15). In FEniCS the implementation of the variational form is very similar to the mathematical formulation, allowing for a simple conversion of equations to code. Coupled equations containing multiple unknowns like the multi-compartment formulation of Darcy's law are implemented using a mixed finite element:

```
N = 3 # number of compartments
P1 = FiniteElement('P', mesh.ufl_cell(), 1)
MixedElem = MixedElement([P1, P1, P1])
FS = FunctionSpace(mesh, MixedElem)
q = TestFunctions(FS)
p = TrialFunctions(FS)

# Variational form of the multi-compartment Darcy's law model
s3 = Constant(-0.01/(Pa*s))*(p[-1]-Constant(22*mmHg))
F = sum([inner(K[i]*grad(p[i]), grad(q[i])) for i in range(N)])*dx\
    + sum([q[i]*beta[i]*(p[i]-p[i+1]) for i in range(N-1)])*dx\
    + sum([q[i]*beta[i-1]*(p[i]-p[i-1]) for i in range(1,N)])*dx\
    - Psi1*q[0]*ds(IN) - s3*q[-1]*dx
```

In compartments 1 and 2 (large and small arterioles) permeability $\mathbf{K}_{1,2}$ depends on the muscle fiber directions of the left ventricle, which have been calculated using a rule-based algorithm (17) and imported from file as a Function on a VectorFunctionSpace. Permeability tensors $\mathbf{K}_1, \mathbf{K}_2, \mathbf{K}_3$ are constructed using

```
fiber = Function(VectorFunctionSpace(mesh, 'P', 1), filename)
d = p[0].geometric_dimension()
I = Identity(d) # Identity tensor
K1 = Constant(k1)*inner(fiber, fiber)
K2 = Constant(k2)*inner(fiber, fiber)
K3 = Constant(k3)*I
K = [K1, K2, K3]
```

where k1, k2, k3 refer to parameters K_1, K_2, K_3 in Table 2. Velocities $\mathbf{w}_i$ are projected onto quadratic Lagrange (P2) elements according to Darcy's law with porosities Φ_i.

```
P2 = VectorFunctionSpace(mesh, 'P', 2)
for i in range(N):
    w[i] = project(-Phi[i]*K[i]*grad(p[i]), P2)
```

Using $\mathbf{w}_i$ the particle tracking method was adapted from the fenicstools package (https://github.com/mikaem/fenicstools). Here, a particle's position $\mathbf{x}_i$ in compartment i is updated at each timestep according to velocity $\mathbf{w}_i$ (6) via

```
import numpy
x[:] = x[:] + dt*numpy.dot(coefficients, basis_matrix)[:]
```

where coefficients refers to the components of $\mathbf{w}_i$ and basis_matrix refers to the algebraic basis spanning the function space of $\mathbf{w}_i$. Exchange of particles between compartments follows a random selection using exchange coefficients γ_i as the probability that a particle travels from compartment 1 to compartment 2.

```
for cell in particle_map_1.keys():
    # Randomly determine whether a particle is removed from cell
    if numpy.random.rand() < gamma:
        # Particle to be removed
        particle = particle_map_1[cell][0]
        # Add particle to particle_map of compartment 2
        particle_map_2.__add__((mesh, cell, particle))
        # Remove particle from particle_map of compartment 1
        particle_map_1.pop(cell, 0)
```

The variables particle_map_1 and particle_map_2 refer to Python dictionaries containing the mesh cell IDs that carry particles. Following exchange from compartment 2 to compartment 3 the nanoparticle bolus is released and nanoparticles are tracked as a concentration that distributes via isotropic diffusion (7)

```
N = total_number_of_particles()
# MeshFunction that stores particles to be released
particles = MeshFunction("size_t", mesh, mesh.topology().dim())
particles.set_all(0)

# Create list of particles to be released
pop_list = []
for cell in particle_map_2.keys():
    if numpy.random.rand() < gamma:
        pop_list.append(cell)
        particle_map_2.pop(cell, 0)

# Set MeshFunction values to 1 if cell contains particles to be released
for cell in pop_list:
    particle_map.pop(cell, 0)
    particles.set_value(cell, 1)

# Iterator over cells containing particles to be released
domain_cells = SubsetIterator(particles, 1)
dofmap = u.function_space().dofmap()
# Get list of dof with particle release events
dofs = sum((dofmap.cell_dofs(cell.index()).tolist() for cell in domain_cells), [])

# Number of released particles
M = N - total_number_of_particles()
# If a particle is released over more than one dof, scale accordingly
scale = M/len(dofs)
# Add released particles to function space of compartment 3
u.vector()[dofs] += scale*c
```

Nanoparticles that leave compartment 2 are recorded in a MeshFunction, which is then used to obtain the corresponding degrees of freedom (dof) on the mesh. A scaling variable is calculated to take into account that multiple dof belong to an element and the nanoparticle concentration for one element should be evenly distributed over the dof. The nanoparticle concentration in compartment 3 is stored in u (P1) using and is updated according to (7) following the compartment exchange

```
FS = FunctionSpace(mesh, 'P', 1)
u = TrialFunction(FS)
v = TestFunction(FS)
un = Function(FS) # concentration at previous time step

# Time discretisation following Crank-Nicolson scheme
theta = 0.5
U = Constant(theta)*u + Constant(theta)*un

# Variational form for any number of compartments
k = Constant(1/dt)
F = k*(u-un)*v*dx + D*dot(grad(U), grad(v))*dx
```

We use a direct solver (MUMPS) for both the variational forms for the multi-compartment Darcy's law equations, and nanoparticle tracking equations. Time discretisation of the diffusion equation is achieved using a Crank-Nicholson scheme. All simulations were performed on a laptop with four Intel Core i7-7700HQ 2.80 GHz CPUs (Intel Corp., Santa Clara, CA, USA) and 16 GB RAM.