Proceedings Paper:

Smith, E. orcid.org/0000-0003-1559-2400, Nie, L. orcid.org/0000-0002-5796-907X, Carpenter, T.M. orcid.org/0000-0001-5676-1739 et al. (3 more authors) (2024) Utilising Haemodynamic Modelling in Volumetric Ultrasound Simulation. In: Proceedings of the 2024 IEEE Ultrasonics, Ferroelectrics, and Frequency Control Joint Symposium (UFFC-JS). 2024 IEEE Ultrasonics, Ferroelectrics, and Frequency Control Joint Symposium (UFFC-JS), 22-26 Sep 2024, Taipei, Taiwan. Institute of Electrical and Electronics Engineers (IEEE) , pp. 1-4.

https://doi.org/10.1109/uffc-js60046.2024.10794050

Utilising Haemodynamic Modelling in Volumetric Ultrasound Simulation

Elliott Smith[1*], Luzhen Nie[1], Thomas M. Carpenter[1], David M. J. Cowell[1], James McLaughlan[1,2] and Steven Freear[1]
[1]*School of Electronic and Electrical Engineering, University of Leeds, Leeds, UK*
[2]*Leeds Institute of Medical Research, Wellcome Trust Brenner Building, St James's University Hospital, Leeds, UK*
[*]e.i.j.smith@leeds.ac.uk

***Abstract*—This study proposes a Python framework (USim) for generating synthetic volumetric ultrasound datasets that incorporate haemodynamic modelling. This framework was demonstrated through the simulation of flowing particles in a simulated model of a human aorta, under interrogation from a volumetric diverging wave ultrasound emission. Particles in the beamformed lumen were localised and tracked between subsequent frames and were shown to be in agreement with the ground-truth model. Thus, the efficacy of the proposed framework for volumetric flow imaging algorithm development and validation has been demonstrated.**

***Index Terms*—acoustic wave simulation, computational fluid dynamics, haemodynamics, volumetric ultrasound imaging, flow imaging, high-frame-rate ultrasound**

I. INTRODUCTION

It is convenient to perform ultrasound-based cardiovascular functional algorithm validation using *in-silico* (simulated) datasets. The ability to manipulate the acoustic media, the acoustic sources and acoustic sensors is attractive during algorithm development. Vector flow imaging techniques [1] benefit from realistic haemodynamic modelling, allowing developers to assess algorithm utility in realistic scenarios. The emerging field of patient-specific, haemodynamic, modelling, driven by advances in physics informed machine learning [2], necessitates datasets derived from accurate Computational Fluid Dynamics (CFD) simulation for training and validation.

There are a number of freely available acoustic wave simulators that can produce synthetic radio frequency (RF) signals. These include Field II [3], k-Wave [4] and SIMUS [5], [6]. All have MATLAB® (MathWorks Inc., Natick, MA, USA) frameworks to simulate ultrasound imaging sequences. SIMUS and Field II utilise linear acoustics in homogeneous media. They are capable of calculating the acoustic response at arbitrary points in a Region of Interest (RoI), interactions between scatterers, however, are ignored. In contrast, k-Wave can model nonlinear propagation in heterogeneous media, calculations, however, are made on a regular grid.

Both SimVascular [7] and CRIMSON [8] are open-source haemodynamic simulators that utilise geometric models of blood vessels derived from diagnostic imaging. The Vascular Model Repository (VMR), a library of SimVascular compatible datasets, includes examples of the aorta, cerebral, coronary, aortofemoral and pulmonary arteries [9].

A Python framework (USim) has been developed to facilitate algorithm development for advanced anatomical and functional ultrasound imaging applications. The USim framework consists, firstly, of classes to design complex imaging simulation specifications. Secondly, of classes to import and export realistic phantom structures, including haemodynamic information, in common 3D formats used in CFD simulation packages. Thirdly, of a HDF5® (HDF Group, Champaign, IL, USA) file format structure for saving simulation specifications and results for maximum reuse.

The utility of the USim framework was demonstrated through the simulation of flowing particles in an *in-silico* model of a human aorta from the VMR. A high-frame-rate volumetric ultrasound imaging sequence, utilising a matrix array probe, was simulated. A k-Wave engine was used to perform the ultrasound simulation. Particles in the beamformed lumen were localised and tracked between subsequent frames.

II. METHOD

A. USim Simulations

USim is a Python framework consisting of classes that describe transducer, phantom, and simulation objects. As illustrated in Fig. 1, these objects allow the design of complex simulation specifications at a higher level of abstraction than the underlying acoustic wave simulator. The IT'IS database of tissue parameters is utilised by the framework to facilitate realistic medical ultrasound simulation [10].

Fig 2 illustrates the hierarchical structure of the USim classes for describing a simulation specification. A simulation can be associated with a list of sequences. A sequence describes a list of simulation events and can be associated with a single transducer. An event consists of a set of attributes to describe the transmission and reception of a single emission. This hierarchical approach has been adopted to maximise flexibility in the description of complex imaging sequences.

As illustrated in Fig. 1, the framework also consists of a simulation engine. This engine performs the conversion of the high-level simulation specification to the format required by the underlying acoustic wave simulator. In the case of a k-Wave simulation, the engine, firstly, performs the conversion and mapping of a phantom, transducer, and simulation event, to the acoustic sources, sensors and material parameters on the

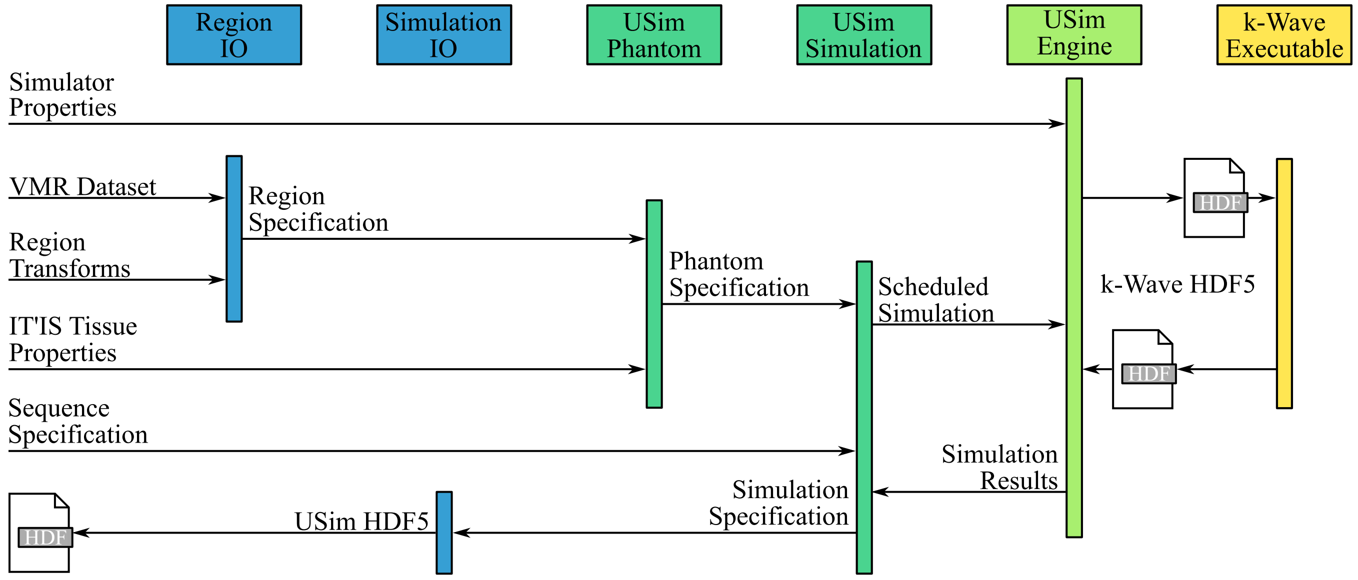

Fig. 1. The sequence of interactions between various components in the USim framework for incorporating Vascular Model Repository (VMR) datasets.

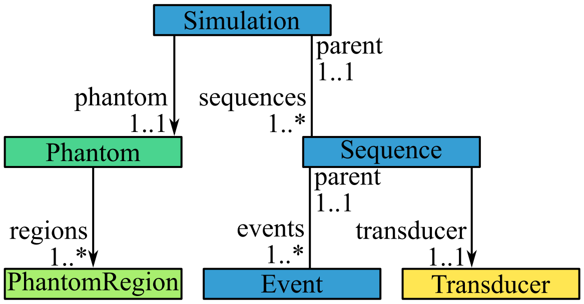

Fig. 2. The USim class structure for a simulation specification.

specified k-Wave grid. These simulation parameters are written to the k-Wave input file required by the k-Wave executable. Secondly, the engine initiates and monitors the simulation to be performed by the k-Wave executable. Upon completion, the simulation results are read from the k-Wave output file. Finally, the engine converts the simulated sensor pressure data into the channel data in the simulation specification.

B. *USim Phantoms*

As illustrated in Fig. 2, a simulation can be associated with a single phantom. A phantom is associated with a list of phantom regions which are used to define the acoustic material properties and structures of the phantom. The order of the phantom regions in the list determines the order in which they are applied to the acoustic wave simulation. Regions at the end of the list have a higher priority than those at the start and are overlaid on top of the preceding regions. Consequently, through a stack of multiple phantom regions, complex phantoms can be defined.

Each phantom region utilises a uniform grid to define the region structure. Both background material attributes and sparse material attributes can be described within a phantom region. The material properties for the background grid points are drawn randomly from a distribution defined by the background material attributes during simulation. The sparse material attributes, and a single arbitrary position at which these attributes should be used, need to be specified for every sparse point. The use of both background and sparse material properties allows flexibility in the definition of a phantom region, enabling a range of complex structures to be described.

For cardiac applications, the incorporation of realistic flow is beneficial. To incorporate flow, a mechanism to share both geometry and flow data between USim and external CFD simulators is required. The framework incorporates libraries to import geometries, and corresponding flow information, in standard 3D file formats. In the specific case of importing VMR datasets, Visualization Toolkit (VTK) file formats are supported. Through the specification of a series of transforms, a user can translate, scale and rotate the CFD dataset and map the data onto the regular grid of a phantom region.

Figure 3 illustrates a pair of USim phantom objects for the specific case of a human aorta imported from the VMR. The first region within both of the illustrated examples is a single, heterogeneous, background region with the material parameters of myocardium. Similarly, the second region utilised by both phantoms is a single, vascular, structure, from the VMR, with the acoustic parameters for blood. The final region in each phantom is a set of sparse points representative of Ultrasound Contrast Agents (UCAs). By utilising the flow information from the VMR dataset, the positions of these sparse points are updated automatically within the framework and stored in a separate phantom region. The second phantom utilises the new set of positions, representing the UCA positions after 5 ms.

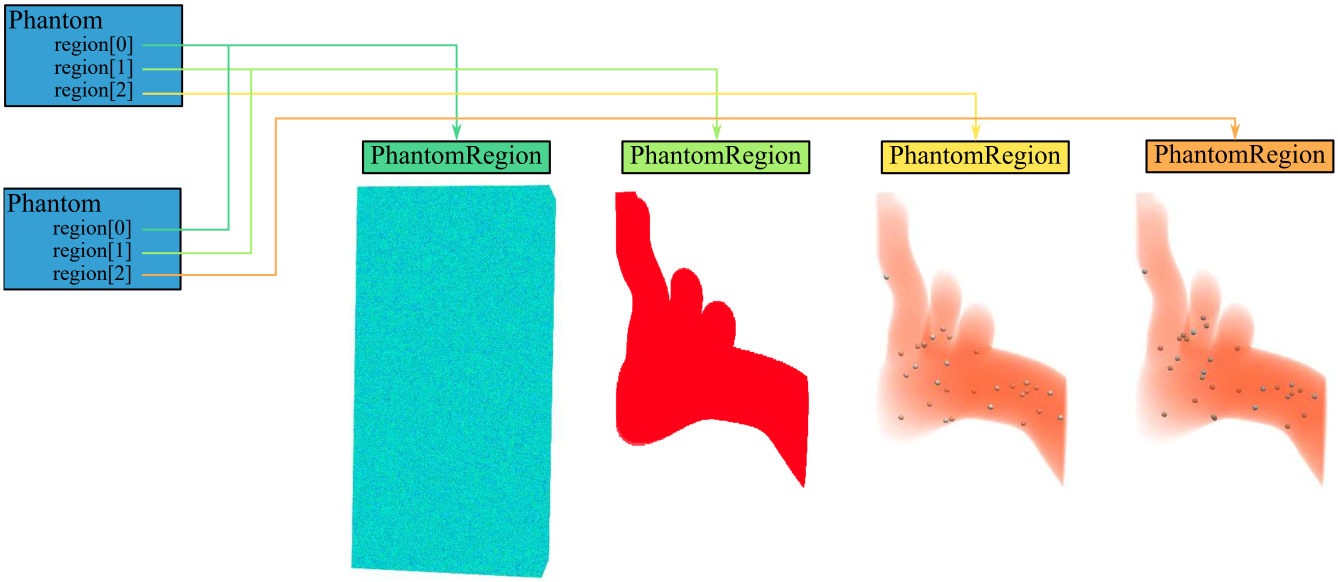

Fig. 3. Two USim phantom definitions for an aortic arch from the Vascular Model Repository (VMR) incorporating Ultrasound Contrast Agents (UCAs). The positions of the UCAs are updated between the two phantoms based on the blood flow velocity. The second phantom represents the same aorta 5 ms later.

C. USim Data Format

Simulations are computationally expensive, as such, it is preferable to run a simulation once and reuse the results wherever possible. This is increasingly important for data-driven algorithms that require large simulation datasets. In addition, for validation and comparison, it is important that simulation results can be shared and that results can be reproduced. To achieve this, however, it is necessary to save a simulation, including the entire configuration, in a well defined and organised structure.

Consequently, the format chosen should support complex structures, relationships and datasets. It should also be portable across multiple platforms and operating systems. For USim, the HDF5® (HDF Group, Champaign, IL, USA) file format was chosen for this purpose. It is designed to manage complex datasets. It works across multiple platforms, can support large amounts of data and retains data and metadata in the same file.

The USim HDF5® file format contains groups for phantoms, phantom regions, transducers and simulations. Soft symbolic links are used to link a phantom region to a phantom it is used within. This allows regions to be reused across multiple phantoms. Likewise, soft symbolic links are used to link a phantom and transducer to the simulation it is used within. This ensures the relationship is retained and prevents unnecessary repeated data being saved into the file.

III. RESULTS AND DISCUSSION

To demonstrate the utility of the USim framework for volumetric flow imaging algorithm development, it was used to simulate a high-frame-rate (HFR) imaging sequence. Diverging wave (DW) imaging [11] was simulated using a matrix array probe model. The framework was used to model a 32 × 32 matrix array probe with a central frequency of 5 MHz and a fractional bandwidth of 60 %. An element pitch of 300 μm in both the lateral and elevational dimension was used. A single virtual focal length of -27 mm and a steering angle of 0 ° was used with a 2-cycle, 5 MHz, excitation waveform. A Pulse Repetition Frequency (PRF) of 1 kHz was used and phantoms, as illustrated in Fig. 3, were generated for each time-step corresponding to this PRF.

Fig. 4(a) shows the beamformed frame generated from a single DW emission. This can be compared to Fig. 4(d) which shows the ground-truth aorta structure. Fig. 4(b) shows the same frame after spatiotemporal clutter filtering [12] is performed using 20 frames. The tissue background has been suppressed and the particles clearly coincide with the ground-truth locations illustrated in Fig. 4(d). Particle localisation methods, based on centroid detection, were used to locate the particles in all 20 frames and the corresponding particles were tracked between sequential frames to generate particle velocity streamlines. 9 of these streamlines are shown in Fig. 4(c). These are comparable to the ground-truth streamlines illustrated in Fig. 4(d), albeit with a greater degree of noise in the measured velocity magnitude.

In the development of USim, several design principles have been adopted. The first of these is usability. The API and simulation objects have been designed at a higher level of abstraction than the underlying acoustic wave simulator. This has the benefit of simplifying the description of simulations that represent complex imaging sequences and phantom structures. It also ensures the simulation specification is independent of the underlying simulator. Whilst, in the first instance, only the k-Wave simulation engine is supported, the incorporation of other simulation engines remains the subject of future work. In applications where nonlinearity, multiple scattering

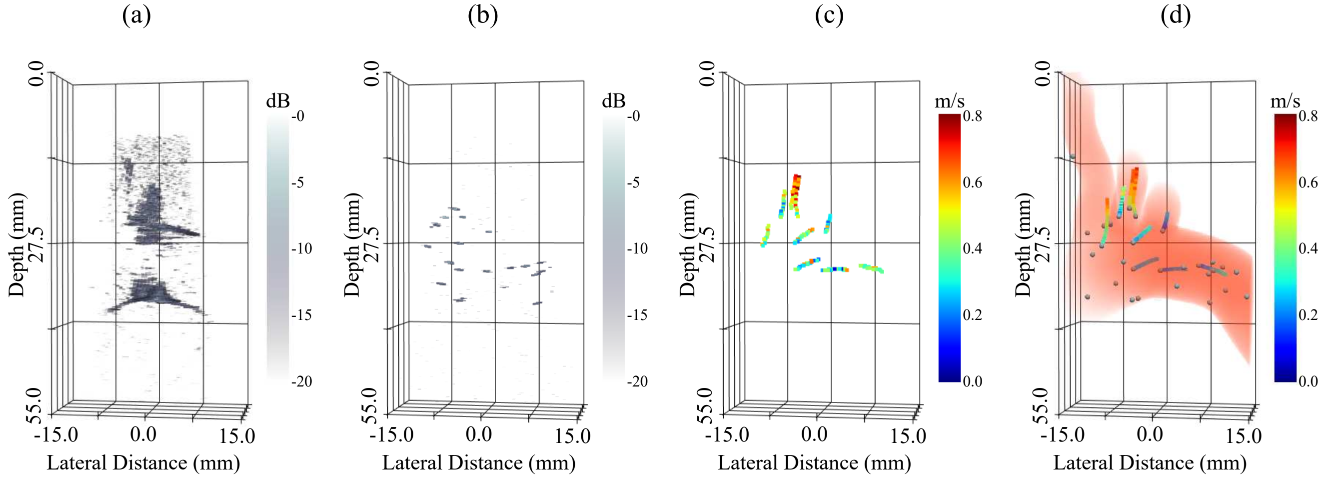

Fig. 4. (a) Beamformed volume generated from a Diverging Wave (DW) imaging simulation. (b) The same beamformed volume after tissue background removal using spatiotemporal filtering of 20 frames. (c) Localised and tracked particle velocity streamlines of 9 scatterers form 20 frames. (d) Ground-truth simulated phantom of the anatomical structure, particle locations and particle velocities of an aortic arch from the Vascular Model Repository (VMR).

and speed of sound heterogeneity are of less importance, and precise localisation of moving sparse scatterers is of greater importance, such as Super Resolution Imaging (SRI) [13], a particle-based simulator is advantageous.

The second design principle adopted is flexibility. The USim phantom objects have been defined in a way that allows a range of phantoms to be defined. In the example described in this work, for instance, the ability to define both anatomical structures, on a uniform grid, and sparse scatterers, at arbitrary points, was exploited. The USim simulation objects have been defined in a way that allows a range of imaging sequences to be described. For instance, the ability to describe arbitrary excitation waveforms, including arbitrary delays, on every element of the transducer array is inherent in the framework.

The third design principle adopted is extensibility. Where the framework does not meet user needs, users are encouraged to extend functionality by developing new classes that inherit the USim super class functionality.

IV. CONCLUSION

The USim framework can facilitate ultrasound beamforming and functional algorithm development and validation in the realms of volumetric flow imaging. The framework is designed at a higher level of abstraction than the underlying simulator, enabling the possibility of use with a variety of simulators in the future. In a volumetric ultrasound imaging simulation of flowing particles in an *in-silico* model of a human aorta, the particles in the beamformed lumen were successfully localised and tracked between subsequent frames. Consequently, the simulated dataset can facilitate volumetric flow imaging algorithm development for cardiovascular functional assessment.

ACKNOWLEDGEMENT

This research was partially supported by EPSRC Grant number EP/V04799X/1. The authors would like to thank the NVIDIA Corporation and its Applied Research Accelerator Program for donating an RTX A6000 48 GB GPU.

REFERENCES

[1] H. J. Vos et al., “Contrast-Enhanced High-Frame-Rate Ultrasound Imaging of Flow Patterns in Cardiac Chambers and Deep Vessels,” Ultrasound Med. Biol., vol. 46, no. 11, pp. 2875–2890, Nov. 2020.
[2] H. J. Ling et al.,“Physics-Guided Neural Networks for Intraventricular Vector Flow Mapping,” IEEE Trans. Ultrason. Ferroelectr. Freq. Control, in press.
[3] J. A. Jensen, “FIELD: A program for simulating ultrasound systems,” Med. Biol. Eng. Comput., vol. 34, pp. 351–352, Aug. 1996.
[4] B. E. Treeby and B. T. Cox, “k-Wave: MATLAB toolbox for the simulation and reconstruction of photoacoustic wave fields,” J. Biomed. Opt., vol. 15, no. 2, p. 021314, Mar. 2010.
[5] D. Garcia, “SIMUS: An open-source simulator for medical ultrasound imaging. Part I: Theory & examples,” Comput. Methods Programs Biomed., vol. 218, p. 106726, May 2022.
[6] A. Cigier, F. Varray, and D. Garcia, “SIMUS: An open-source simulator for medical ultrasound imaging. Part II: Comparison with four simulators,” Comput. Methods Programs Biomed., vol. 220, p. 106774, Jun. 2022.
[7] A. Updegrove et al., “SimVascular: An Open Source Pipeline for Cardiovascular Simulation,” Ann. Biomed. Eng., vol. 45, no. 3, pp. 525–541, Mar. 2017.
[8] C. J. Arthurs et al., “CRIMSON: An open-source software framework for cardiovascular integrated modelling and simulation,” PLOS Comput. Biol., vol. 17, no. 5, p. e1008881, May 2021.
[9] N. M. Wilson, A. K. Ortiz, and A. B. Johnson, “The Vascular Model Repository: A Public Resource of Medical Imaging Data and Blood Flow Simulation Results,” J. Med. Device., vol. 7, no. 4, p. 1, Dec. 2013.
[10] P. Hasgall et al., “IT’IS Database for thermal and electromagnetic parameters of biological tissues,” 2018. [Online]. Available: itis.swiss/database.
[11] L. Nie et al., “High Frame-Rate Contrast-Enhanced Echocardiography using Diverging Waves: 2-D Motion Estimation and Compensation,” IEEE Trans. Ultrason. Ferroelectr. Freq. Control, vol. 66, no. 2, pp. 359–371, Feb. 2019.
[12] C. Demene et al., “Spatiotemporal Clutter Filtering of Ultrafast Ultrasound Data Highly Increases Doppler and fUltrasound Sensitivity,” IEEE Trans. Med. Imaging, vol. 34, no. 11, pp. 2271–2285, Nov. 2015.
[13] K. Christensen-Jeffries et al., “Super-resolution ultrasound imaging,” Ultrasound Med. Biol., vol. 46, no. 4, pp. 865-891, Apr. 2020.