






# HPC Optimization Algorithm for Assessing Haemodynamic Parameters in Synthetic Patient Cohorts

Artem V. Rogov<sup>1,2</sup> , Timur M. Gamilov<sup>1,3</sup> , Yaroslav Yu. Kirichenko<sup>4</sup> , Philipp Yu. Kopylov<sup>1</sup> , Sergey S. Simakov<sup>2,3,4</sup> 

© The Authors 2024. This paper is published with open access at SuperFri.org

A computational framework for the generation of a synthetic pulse wave database is developed. This framework demonstrates the feasibility of generating large-scale, high-fidelity virtual patient cohorts for biomedical research. Pulse waves are generated using a one-dimensional hemodynamic model of the systemic circulation coupled with a model of the left heart. Each virtual patient in the database is defined by a set of physiological parameters, including systolic and diastolic blood pressure, stroke volume, and heart rate. The parameters are optimized to match the desired outputs by solving an inverse problem using the Unscented Kalman Filter (UKF). The UKF is selected for its ability to accurately and efficiently estimate parameters in nonlinear systems. The generation of a single virtual patient requires between one and several hundred iterations of the UKF, depending on the complexity of the desired outputs. To meet the computational demands of generating a database with thousands of virtual patients, a computing cluster with 24 CPU nodes, each containing 52 cores, is employed. Two levels of parallelization are implemented, resulting in a speedup factor of 8.

*Keywords:* Unscented Kalman Filter, hemodynamic model, inverse problem, synthetic data base, virtual population, parallel computing, high-performance computing.

## Introduction

Synthetic patient cohorts are becoming increasingly popular in clinical trials. They make it possible to generate synthetic data that is very close to real data, without having to deal with financial costs or ethical restrictions. Synthetic data can be used to design clinical trials, develop novel educational tools, or perform preliminary tests of various diagnostic techniques. This is particularly important in the age of data mining and artificial intelligence. Synthetic databases can be generated with machine learning and neural networks, or they can be used to train a new diagnostic method based on machine learning.

There are many ways to generate synthetic patient data. Some of them are based on statistical simulations [13, 14]. These methods analyse existing real patient cohorts and look for different patterns and distributions that can be used to generate a new virtual patient. Another approach is based on mathematical models of physiological processes [2, 7]. In this case, we assume that the process can be described by a system of differential and/or algebraic equations. Each virtual patient is associated with a set of input parameters. Once the input parameters are defined, we can solve the system of equations and calculate physiological characteristics: pulse waves [7], blood pressures [21], body weight changes [11], etc. It can be expensive or time-consuming to measure these characteristics in large numbers of patients. An example of this approach are synthetic pulse wave databases generated using blood flow models [7, 24].

<sup>1</sup>World-Class Research Center “Digital Biodesign and Personalized Healthcare”, Sechenov First State Medical University, Moscow, Russian Federation

<sup>2</sup>Moscow Institute of Physics and Technology, Dolgoprudny, Russian Federation

<sup>3</sup>Marchuk Institute of Numerical Mathematics of the Russian Academy of Sciences, Moscow, Russian Federation

<sup>4</sup>I.M. Sechenov First Moscow State Medical University (Sechenov University), Moscow, Russian Federation

Synthetic pulse wave databases consist of thousands of virtual patients with unique characteristics: blood pressure, heart rate, stroke volume, etc. Each synthetic database should represent a subset of the real world demographics. This is achieved by assigning some target characteristics to each virtual patient. Some of these characteristics can be associated with input parameters of the blood flow model, but some are associated with output values (blood pressure). As a result, one must solve an inverse problem [18] to obtain a desired virtual patient. An inverse problem is usually a computationally expensive task that requires numerous simulations to solve. A number of methods have been developed to approach an inverse problem. Some modern methods utilize physics-informed neural networks [10], which are used in 3D computations when the boundary information is difficult to model. A more classical approach involves Bayesian optimization [8] or Kalman filter [23].

There are several types of algorithms used in conjunction with Kalman filtering. All of these algorithms are used to solve nonlinear optimisation problems and are primarily used in navigation systems, control systems and signal processing. The algorithm allows system feedback (e.g., noisy experimental data) to be incorporated into the estimation of system state parameters.

The Unscented Kalman Filter (UKF) is a modification of Kalman filter that has been widely used for parameter estimation in discrete-time dynamic models, particularly in biomedical applications. For example, in [6], the UKF is used to estimate terminal resistances and arterial wall parameters from experimental flow and pressure data. Similarly, [19] demonstrates the use of UKF to estimate absolute blood volume during haemodialysis based on dynamic measurements of hematocrit and ultrafiltration rates. Another related study focuses on dialysis patients with an emphasis on personalised treatment approaches [1]. In addition, in [3] the UKF is used to estimate parameters of the Windkessel model and arterial wall stiffness.

The main focus of this work is the Unscented Kalman Filter (UKF), which offers several advantages. The first advantage is the fact that the UKF considers an exact nonlinear system, unlike the modifications and other optimisation methods which use a linearised form. The second advantage is speed. To capture the current state, which is assumed to be a Gaussian random variable distribution, only a minimal set of so-called sigma points is used. The third advantage is robustness. The UKF is robust in estimating the posterior distribution. It accurately approximates both the covariance and the mean to third order precision for any type of non-linearity [23].

An overview of the relationship and differences between the UKF and other algorithms, as well as a brief description, mathematical background and fundamentals can be found in [23]. A detailed description of the UKF and a demonstration of some of its most important properties, as well as the definition of sigma points, can be found in [17].

We propose a computational framework that uses UKF to generate a virtual patient from a one-dimensional hemodynamic model. One-dimensional hemodynamic used to generate virtual patients is based on the ADAN56 structure of systemic circulation [5]. ADAN56 includes 56 major arteries of systemic circulation and elastic compartments connected to the terminal vessels to simulate peripheral arteries. The inlet of the aorta is connected to two-chamber model of the left heart that take into account the angle of valve opening.

Each virtual patient has three target output characteristics: systolic blood pressure, diastolic blood pressure and stroke volume. Target input characteristics include heart rate, left ventricular ejection time and aortic diameter. To achieve the target output characteristics, a set of 13 model parameters are adjusted. Adjusting the model parameters for each virtual patient

involves dozens and sometimes hundreds of calculations with a blood flow model. At the same time, a synthetic database should include a few thousand virtual patients. This means that the generation of a single database requires around  $10^5$ – $10^6$  blood flow simulations of the full systemic circulation. This computationally intensive task is handled with the help of a computing cluster with 24 CPU nodes, each with 52 cores. The unscented Kalman filter allows effective parallelization of the computations. We describe two types of parallelization and estimate the reduction in computation time. After implementation of the optimizations computational time was reduced from 700 days to 10 days.

The remainder of this paper is organised as follows. Section 1.1.1 presents the description of 1D blood flow model. Section 1.1.2 describes lumped model of the left heart. Inverse problem is formulated in section 1.2. Unscented Kalman Filter and it's modification that is used to solve the inverse problem are described in sections 1.3 and 1.4. Section 2.1 presents calculation results: we estimate the amount of UKF iterations required to solve an inverse problem and estimate the speedup achieved with the help of parallelization. We discuss results and possible future work in section 2.1. The list of abbreviations used is presented in the end of the paper.

## 1. Methods

### 1.1. One-dimensional Hemodynamic Model with Lumped Heart

#### 1.1.1. One-dimensional hemodynamic model

We generate pulse waves of virtual patients with the help of one-dimensional model of blood flow in systemic circle [9, 21]. This model represents each artery as one-dimensional elastic tube. The blood is assumed to have constant density  $\rho = 1.06 \text{ g/cm}^3$  and constant viscosity  $\mu = 4 \text{ mPa}\cdot\text{s}$ . For each artery with index  $k$  we formulate mass and momentum conservation laws:

$$\frac{\partial A_k}{\partial t} + \frac{\partial(A_k u_k)}{\partial x} = 0, \quad (1)$$

$$\frac{\partial u_k}{\partial t} + \frac{\partial(u_k^2/2 + p_k/\rho)}{\partial x} = \psi_k, \quad (2)$$

where  $x$  is the coordinate along the artery,  $t$  is time,  $A_k(t, x)$ ,  $u_k(t, x)$ ,  $p_k(t, x)$  are cross-sectional area, blood flow velocity (averaged over cross-section), and blood pressure, respectively. The  $\psi_k$  is the friction force:

$$\psi_k = -\frac{8\pi\mu u_k}{\rho A_k}. \quad (3)$$

The tube law closes the system (1)–(2) and describes the elastic properties of the artery:

$$p_k(S_k) = \rho c_k^2 f(A_k), \quad (4)$$

where  $c_k$  is the speed of small disturbances propagation.  $f(A_k)$  is a piecewise function:

$$f(S_k) = \begin{cases} \exp(A_k/A_k^0 - 1) - 1, & A_k > A_k^0, \\ \ln(A_k/A_k^0), & A_k \leq A_k^0. \end{cases} \quad (5)$$

$A_k^0$  is a cross-section when the pressure is zero.  $A_k^0$  can be derived from the diameter of the artery  $d_k^0$ :  $A_k^0 = \frac{\pi(d_k^0)^2}{4}$ .

At arterial junctions points we impose mass conservation law and continuity of total pressure. Conservation of mass implies that the algebraic sum of flows in vessels at the junction is zero:

$$\sum_{k=k_1, \dots, k_M} Q_k \epsilon_k = 0, \quad (6)$$

where  $M$  is the number of arteries attached to the junction;  $Q_k = u_k(t, \tilde{x}_i) A_k(t, \tilde{x}_i)$ ;  $\tilde{x}_k = L_k$  and  $\epsilon_k = 1$  for incoming arteries ( $L_k$  is the length of the artery);  $\tilde{x}_k = 0$  and  $\epsilon_k = -1$  for outgoing arteries.

Continuity of total pressure is expressed as:

$$p_i(A_i(t, \tilde{x}_i)) + \frac{\rho u_i^2(t, \tilde{x}_i)}{2} = p_j(A_j(t, \tilde{x}_j)) + \frac{\rho u_j^2(t, \tilde{x}_j)}{2}, \quad (7)$$

where  $i$  and  $j$  are indices of two arteries connected to the junction. Index  $i$  is fixed and index  $j$  runs through the indices of all other connected arteries,  $i \neq j$ .

All arteries are connected into a network that represents arterial part of systemic circulation. Lengths, diameters and the structure of the network correspond to ADAN56 model [5], which contains 56 largest arteries of systemic circulation under normal physiological conditions.

The inlet of the aorta is connected to the two-chamber lumped parameter left heart model described in section 1.1.2. The outflow conditions assume that the systemic arteries are connected to the 3-element Windkessel compartments, which represent the peripheral vasculature. Thus, at the end of each terminal vessel with an index  $k$  we impose the following condition:

$$Q_k(t, L_k) \left(1 + \frac{R_1}{R_2}\right) + C R_1 \frac{dQ_k(t, L_k)}{dt} = \frac{p_k(t, L_k) - p_{out}}{R_2} + C \frac{dp_k(t, L_k)}{dt}, \quad (8)$$

where  $Q_k(t, L_k) = A_k(t, L_k) u_k(t, L_k)$ ;  $R_1, R_2, C$  are two resistances and compliance of Windkessel compartments;  $p_{out}$  is the outflow pressure (same for all compartments). Parameters  $R_1, R_2, C$  are individual for each terminal vessel. They are defined in ADAN56 model [5].

Equations (1) and (2) are solved in inner points of each artery with the help of explicit grid-characteristic scheme. The mass conservation law and total pressure continuity equations (7) together with compatibility conditions of the set (1)–(2) are solved with the help of Newton's method. Details of numerical implementation are presented in [20]. This approach allows us calculate values of cross-sectional area, velocity and pressure at the new time step independently in all arteries and junction points. It simplifies parallelization of calculations since each artery and each junction can be assigned to a separate thread.

### 1.1.2. Lumped parameter model of left heart with valve dynamics

In this part, we introduce a set of differential equations that describe the dynamics of the heart chambers, specifically the left atrium (la) and left ventricle (lv). These chambers are connected to the pulmonary veins (plv) and the aortic root. Additionally, the dynamics of the mitral valve (mv) and aortic valve (av) are also described. In contrast to predefined cardiac output [7], this model incorporates heart disease dynamics such as valve regurgitation.

The heart chambers volumes follow equation

$$I_k \frac{d^2 V_k}{dt^2} + R_k P_k \frac{dV_k}{dt} + E_k(t)(V_k - V_k^0) + P_k^0 = P_k, \quad k = la, lv, \quad (9)$$

where  $k$  is the index of chambers;  $I_k$  is the coefficient of inertia;  $R_k$  is the coefficient of hydraulic resistance;  $V_k$  is the pressure averaged over the chamber volume;  $P_k$  is the pressure averaged over the chamber volume;  $P_0$  and  $V_0$  are reference values of pressure and volume.  $E_k(t)$  reflects variable elasticity during contraction [16].

$$E(t) = E^d + \frac{E^s - E^d}{2}e(t), \quad 0 \leq e(t) \leq 1, \quad (10)$$

where  $e(t)$  represents periodic activation potential. For the left ventricle the function is:

$$e_{lv}(t) = \begin{cases} 0.5(1 - \cos(\frac{t}{T_{s1}}\pi)), & 0 \leq t < T_{s1}, \\ 0.5(1 - \cos(\frac{t-T_{s2}}{T_{s1}-T_{s2}}\pi)), & T_{s1} \leq t < T_{s2}, \\ 0, & T_{s2} \leq t \leq T. \end{cases} \quad (11)$$

$T = \frac{60}{HR}$  is the duration of cardiac cycle, which can be derived from the heart rate (HR);  $T_{s1}$  is the time to systolic peak;  $T_{s2}$  is the duration of the systole which we assume to be equal to left ventricular ejection time (LVET). For left atrium the function is:

$$e_{la}(t) = \begin{cases} 0, & 0 \leq t < T_{pb}, \\ 0.5(1 - \cos(\frac{t-T_{pb}}{T_{pw}}2\pi)), & T_{pb} \leq t < T, \end{cases} \quad (12)$$

where  $T_{pb}$  is the beginning of the atrial contraction and  $T_{pw}$  is the duration of the atrial contraction.

The mass conservation law for left atrium and left ventricle can be expressed as:

$$\begin{aligned} \frac{dV_{la}}{dt} &= Q_{plv} - Q_{mv}, \\ \frac{dV_{lv}}{dt} &= Q_{mv} - Q_{av}. \end{aligned} \quad (13)$$

The relationship between flow and pressure drop across left atrial inlet ( $Q_{plv}, \Delta P_{plv}$ ), mitral valve ( $Q_{mv}, \Delta P_{mv}$ ) and aortic valve ( $Q_{av}, \Delta P_{av}$ ) is defined as:

$$\Delta P_m = L_m \frac{dQ_m}{dt} + \beta_m Q_m |Q_m|, \quad m = plv, mv, av. \quad (14)$$

Here  $L_m$  is constant,  $\beta_m$  depends on area of the valve according to [25, 26]:

$$\begin{aligned} \beta_{mv} &= \frac{\rho}{2B_{mv}} \left( \frac{1}{A_{mv}} - \frac{1}{A_{ref}} \right)^2, \\ \beta_{av} &= \frac{\rho}{2B_{av}} \left( \frac{1}{A_{av}} - \frac{1}{A_a} \right)^2, \\ \beta_{plv} &= const. \end{aligned} \quad (15)$$

$B_{mv}$  and  $B_{av}$  are constant parameters;  $A_{mv}$ ,  $A_{av}$  are valve effective cross-section areas (depend on the angle);  $A_a$  is cross-sectional area of aorta at the inlet of the network of arteries;  $A_{ref} = 5 \text{ cm}^2$ .

For each valve effective cross-section area  $A_v$  can be calculated from the maximum cross-section area  $A_v^{max}$  and opening angle  $\theta$  as  $A_v(\theta) = g(\theta)A_v^{max}$ , where  $g(\theta)$  is a smooth function

that equals 1 when the valve is fully open ( $\theta = \theta_{max}$ ) and 0 when the valve is fully closed ( $\theta = \theta_{min}$ ):

$$g(\theta) = \begin{cases} \frac{(1 - \cos \theta_{min})^2}{(1 - \cos \theta_{max})^2}, & \theta < \theta_{min}, \\ \frac{(1 - \cos \theta)^2}{(1 - \cos \theta_{max})^2}, & \theta_{min} \leq \theta \leq \theta_{max}, \\ 1, & \theta > \theta_{max}. \end{cases} \quad (16)$$

The opening-closing dynamics of the valves are modeled using Newton's second law, where valve motion depends on the pressure gradient, friction, and a restoring force [15]:

$$\begin{aligned} \frac{d^2\theta_{mv}}{dt^2} &= (P_{la} - P_{lv})K_{mv}^p \cos\theta_{mv} - K_{mv}^f \frac{d\theta_{mv}}{dt} - F_{mv}(\theta_{mv}), \\ \frac{d^2\theta_{av}}{dt^2} &= (P_{lv} - P_a)K_{av}^p \cos\theta_{av} - K_{av}^f \frac{d\theta_{av}}{dt} - F_{av}(\theta_{av}), \end{aligned} \quad (17)$$

where  $\theta_{mv}$  and  $\theta_{av}$  are opening angles of mitral and aortic valves;  $K_{mv}^p$ ,  $K_{mv}^f$ ,  $K_{av}^p$ ,  $K_{av}^f$  are constants;  $P_a$  is the pressure in the aortic root. The virtual dumping force  $F(\theta)$  for each valve prevents it from opening more than maximum angle  $\theta_{max}$  and less than minimum angle  $\theta_{min}$ :

$$F(\theta) = \begin{cases} 0, & \theta_{min} < \theta < \theta_{max}, \\ e^{B(\theta - \theta_{max})} - 1, & \theta > \theta_{max}, \\ -(e^{B(\theta_{min} - \theta)} - 1), & \theta_{min} < \theta, \end{cases} \quad (18)$$

where  $B = 10^3$ .

Pressure drop across left atrial inlet  $\Delta P_{plv}$  in (14) is the difference between the pressure in pulmonary veins  $P_{plv}$  and the pressure in left atrium  $P_{la}$ . Pulmonary veins pressure  $P_{plv}$  is equal to 13 mm Hg by default but can be adjusted during parameter optimization.

## 1.2. Parameter Optimization Problem

The aim of the optimisation is to fit the model parameters to three key medical indicators: systolic blood pressure (SBP), diastolic blood pressure (DBP), and left ventricular stroke volume (SV). These three indicators were chosen for their clinical relevance and ease of measurement in practice [22]. SBP and DBP are associated with maximum and minimum blood pressures in the brachial artery of our arterial network. SV is the amount of blood ejected from the left ventricle into the aorta during a single cardiac cycle. The target values for SBP, DBP, and SV are indicated by the lower index  $t$ .

To achieve target values  $SBP_t$ ,  $DBP_t$  and  $SV_t$ , we adjust the following set of 13 parameters:  $R_{la}, R_{lv}$ ,  $I_{la}, I_{lv}$  in (9);  $K_{mv}^p, K_{av}^p$ ,  $K_{mv}^f, K_{av}^f$  in (17);  $p_{out}$  in (8);  $P_{plv}$ ; coefficients for total resistance and total compliance of all Windkessel compartments  $\kappa_R$  and  $\kappa_C$ ; coefficient for all  $c_k$  in (4)  $\kappa_s$ .

The coefficient  $\kappa_s$  for the speed of small disturbances propagation  $c_k$  is used to change the stiffness of all arteries at once. We do not change  $c_k$  for each artery independently, we multiply  $c_k$  of all arteries by a single coefficient. Similar coefficients are introduced for windkessel resistances  $\kappa_R$  and windkessel compliances  $\kappa_C$ . These three coefficients are used to modify the elastic properties of large arteries and peripheral vessels of a virtual patient.

The loss function  $\mathcal{L} = \mathcal{L}(SBP, DBP, SV)$  for optimization task is defined as:

$$\mathcal{L} = \max \left\{ \frac{|SBP - SBP_t|}{SBP_t}, \frac{|DBP - DBP_t|}{DBP_t}, \frac{|SV - SV_t|}{SV_t} \right\}. \quad (19)$$

We adjust parameters of a virtual patient until  $\mathcal{L} < 1\%$ .

Each virtual patient is defined by desired (target) SBP, DBP, SV, HR, LVET, and major arteries diameters. SBP, DBP, and SV were discussed above. HR corresponds to the duration of cardiac cycle in (11) and (12). LVET is equal to  $T_{s2}$  in (11). Diameters of arteries are used to calculate  $A_k^0$  in (5). Typical diameters of major arteries are taken from clinical literature [12].

To create a virtual population spanning different age groups, typical virtual patients were developed for each decade from 20 to 80 years of age [7]. Vessel dimensions and their standard deviations (SD) in diameter in an age group were adjusted based on age-specific data [12]. For each decade diameters of arteries were varied simultaneously and each artery can have three possible diameters: mean, mean + SD, mean - SD. Predefined SBP, DBP, SV, HR, LVET, and their standard deviations for each decade follow findings from [7].

The following parameters were varied for direct generation: SBP, DBP, SV, HR, LVET, diameters of large arteries. We modify the mean values of the parameters by incorporating the standard deviation, adjusting them both upwards and downwards. Consequently, each parameter takes on three possible values: the mean, the mean plus standard deviation, and the mean minus standard deviation. This approach generates  $3^6 = 729$  virtual patients for each age group.

In this work we focus on the age group of 20–30 years old. Using a baseline model for a 25-year-old patient, we optimize parameters to achieve target SBP, DBP, and SV values. Table 1 presents baseline and SD values for SBP, DBP, SV, HR, and LVET. Diameters of major arteries were modified simultaneously according to the data from [12].

**Table 1.** Parameter of a baseline 25 year old virtual patient [7]

Parameter	Value $\pm$ SD
SBP, mmHg	112.3 $\pm$ 8.7
DBP, mmHg	72.0 $\pm$ 5.6
SV, ml	73.0 $\pm$ 13.1
HR, beats/min	73.0 $\pm$ 9.1
LVET, ms	283 $\pm$ 23

### 1.3. The Unscented Kalman Filter for Parameter Estimation

The following section outlines the general structure of the Unscented Kalman Filter. Subsequently, it will present the specific implementation employed in this study. Time-discretized system of the one-dimensional hemodynamic model with left heart described in Section 1.1.2 can be presented as:

$$X_{n+1} = \tilde{F}(X_n, \Theta_n), \quad (20)$$

where  $X_n$  is vector, which represents state of the system at  $n$ -th iteration.  $X_n$  represents all values of pressure, flow, cross-sectional area in all point of all arteries at each moment of a cardiac cycle, along with the parameters of the Windkessel models and the lumped parameters associated with the left heart.  $\Theta_n$  is a set of 13 parameters to be estimated,

$\Theta = [R_{la}, R_{lv}, I_{la}, I_{lv}, K_{mv}^p, K_{av}^p, K_{mv}^f, K_{av}^f, p_{out}, \kappa_R, \kappa_C, \kappa_s]^\top$ ;  $\tilde{F}$  is operator of propagation of the system on next step, which depends on underlying model equations and particular type of discretization.

$$Z_n = \tilde{H}(X_n) + \epsilon, \quad (21)$$

where  $Z_n$  is vector of measured variables at some time instants,  $\tilde{H}$  is operator, which transforms state of the system to specific measured variables,  $\epsilon$  is a vector of Gaussian noise. Further in this paper it is assumed, that  $Z_n = [SBP_n, DBP_n, SV_n]^\top$ , and  $SBP_t, DBP_t, SV_t$  are target or required values for virtual patient. Bearing in mind that all these values can only be calculated after a complete cardiac cycle, one sampling step is considered as one cardiac cycle. For other tasks and applications, the temporal sampling interval may vary according to the specific objective and the frequency of measurements available for the system under consideration.

Algorithm of any Kalman filter can be represented as prediction-correction scheme for state  $(X_n, \Theta_n)$ , where prediction step calculates  $X_{n+1}^-$  and  $\Theta_{n+1}^-$ , assuming that parameters have trivial dynamic and do not change over time.

$$\begin{aligned} X_{n+1}^- &= \tilde{F}(X_n^+, \Theta_n), \\ \Theta_{n+1}^- &= \Theta_n. \end{aligned} \quad (22)$$

In the correction step the difference between the measurement and the one calculated according to the new state of the system is taken into account.

$$\begin{aligned} X_{n+1}^+ &= X_{n+1}^- + K_X(Z_{n+1} - \tilde{H}(X_{n+1}^-)), \\ \Theta_{n+1}^+ &= \Theta_{n+1}^- + K_\Theta(Z_{n+1} - \tilde{H}(X_{n+1}^-)). \end{aligned} \quad (23)$$

$K_X$  and  $K_\Theta$  are Kalman matrices defined in a way to minimize difference between actual measurement and observed state.

#### 1.4. The Reduced-order Unscented Kalman Filter

The aim of this section is to examine the reduced-order Unscented Kalman Filter (ROUKF), which is the UKF implementation used in this study. These two methods share core principles: both filters approximate the mean and covariance of a nonlinear system by propagating a set of sigma points through the nonlinear function, unlike other versions of the Kalman filter that use a linearised system; both filters employ an identical prediction-correction strategy for state and covariance estimation.

As shown in [17], the filtering operation can only involve matrices of the same size as the unknown parameter space. Furthermore, it is shown that to estimate  $p$  parameters, only  $p + 1$  sigma points are needed to capture the current state of the system. These sigma points can be defined recursively (the algorithm of this procedure is described in detail in [17]). Then, all the sigma points are integrated into a matrix  $I$ . This matrix has  $p + 1$  vectors of size  $p$ , where the notation  $I_{(i)}$  will represent one column of the  $I$  matrix and one concrete sigma point. The weights of the sigma points are stored in the diagonal matrix  $D$ , where the same and equal coefficients are on the diagonal  $\alpha = \frac{1}{p+1}$ .

The algorithm of ROUKF is presented in Algorithm 1. The stopping criterion for this algorithm can be chosen at the discretion of the researcher and is typically determined by the loss function. Once the criterion is met, the problem of selecting the parameter is considered to be solved. For more details check [4, 17].



---

**Algorithm 1** Iterative Process of UKF

---

1: **Initialization step:**

2: Define covariance matrices:

$$\Sigma^\Theta = \text{diag} \{ \sigma_{\text{param},i}^2 \}, \quad i = 1, \dots, p \quad (24)$$

$$\Sigma^Z = \text{diag} \{ \sigma_{\text{obs},i}^2 \}, \quad i = 1, \dots, m \quad (25)$$

3: For the first step ( $n = 0$ ), set the following matrices:

$$L^\Theta = \mathbb{1}, \quad L^X = \mathbf{0}, \quad U_0 = (\Sigma^\Theta)^{-1} \quad (26)$$

4: **while** loss function is higher than the threshold **do**

5:     **Sigma-point Sampling:**

$$C_n = \sqrt{U_n^{-1}} \quad (\text{Cholesky factorization}) \quad (27)$$

$$X_{n,(i)}^+ = X_n^+ + L_n^X C_n^\top I_{(i)}, \quad i = 1, \dots, p+1 \quad (28)$$

$$\Theta_{n,(i)}^+ = \Theta_n^+ + L_n^\Theta C_n^\top I_{(i)}, \quad i = 1, \dots, p+1 \quad (29)$$

6:     **Forward Propagation / Prediction:**

$$X_{n+1,(i)}^- = \tilde{F}(X_{n,(i)}^+, \Theta_{n,(i)}), \quad i = 1, \dots, p+1 \quad (30)$$

$$\Theta_{n+1,(i)}^- = \Theta_{n,(i)}, \quad i = 1, \dots, p+1 \quad (31)$$

$$X_{n+1}^- = \sum_{i=1}^{p+1} \alpha X_{n+1,(i)}^- \quad (32)$$

$$\Theta_{n+1}^- = \sum_{i=1}^{p+1} \alpha \Theta_{n+1,(i)}^- \quad (33)$$

7:     **Calculate Innovation:**

$$\Delta = Z_{n+1} - \tilde{H}(X_{n+1}^-) \quad (34)$$

8:     **Update Covariances:**

$$L_{n+1}^X = X_{n+1}^- D I^\top, \quad D = \text{diag} \{ \alpha, \dots, \alpha \}, \quad (35)$$

$$L_{n+1}^\Theta = \Theta_{n+1}^- D I^\top \quad (36)$$

$$L_{n+1}^\Delta = \Delta_{n+1} D I^\top \quad (37)$$

$$U_{n+1} = I D I^\top + (L_{n+1}^\Delta)^\top (\Sigma^Z)^{-1} L_{n+1}^\Delta \quad (38)$$

9:     **Correction:**

$$\Delta^{\text{sum}} = \sum_{i=1}^{p+1} \alpha \Delta_{(i)} \quad (\text{weighted sum of columns}) \quad (39)$$

$$X_{n+1}^+ = X_{n+1}^- - L_{n+1}^X U_{n+1}^{-1} (L_{n+1}^\Delta)^\top (\Sigma^Z)^{-1} \Delta^{\text{sum}} \quad (40)$$

$$\Theta_{n+1}^+ = \Theta_{n+1}^- - L_{n+1}^\Theta U_{n+1}^{-1} (L_{n+1}^\Delta)^\top (\Sigma^Z)^{-1} \Delta^{\text{sum}} \quad (41)$$

10:     Set iteration number  $n = n + 1$

11: **end while**

---

It is worth noting that the algorithmic implementation of the above problem has a number of peculiarities. The target parameters are  $Z_n = [SBP_n, DBP_n, SV_n]^T$ . In order to calculate  $SBP_n$ ,  $DBP_n$ , and  $SV_n$ , it is necessary to perform several cardiac cycles calculations until the values of SBP, DBP, and SV become constant from one cycle to another. The calculations are carried out until the periodic mode is established. For the purposes of this paper, we assume that a sufficient number of cardiac cycles is 9. Consequently, the determination of the set of states  $X_{n+1,(i)}^-$  via the propagation operator  $\tilde{F}$  is a computationally demanding task. It is necessary to compute the problem of fluid flow through a network of pipes  $p + 1$  times before it can be determined.

Furthermore, the vector of observations or the vector of target parameters  $Z_n$  remains constant, as the required values are invariant. This provides an alternative view of the algorithm. In more typical cases, the target vector will change over time, reflecting the evolution of the system.

The primary objective of the ROUKF is to overcome the limitations of the UKF. The UKF may require the implementation of costly matrix operations such as factorisation and inverse matrix computation on matrices of considerable dimension, equivalent to the number of state variables. Furthermore, if  $n$  represents the number of state variables, it propagates  $2n + 1$  sigma points through the nonlinear system. Unlike the UKF, the ROUKF focuses on a lower dimensional state, thereby reducing the number of sigma points and consequently the computational complexity.

## 2. Results

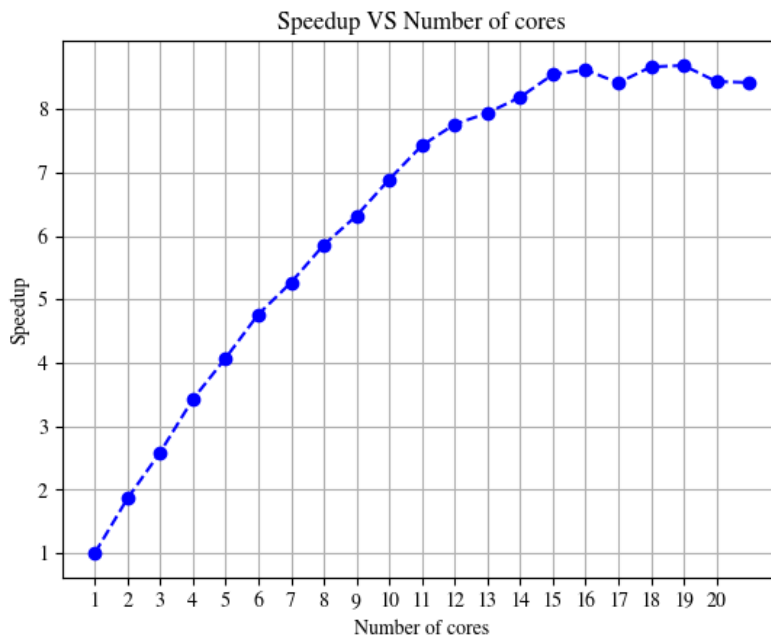
### 2.1. Parallelization of Virtual Patient Generation Process

The process of selecting the appropriate parameters to solve the inverse problem using the Kalman filter is time consuming. To complete a single iteration of the Kalman filter algorithm, the problem must be computed 13 times until a solution is found. In the absence of parallelization, the completion of a single calculation requires around 320 seconds. It is now necessary to estimate the time required to generate the entire virtual population of 25 year olds. As previously stated, a mean of 13 iterations is required for each update of the parameters (for a single iteration of UKF). The average number of iterations required for the entire process is several dozen, with an estimated mean of 20. The total number of objects in the population is 729, thus the generation will take more than 700 days, which is an unacceptable timespan. In this section, the methodology used to parallelize the generation of virtual populations is described in detail.

The first type of parallelization is used in the calculation of the result of the forward propagation operator, when the resulting vector  $X_n$  of  $SBP$ ,  $DBP$ ,  $SV$  is obtained from the set of parameters  $\Theta_n$ . The processes of calculating the values at the internal points of the vessels and at the points where the vessels are connected are both subject to parallelization.

The high-performance computational cluster of Sechenov University, comprising 24 CPU nodes with Intel(R) Xeon(R) Gold 6230R processors at 2.10GHz and 52 cores each, was employed for the computation. The cluster exhibited a peak performance of 84 tflops and a total distributed RAM capacity of 10.5 TB. The optimal number of cores for the calculation was determined by empirical experiments. Figure 1 illustrates the effect of core allocation on the speedup of the calculation of the vector of resulting values for a single virtual patient. An examination of the

data in the graph shows that it is pointless to exceed 15 cores for the specified calculation. In addition, there is no significant improvement in speed.



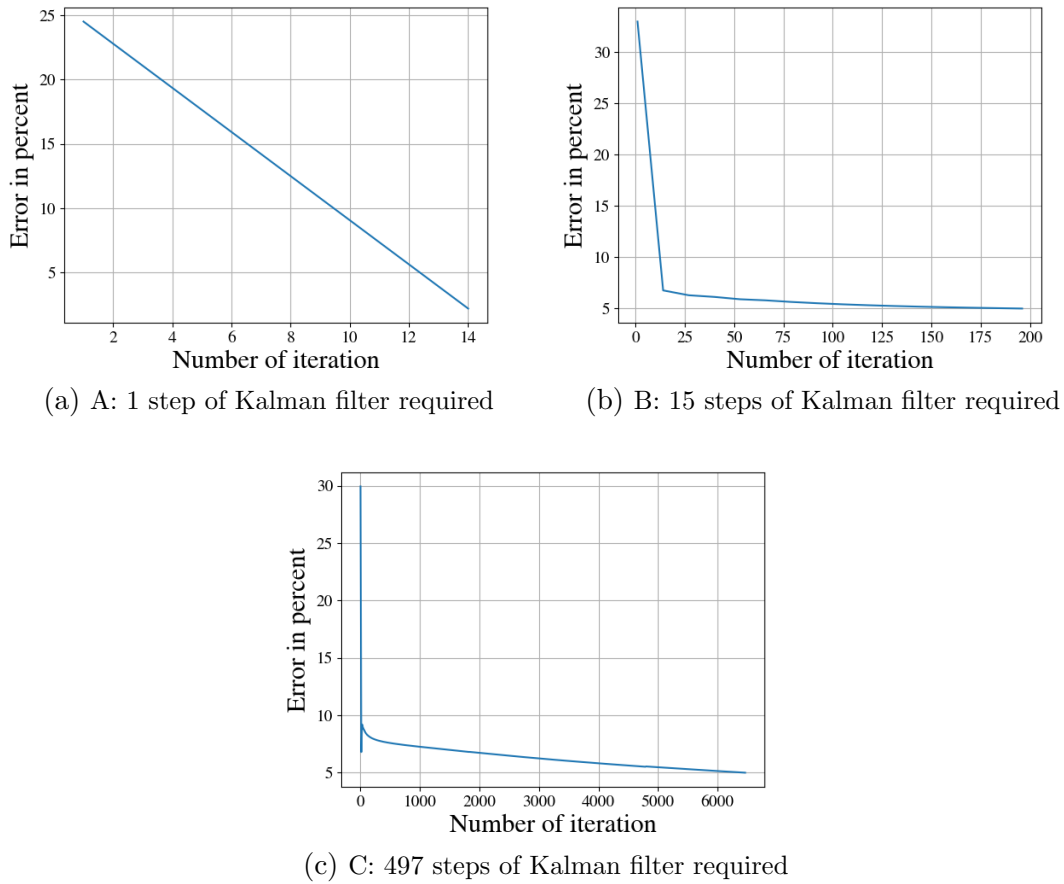
**Figure 1.** Dependence of calculation speedup on the number of cores

The second type of parallelization results from the fact that the parameter optimisation task is configured using a configuration file. This file contains a description of the geometric properties of the vessels as well as the values of the vascular and cardiac parameters. Therefore, it was convenient to organise the start of the parameter selection tasks by fixing the values of LVET and vessel diameters. Accordingly, once the configuration files had been fixed and all possible combinations of the corresponding parameters had been collected, 9 different tasks were identified. In other words, all virtual objects were divided into 9 different subsamples corresponding to different diameters of large vessels and LVET. These subsamples can be run in parallel on a cluster.

It should be noted that an additional way of speeding up the calculations is to distribute 13 independent tasks to different nodes of the cluster. These 13 tasks must be computed during the forward propagation step of the UKF algorithm. Due to the workload of the cluster, this feature is not implemented, but may be added in the future.

In consideration of the two aforementioned parallelizations, it can be estimated that the calculation of the entire virtual population will require approximately 9–10 days, which is consistent with the observed results.

The following three plots on Fig. 2 illustrate the convergence rate characteristic of the calculation of the required parameters for the whole sample. All cases can be classified into three categories based on their convergence characteristics: (A) fast convergence (1–2 filter steps, Fig. 2a), (B) “medium convergence” (several tens of filter steps, Fig. 2b), and (C) slow convergence (100 or more filter steps, Fig. 2c). The parameters of these cases are listed in Tab. 2.



**Figure 2.** Convergence rate of representative cases

**Table 2.** Parameters of cases presented in Fig. 2

Varying parameters	Case A	Case B	Case C
SBP, mmHg	121.0	112.3	112.3
DBP, mmHg	72.0	77.6	77.6
SV, ml	66.8	79.9	79.9
HR, beats/min	73.0	63.9	73.0
Diameters of major arteries	Decreased	Decreased	Decreased
LVET, ms	260	306	260

## Conclusion

In this paper, we have shown how the generation of synthetic patient cohorts can be facilitated using computer clusters. The process is automatic once the target values are defined. We have generated a synthetic pulse wave database based on a one-dimensional blood flow model of the systemic arterial circulation with a two-chambered lumped heart. This method can be used to study the effect of different cardiovascular pathologies on pulse wave shapes. The introduction of a lumped heart model with leaflet dynamics allows us to simulate databases with valvular disease.

By parallelizing and optimising computations, we can calculate hundreds of virtual patients in 10 days instead of months. In our example, we have simulated pulse waves for a healthy

synthetic population. The introduction of any cardiovascular pathology, such as aortic stenosis, would require a separate synthetic cohort of thousands of virtual patients. Ideally, we need a separate cohort for each degree of pathology and for each combination of pathologies. The amount of computation required for such a task is impossible without high performance optimisations.

We have used parallelization on two levels: parallelization of different branches and junctions of the blood flow model and parallelization of different virtual patients. It is possible to implement another level of parallelization: the distribution of the “forward propagation” of the UKF algorithm. This would allow us to distribute 13 independent tasks to different nodes of the cluster. This was not necessary due to the limitations of our cluster, but it provides opportunities for further optimisation if more computing power is available.

## Acknowledgements

This work was financed by the Ministry of Science and Higher Education of the Russian Federation within the framework of state support for the creation and development of World-Class Research Centers “Digital biodesign and personalized healthcare” № 075-15-2022-304.

### *Abbreviations*

The following abbreviations are used in this manuscript:

CPU	Central processing unit
DBP	Diastolic blood pressure
HPC	High performance computing
HR	Heart rate
LVET	Left ventricular ejection time
ROUKF	Reduced-order Unscented Kalman Filter
SD	Standard deviation
SBP	Systolic blood pressure
SV	Stroke volume
UKF	Unscented Kalman Filter

*This paper is distributed under the terms of the Creative Commons Attribution-Non Commercial 3.0 License which permits non-commercial use, reproduction and distribution of the work without further permission provided the original work is properly cited.*

## References

1. Abohtyra, R.M., Vincent, T.L.: Nonlinear parameter and state estimation approach in end-stage kidney disease patients. bioRxiv (2023). <https://doi.org/10.1101/2022.04.02.486844>
2. Allen, R.J., Rieger, T.R., Musante, C.J.: Efficient generation and selection of virtual populations in quantitative systems pharmacology models. *CPT Pharmacometrics Syst Pharmacol* 5(3), 140–146 (2016). <https://doi.org/10.1002/psp4.12063>
3. Arthurs, C., Xiao, N., Moireau, P., *et al.*: A flexible framework for sequential estimation of model parameters in computational hemodynamics. *Advances in Modeling and Simulation*

- in Engineering Sciences 7, 48 (2020). <https://doi.org/10.1186/s40323-020-00186-x>
4. Bertoglio, C., Moireau, P., Gerbeau, J.F.: Sequential parameter estimation for fluidstructure problems: Application to hemodynamics. *International Journal for Numerical Methods in Biomedical Engineering* 28(4), 434–455 (2012). <https://doi.org/10.1002/cnm.1476>
  5. Boileau, E., Nithiarasu, P., Blanco, P.J., *et al.*: A benchmark study of numerical schemes for one-dimensional arterial blood flow modelling. *International Journal for Numerical Methods in Biomedical Engineering* 31(10) (2015). <https://doi.org/10.1002/cnm.2732>
  6. Caiazzo, A., Caforio, F., Montecinos, G., *et al.*: Assessment of reduced-order unscented Kalman filter for parameter identification in 1-dimensional blood flow models using experimental data. *International Journal for Numerical Methods in Biomedical Engineering* 33(8), e2843 (2017). <https://doi.org/10.1002/cnm.2843>
  7. Charlton, P.H., Harana, J.M., Vennin, S., *et al.*: Modeling arterial pulse waves in healthy aging: a database for in silico evaluation of hemodynamics and pulse wave indexes. *American Journal of Physiology - Heart and Circulatory Physiology* 317(5), H1062–H1085 (2019). <https://doi.org/10.1152/ajpheart.00218.2019>
  8. Frazier, P.I.: A tutorial on bayesian optimization (2018)
  9. Gamilov, T., Danilov, A., Chomakhidze, P., *et al.*: Computational analysis of hemodynamic indices in multivessel coronary artery disease in the presence of myocardial perfusion dysfunction. *Computation* 12(6) (2024). <https://doi.org/10.3390/computation12060110>
  10. Garay, J., Dunstan, J., Uribe, S., Costabal, F.: Physics-informed neural networks for blood flow inverse problems (08 2023). <https://doi.org/10.48550/arXiv.2308.00927>
  11. Hall, K.: Predicting metabolic adaptation, body weight change, and energy intake in humans. *Am J Physiol Endocrinol Metab* 298(3), E449–66 (2009). <https://doi.org/10.1152/ajpendo.00559.2009>
  12. Hickson, S.S., Butlin, M., Graves, M., *et al.*: The relationship of age with regional aortic stiffness and diameter. *JACC.Cardiovascular Imaging* 3(12), 1247–1255 (2010). <https://doi.org/10.1016/j.jcmg.2010.09.016>
  13. Jeanson, F., Farkouh, M.E., Godoy, L.C., *et al.*: Medical calculators derived synthetic cohorts: a novel method for generating synthetic patient data. *Scientific Reports* 14, 11437 (2024). <https://doi.org/10.1038/s41598-024-61721-z>
  14. Khorchani, T., Gadiya, Y., Witt, G., *et al.*: SASC: A simple approach to synthetic cohorts for generating longitudinal observational patient cohorts from COVID-19 clinical data. *Patterns (N Y)* 3(4), 100453 (2022). <https://doi.org/10.1016/j.patter.2022.100453>
  15. Korakianitis, T., Shi, Y.: A concentrated parameter model for the human cardiovascular system including heart valve dynamics and atrioventricular interaction. *Medical Engineering & Physics* 28(7), 613–628 (2006). <https://doi.org/10.1016/j.medengphy.2005.10.004>
  16. Korakianitis, T., Shi, Y.: Numerical simulation of cardiovascular dynamics with healthy and diseased heart valves. *Journal of Biomechanics* 39(11), 1964–1982 (2006). <https://doi.org/10.1016/j.jbiomech.2005.06.016>

17. Moireau, P., Chapelle, D.: Reduced-order Unscented Kalman Filtering with application to parameter identification in large-dimensional systems. *ESAIM: Control, Optimisation and Calculus of Variations* 17(2), 380–405 (2011). <https://doi.org/10.1051/cocv/2010006>
18. Nolte, D., Bertoglio, C.: Inverse problems in blood flow modeling: A review. *International Journal for Numerical Methods in Biomedical Engineering* 38(8), e3613 (2022). <https://doi.org/10.1002/cnm.3613>
19. Rammah Abohtyra, T.V., Schneditz, D.: Magnitude and precision of absolute blood volume estimated during hemodialysis. *Renal Failure* 46(2), 2377781 (2024). <https://doi.org/10.1080/0886022X.2024.2377781>
20. Simakov, S., Gamilov, T., Soe, Y.N.: Computational study of blood flow in lower extremities under intense physical load. *Russian Journal of Numerical Analysis and Mathematical Modelling* 28(5), 485–504 (2013). <https://doi.org/10.1515/rnam-2013-0027>
21. Simakov, S., Timofeev, A., Gamilov, T., *et al.*: Analysis of the impact of left ventricular assist devices on the systemic circulation. *Russian Journal of Numerical Analysis and Mathematical Modelling* 35(5), 295–314 (2020). <https://doi.org/10.1515/rnam-2020-0025>
22. Stamler, J., Stamler, R., Neaton, J.D.: Blood pressure, systolic, and diastolic, and cardiovascular risks: Us population data. *Archives of Internal Medicine* 153, 598–615 (1993). <https://doi.org/10.1001/archinte.153.5.598>
23. Wan, E.A., Merwe, R.V.D.: The unscented Kalman filter for nonlinear estimation. In: *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*. pp. 153–158. Lake Louise, AB, Canada (2000). <https://doi.org/10.1109/ASSPCC.2000.882463>
24. Wang, T., Jin, W., Liang, F., Alastruey, J.: Machine learning-based pulse wave analysis for early detection of abdominal aortic aneurysms using in silico pulse waves. *Symmetry* 13(5) (2021). <https://doi.org/10.3390/sym13050804>
25. Young, D.F., Tsai, F.Y.: Flow characteristics in models of arterial stenoses. I. Steady flow. *Journal of Biomechanics* 6(4), 395–410 (1973). [https://doi.org/10.1016/0021-9290\(73\)90099-7](https://doi.org/10.1016/0021-9290(73)90099-7)
26. Young, D.F., Tsai, F.Y.: Flow characteristics in models of arterial stenoses. II. Unsteady flow. *Journal of Biomechanics* 6(5), 547–559 (1973). [https://doi.org/10.1016/0021-9290\(73\)90012-2](https://doi.org/10.1016/0021-9290(73)90012-2)